อัลกอริทึมใหม่สำหรับการจำแนกประเภทเอกสาร โดยใช้กฎความสัมพันธ์

TEXT CATEGORIZATION USING A NEW ASSOCIATION RULE-BASED CLASSIFIER ALGORITHM

สุภาภรณ์ บุตรดีวงษ์
SUPAPORN BUDDEEWONG

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2549

ISBN 974-15-2156-1

อัลกอริทึมใหม่สำหรับการจำแนกประเภทเอกสาร โดยใช้กฎความสัมพันธ์

TEXT CATEGORIZATION USING A NEW ASSOCIATION RULE-BASED CLASSIFIER ALGORITHM

สุภาภรณ์ บุตรดีวงษ์

SUPAPORN BUDDEEWONG

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2549

ISBN 974-15-2156-1

TEXT CATEGORIZATION

USING A NEW ASSOCIATION RULE-BASED CLASSIFIER ALGORITHM

SUPAPORN BUDDEEWONG

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

ISBN 974-15-2156-1

หัวข้อวิทยานิพนธ์	อัลกอริทึมใหม่สำหรับการจำแนกประเภทเอกสาร โดยใช้กฎความสัมพันธ์
นักศึกษา	นางสาว สุภาภรณ์ บุตรดีวงษ์
รหัสนักศึกษา	45066003
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
พ.ศ.	2549
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ.ดร.วรพจน์ กรีสุระเดช

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนออัลกอริทึมใหม่สำหรับใช้จำแนกเอกสารโดยใช้กฎความสัมพันธ์ (Association Rule-Based Text Classifier: ARTC) มีวัตถุประสงค์เพื่อปรับปรุงการจำแนกเอกสารของอัลกอริทึม Association Rule-based Classifier By Categories (ARC-BC) ซึ่งเป็นการที่ใช้กฎความสัมพันธ์ในการจำแนกประเภทเอกสารโดยใช้กฎความสัมพันธ์ที่สามารถจำแนกเอกสารได้ดีหากเอกสารนั้น ๆ มีลักษณะเอกสารที่กลุ่มไม่ซ้อนทับกัน แต่ในความเป็นจริงของเอกสารที่มีอยู่ในปัจจุบันนั้นมีเอกสารทั้งที่ไม่มีกลุ่มซ้อนทับกัน และกลุ่มซ้อนทับกัน ดังนั้นวิทยานิพนธ์ฉบับนี้จึงได้นำเสนอวิธีการจำแนกเอกสารโดยใช้กฎความสัมพันธ์ที่สามารถจำแนกเอกสารที่มีลักษณะซ้อนทับกันและไม่ซ้อนทับกันได้เป็นอย่างดี โดยเริ่มจากการค้นหากฎความสัมพันธ์ของข้อมูล ซึ่งได้ Frequent Itemset 2 ชนิด คือ Frequent Itemset ที่เก็บคุณลักษณะของเอกสาร ที่ไม่ซ้อนทับกับกลุ่มใด ๆ และ Frequent Itemset ที่เก็บคุณลักษณะของเอกสารที่ซ้อนทับกันมากกว่าหนึ่งกลุ่ม ซึ่งเกิดจากการเชื่อมความสัมพันธ์แบบใหม่ที่เรียกว่า ARTC join เมื่อได้ความสัมพันธ์ที่จะนำไปสร้างเป็นกฎความสัมพันธ์แล้ว นำกฎความสัมพันธ์ที่ได้มาคัดทิ้ง โดยใช้โครงสร้างต้นไม้ ในการคัดกฎความสัมพันธ์ที่ไม่จำเป็นทิ้ง ก่อนที่จะนำกฎที่ได้ไปจำแนกประเภทเอกสาร ผลการทดลองที่ได้อัลกอริทึมที่นำเสนอ ให้อัตราความถูกต้อง (Accuracy rate)ในการจำแนกประเภทเอกสารได้ดีกว่าอัลกอริทึม ARC-BC

Thesis Title	Text Categorization using a new Association Rule-Based Classifier Algorithm
Student	Miss Supaporn Buddeewong
Student ID.	45066003
Degree	Master of Science
Programme	Information Technology
Year	2006
Thesis Advisor	Asst.Prof.Dr.Worapoj Kreesuradej

ABSTRACT

This thesis proposes a new Association Rule-Based Text Classifier (ARTC) algorithm to improve the prediction accuracy of Association Rule-based Classifier By Categories (ARC-BC) algorithm. ARC-BC has shown a good performance. In addition, the classifier based on ARC-BC algorithm produces clear and understandable results. However, the classifier can not work well for the single-class document that has some terms of document mutually associated with other classes. Unlike ARC-BC algorithm, a new Association Rule-Based Text Classifier (ARTC) algorithm consists of three main phase to construct a classifier. The first phase is association rule generation. The proposed association rule generation algorithm constructs two types of frequent itemsets. The first frequent itemset contain all terms that have no an overlap with other categories. The second frequent itemset contain all features that have an overlap with other categories that generated by a new join method, ARTC join. The second pahse is pruning step. The pruning step uses tree structure for pruning association rule that have confidence value more that a threshold value of confidence factor. The last phase is the prediction of classes associated with new documents. The experimental results are shown a good performance of the proposed classifier.

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จลุล่วงได้อย่างดี ด้วยความกรุณาจากอาจารย์ผู้ควบคุมวิทยานิพนธ์ ผศ.ดร.วรพจน์ กรีสุระเดช ที่ให้คำปรึกษา ชี้แนะแนวทางในการทำวิจัย และการแก้ปัญหาของงานวิจัยนี้จนสำเร็จลุล่วง ตลอดจนประสิทธ์ประสาทวิชาความรู้อื่น ๆ ที่ไม่สามารถหาในห้องเรียนใด ๆ ได้

ขอขอบคุณสำนักหอสมุดมหาวิทยาลัยเกษตรศาสตร์ หน่วยงานสังกัดการทำงาน ที่ให้เวลาในการทำวิจัยอย่างเต็มที่จนสำเร็จลุล่วง

ขอขอบคุณ DME lab ที่สนับสนุนอุปกรณ์การทำวิจัยจนสำเร็จ

ขอขอบคุณ นายอัศวิน มีเงิน นักศึกษาคณะเทคโนโลยีสารสนเทศ (IS 12.1) ที่ช่วยเหลือให้คำแนะนำในเรื่องการเขียนโปรแกรมสำหรับเตรียมข้อมูลในการทดลองของงานวิจัยนี้

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดา และพี่น้องร่วมอุทรที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่อง จนทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่า และประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่บิดา มารดา อันเป็นที่รักยิ่ง ซึ่งเป็นผู้ให้กำเนิดและทำให้ข้าพเจ้าได้มีวันนี้

สุภาภรณ์ บุตรดีวงษ์

สารบัญ

สารบัญ(ต่อ)

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

สารบัญตาราง(ต่อ)

สารบัญตาราง(ต่อ)

สารบัญตาราง(ต่อ)

สารบัญรูป

สารบัญรูป(ต่อ)

สารบัญรูป(ต่อ)

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากความเจริญก้าวหน้าทางเทคโนโลยีคอมพิวเตอร์และอินเตอร์เน็ตเป็นไปอย่างรวดเร็ว ทำให้ข้อมูลอิเล็กทรอนิกส์เกิดขึ้นอย่างมากมายในแต่ละวัน ซึ่งข้อมูลส่วนใหญ่จะเป็นข้อมูลประเภทข้อความ (Text) ที่เก็บอยู่ในฐานข้อมูล การจัดการข้อมูลขนาดใหญ่ที่เกิดขึ้นเกินกว่ากำลังคนจะสามารถจัดการได้ ดังนั้นจึงจำเป็นที่จะต้องมีเครื่องมือช่วยในการวิเคราะห์ข้อมูลเพื่อค้นหาสิ่งที่เป็นประโยชน์ออกมาจากข้อมูล เท็กซ์ไมน์นิ่ง (Text mining) เป็นวิธีการหนึ่งที่สามารถนำมาวิเคราะห์หาความรู้ที่ซ่อนอยู่ในข้อมูลมาใช้ประโยชน์ได้ ซึ่งวิธีที่นิยมใช้ในการทำเท็กซ์ไมน์นิ่งได้แก่ การจัดกลุ่มเอกสาร (Text Clustering) และ การจำแนกกลุ่มเอกสาร (Text Classification หรือ Text Categorization)

วิธีการในการจำแนกเอกสาร มีหลายวิธี [1, 2] เช่นBayesian Networks, Decision Trees, Neural Networks, Support Vector Machines (SVM), k-Nearest Neighbor (k-NN) และการค้นหากฎความสัมพันธ์ (Association rule discovery) ซึ่งการค้นหากฎความสัมพันธ์เป็นวิธีที่วิทยานิพนธ์ฉบับนี้เลือกใช้ในการจำแนกประเภทเอกสาร

การค้นหากฎความสัมพันธ์ เป็นการค้นหาความสัมพันธ์ของข้อมูลแล้วนำมาสร้างเป็นกฎ (Rule) ที่เรียกว่ากฎความสัมพันธ์ (Association rule) และนำกฎความสัมพันธ์ที่ได้มาใช้ในการจำแนกเอกสาร อัลกอริทึม Association Rule-Based Classifier By Category (ARC-BC) เป็นวิธีการหนึ่งที่ใช้การจำแนกเอกสารโดยใช้กฎความสัมพันธ์ ซึ่งให้ผลลัพธ์ในการจำแนกเอกสารได้ดี แต่อย่างไรก็ตามหากเอกสารที่ใช้ในการจำแนกมีลักษณะข้อมูลที่ซ้อนทับกันหลายกลุ่ม อัลกอริทึม ARC-BC จะจำแนกไม่ได้ดี ดังนั้นวิทยานิพนธ์ฉบับนี้จึงได้นำเสนอวิธีใหม่ เรียกว่าอัลกอริทึม Association Rule-Based Text Classifier (ARTC) ซึ่งสามารถค้นพบกฎความสัมพันธ์ของข้อมูลทั้งที่ไม่ซ้อนทับกัน และซ้อนทับกันได้ ทำให้สามารถจำแนกเอกสารที่มีลักษณะดังกล่าวได้ถูกต้องมากว่าอัลกอริทึม ARC-BC

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์ฉบับนี้มีวัตถุประสงค์เพื่อนำเสนอวิธีใหม่ในการค้นหากฎความสัมพันธ์ และการจำแนกประเภทเอกสารที่ให้ความถูกต้องมากกว่าวิธีเดิม โดยนำผลลัพธ์ที่ได้ในการจำแนกประเภทเอกสารเปรียบเทียบกับอัลกอริทึม ARC-BC

1.3 สมมติฐานของการศึกษา

ข้อด้อยของอัลกอริทึม ARC-BC คือการค้นหา Frequent Itemset ที่ได้ปริมาณมากเกินไป ทำให้เสียเวลามากในการค้นหาความสัมพันธ์เพื่อนำมาสร้างกฎความสัมพันธ์ นอกจากนี้ยังมีข้อจำกัด คือการจำแนกประเภทเอกสาร ซึ่งสามารถจำแนกได้ดีหากเป็นเอกสารกลุ่มเดียวที่ไม่มีความสัมพันธ์กับกลุ่มอื่นหรือไม่มีการซ้อนทับกัน (Non overlap) ของคุณลักษณะเอกสาร นั่นคือหากเอกสารที่ต้องการจำแนกประเภทเอกสารมีความสัมพันธ์กับกลุ่มอื่นหรือมีการซ้อนทับกัน (Overlap) ของคุณลักษณะเอกสาร ประสิทธิภาพในการจำแนกประเภทเอกสารจะลดน้อยลงมากกว่าการจำแนกประเภทเอกสารที่ไม่มีความสัมพันธ์กับกลุ่มใด

การแก้ปัญหาข้างต้นนี้ เราได้นำเสนอวิธีการค้นหากฎความสัมพันธ์แบบใหม่ซึ่งจะค้นหาเฉพาะความสัมพันธ์ที่สามารถนำมาสร้างเป็นกฎความสัมพันธ์ได้ ซึ่งจะลดจำนวนความสัมพันธ์ที่ไม่จำเป็นต้องใช้ในการจำแนกประเภทเอกสาร นอกจากนี้ยังขยายประสิทธิภาพการจำแนกประเภทเอกสารที่ไม่มีความสัมพันธ์กับกลุ่มอื่น และมีความสัมพันธ์กับกลุ่มอื่นให้ดีมากกว่าเดิม

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

การค้นหากฎความสัมพันธ์ เป็นการค้นหาความสัมพันธ์ของข้อมูลในฐานข้อมูล โดยความสัมพันธ์ที่ได้สามารถนำมาทำนายแนวโน้มของข้อมูลหรือนำมาทำนายประเภทเอกสารได้ ข้อดีของการหากฎความสัมพันธ์มีดังนี้

1. ทำงานได้ดีกับข้อมูลขนาดใหญ่ ขณะที่เทคนิคอื่น ๆ จะมีปัญหาการทำงานกับข้อมูลปริมาณมาก
2. ผู้ใช้สามารถระบุค่าสนับสนุนน้อยที่สุด (Minimum support) และค่าความเชื่อมั่นน้อยที่สุด (Minimum confidence) ได้ ทำให้ควบคุมผลลัพธ์ได้
3. ง่ายต่อการทำความเข้าใจเพราะใช้สัญลักษณ์ ในการแสดงผล

วิธีการของอัลกอริทึม ARC-BC สามารถจำแนกเอกสารได้ดีในกรณีที่ลักษณะของข้อมูลไม่มีการซ้อนทับกัน หากข้อมูลที่ใช้ในการจำแนกเอกสารมีลักษณะทีซ้อนทับกันจะทำให้การจำแนกเอกสารของอัลกอริทึม ARC-BC แย่ลง ดังนั้นวิทยานิพนธ์ฉบับนี้จึงได้เสนออัลกอริทึม ARTC ที่สามารถจำแนกเอกสารได้ดีในลักษณะของข้อมูลดังที่กล่าวมาแล้ว

1.5 ขอบเขตการวิจัย

1.นำเสนออัลกอริทึมใหม่ในการค้นหากฎความสัมพันธ์ของข้อมูล เพื่อใช้ ในการจำแ นกประเภทเอกสารประเภทข้อความ

2. นำเสนอวิธีคัดกฎความสัมพันธ์ทิ้งโดยใช้หลักของ Tree

3. อัลกอริทึมนี้ใช้สำหรับข้อมูล 1 มิติ โดยงานวิจัยนี้ใช้คำสำคัญ

4. การจำแนกประเภทเอกสารใด ๆ จะอยู่ได้เพียง 1 กลุ่ม (Single class) เท่านั้น

5.ข้อมูลในการทดลองคือ ชุดข้อมูลข้อความ ชุดข้อมูล CSTR ชุดข้อมูล K-dataset และชุดข้อมูล Reuters-Top10

6.ใช้อัตราความถูกต้อง (Accuracy rate) ในการจำแนกประเภทเอกสารเป็นตัววัดประสิทธิภาพ

1.6 ขั้นตอนของการศึกษา

ในขั้นตอนของการศึกษานี้ ได้แสดงลำดับการทำงานตั้งแต่เริ่มต้นจนถึงสิ้นสุดการทำงานวิจัย ดังรายละเอียดต่อไปนี้

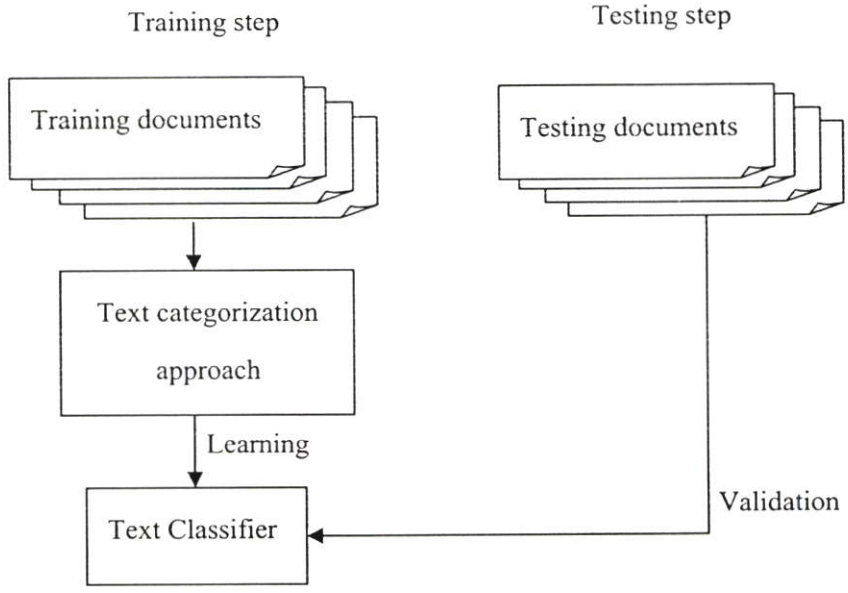
1. ศึกษาทฤษฎีและงานวิจัยจากเอกสาร บทความต่าง ๆ ที่เกี่ยวข้องกับการทำงานวิจัย
2. กำหนดหัวข้อ เป้าหมาย วัตถุประสงค์ และขอบเขตการทำงานวิจัย
3. วิเคราะห์และออกแบบอัลกอริทึมใหม่
4. เขียนโปรแกรมสร้างชุดข้อมูลข้อความ
5. เขียนโปรแกรมของงานวิจัยที่เกี่ยวข้อง และงานวิจัยของวิทยานิพนธ์นี้
6. ทดสอบโปรแกรมกับชุดข้อมูลข้อความ
7. ค้นหาชุดข้อมูลที่จะนำมาใช้ในการทดลอง (CSTR, K-dataset, Reuters- Top10)
8. เตรียมข้อมูลที่จะนำมาใช้ทดลอง
9. พัฒนาโปรแกรมและแก้ไขข้อผิดพลาดโปรแกรมของงานวิจัยนี้
10. ทดสอบอัลกอริทึมกับชุดข้อมูล CSTR, K-dataset และ Reuters- Top10
11. รวบรวมผลการทดลอง จากผลการทำงานของโปรแกรม
12. วิเคราะห์และสรุปผลการดำเนินงาน
13. จัดทำเอกสารประกอบงานวิจัย

บทที่ 2
ทฤษฎีพื้นฐาน และงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ และงานวิจัยที่เกี่ยวข้องในการทำวิจัย โดยเนื้อหาในบทนี้จะกล่าวถึง การจำแนกประเภทเอกสาร (Text categorization) การค้นหากฎความสัมพันธ์ (Association rule discovery) และงานวิจัยที่เกี่ยวข้องซึ่งก็คืออัลกอริทึม ARC-BC

2.1 การจำแนกประเภทเอกสาร

การจำแนกประเภทเอกสาร (Text categorization) เป็นกระบวนการในการจำแนกเอกสารใหม่ที่ยังไม่ทราบกลุ่มให้อยู่ในกลุ่มที่เหมาะสมสำหรับเอกสารนั้น ๆ สามารถแสดงเป็นรูปภาพดังดังรูปที่ 2.1



รูปที่ 2.1 กระบวนการการจำแนกเอกสาร

จากรูปที่ 2.1 แสดงกระบวนการการจำแนกเอกสาร ซึ่งแบ่งออกเป็น 2 ขั้นตอนดังนี้

1. การฝึกหัดระบบ (Training step) เป็นขั้นตอนสร้างตัวจำแนกประเภทเอกสาร (Text Classifier) โดยการนำเอกสารสำหรับฝึกหัดระบบ (Training documents) ไปใช้ในวิธีการจำแนกประเภทเอกสาร (Text categorization approach) เช่น Neural network, k-Nearest Neighbor, Support Vector Machine (SVM) และ Association rule เป็นต้น ซึ่งเป็นการเรียนรู้ (Learning) ถึงแบบแผนของข้อมูลจากกลุ่มข้อมูลที่จัดเตรียมไว้ โดยวิธีนี้เรียกว่าเป็นการเรียนรู้แบบมีการชี้นำ

(Supervised learning) ลักษณะของเอกสารแต่ละเอกสารในกลุ่มของเอกสารสำหรับฝึกหัดระบบ แสดงได้ดังนี้

$$D_i = \{c_1, c_2, \ldots, c_n, t_1, t_2, \ldots, t_n\} \quad (2.1)$$

โดยที่ D_i คือ เอกสารที่ i

C คือชื่อกลุ่ม โดยที่ $C = \{c_1, c_2, \ldots, c_n\}$

T คือคุณลักษณะของเอกสาร โดยที่ $T = \{t_1, t_2, \ldots, t_n\}$

2. การทดสอบระบบ (Testing step) เป็นขั้นตอนการทดสอบประสิทธิภาพตัวจำแนกประเภทเอกสาร โดยใช้กลุ่มเอกสารสำหรับทดสอบระบบ (Testing documents) ซึ่งเอกสารสำหรับทดสอบระบบนั้นจะเป็นเอกสารที่มีลักษณะเช่นเดียวกับ (2.1) แต่ในการนำเอกสารเข้าทดสอบนั้นจะนำเข้าเฉพาะคุณลักษณะของเอกสารดังนี้

$$D_j = \{t_1, t_2, \ldots, t_n\} \quad (2.2)$$

โดยที่ D_j คือ เอกสารที่ j

T คือคุณลักษณะของเอกสาร โดยที่ $T = \{t_1, t_2, \ldots, t_n\}$

2.2 การค้นหากฎความสัมพันธ์

การค้นหากฎความสัมพันธ์ (Association rule discovery) เป็นการค้นหาความสัมพันธ์ระหว่างรายการในแต่ละรายการหรือกลุ่มของรายการ ที่ปรากฏขึ้นในฐานข้อมูล ความสัมพันธ์ที่ได้สามารถบอกลักษณะของข้อมูลหรือทำนายลักษณะของข้อมูลต่อไปได้ โดยทั่วไปความสัมพันธ์จะปรากฏอยู่ในรูปของกฎ "ถ้า ... แล้ว ..." (If ... Then ...) ซึ่งในกฎหนึ่ง ๆ ประกอบด้วย 2 ส่วนคือ ส่วนด้านซ้ายของกฎ (ส่วน "ถ้า" หรือ Rule body หรือ Antecedent หรือ Left-hand side) และส่วนด้านขวาของกฎ (ส่วน "แล้ว" หรือ Rule head หรือ Consequent หรือ Right-hand side) โดยส่วนด้านซ้ายอาจประกอบด้วยหนึ่งหรือมากกว่าหนึ่งเงื่อนไขที่เป็นจริง ที่จะทำให้ส่วนด้านขวาของกฎเป็นจริง เช่น "ถ้า T แล้ว C" (If T Then C) ใช้สัญลักษณ์แทน "T => C" หมายถึง ถ้าเกิด T แล้วจะเกิด C ด้วย หากนำมาใช้สำหรับการจำแนกประเภทเอกสาร จะตีความได้ว่า ถ้าเอกสารมีคุณลักษณะ T แล้วจะบอกได้ว่าเอกสารอยู่กลุ่ม C โดยที่ T เป็นคุณลักษณะของเอกสารและ C เป็นชื่อกลุ่มเอกสาร

นิยามและความหมายของคำต่าง ๆ ที่ใช้ในการค้นหากฎความสัมพันธ์ได้แก่

1. ค่าสนับสนุน (Support value) เป็นค่าแสดงความสัมพันธ์ระหว่างจำนวนของเหตุการณ์ที่เกิดขึ้นระหว่างจำนวนเอกสารที่มีคุณลักษณะ T และบอกว่าอยู่กลุ่ม C กับจำนวนรายการที่เกิดขึ้นทั้งหมด สามารถแสดงเป็นสมการได้ดังสมการที่ 2.3

$$Support = \frac{n}{N} \tag{2.3}$$

โดยที่ n คือจำนวนเอกสารที่มีคุณลักษณะ T และบอกว่าอยู่กลุ่ม C
N คือจำนวนเอกสารทั้งหมด

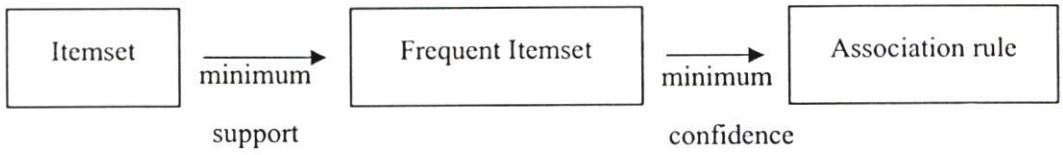
2. ค่าสนับสนุนน้อยที่สุด (Minimum support) คือค่าสนับสนุนที่น้อยที่สุดที่ทำให้ความสัมพันธ์ที่ได้นั้นยังมีความน่าสนใจ
3. ค่าความเชื่อมั่น (Confidence value) เป็นค่าแสดงความเป็นจริงของกฎ สามารถคำนวณได้ แสดงดังสมการที่ 2.4

$$Confidence = \frac{n}{M} \tag{2.4}$$

โดยที่ n คือจำนวนเอกสารที่มีคุณลักษณะ T และบอกว่าอยู่กลุ่ม C
M คือจำนวนเอกสารที่มีคุณลักษณะ T

4. ค่าความเชื่อมั่นน้อยที่สุด (Minimum confidence) คือค่าความเชื่อมั่นที่น้อยที่สุดที่ทำให้กฎความสัมพันธ์ที่ได้นั้นยังมีความน่าสนใจ
5. ไอเทม (Item) คือข้อมูลแต่ละตัวที่ใช้ในการหากฎความสัมพันธ์ เช่น tour, money, management, market เป็นต้น
6. ไอเทมเซต (Itemset) คือ ความสัมพันธ์ของข้อมูลที่หาได้ Itemset ประกอบด้วย Item โดย k-itemsets ประกอบด้วย k-items ใน itemset นั้น ๆ เช่น 2-itemsets ยกตัวอย่างเช่น {tour, money} , {tour, management} เป็นต้น และถ้า 3-itemsets ยกตัวอย่างเช่น {tour, money, management} , {tour, money, market} เป็นต้น
7. ไอเทมเซตตัวเลือก (Candidate Itemset) คือ Itemset ที่ได้จากการเชื่อมความสัมพันธ์ของ Itemset ก่อนหน้านี้ ซึ่งเป็นไอเทมเซตตัวเลือกสำหรับฟรีเควนท์ไอเทมเซต
8. ฟรีเควนท์ไอเทมเซต (Frequent Itemset) หรือ ลาร์จไอเทมเซต (Large Itemset) คือชุดของ Itemset ที่มีค่าสนับสนุนมากกว่าหรือเท่ากับค่าสนับสนุนน้อยที่สุด

การค้นหากฎความสัมพันธ์มีกระบวนการทำงาน 2 ขั้นตอนคือ หา Frequent Itemset ทั้งหมด และสร้างกฎความสัมพันธ์จาก Frequent Itemset แสดงดังรูปที่ 2.2



รูปที่ 2.2 ขั้นตอนการค้นหากฎความสัมพันธ์

2.2.1 อัลกอริทึม Apriori

อัลกอริทึมที่นิยมใช้ในการหากฎความสัมพันธ์ของข้อมูลคือ อัลกอริทึม Apriori ซึ่งเป็นอัลกอริทึมดั้งเดิมสำหรับหา Frequent Itemset ถึงแม้ว่าจะมีอัลกอริทึมอื่นที่มีประสิทธิภาพดีกว่าแต่ก็มีพื้นฐานมาจากอัลกอริทึมนี้เป็นส่วนใหญ่ ดังนั้นจึงใช้อัลกอริทึมนี้ในการอธิบายการหากฎความสัมพันธ์ ซึ่งประกอบด้วย 2 ขั้นตอนคือ การหา Frequent Itemset และการสร้างกฎความสัมพันธ์จาก Frequent Itemset โดยแต่ละขั้นตอนมีวิธีการต่าง ๆ อธิบายดังนี้

2.2.1.1 การหา Frequent Itemset

การหา Frequent Itemset มีขั้นตอนการทำงานที่ทำซ้ำไปเรื่อย ๆ จนกว่าจะไม่สามารถหา Frequent Itemset ได้อีก กล่าวคือ Frequent k-Itemsets จะถูกใช้ในการหา Frequent (k+1)-Itemsets (ในที่นี้ใช้ L_1 เป็นสัญลักษณ์แทน Frequent 1-Itemset และ L_k เป็นสัญลักษณ์แทน Frequent k-Itemsets) กล่าวคือ L_1 จะถูกใช้ในการหา Frequent 2-Itemsets หรือ L_2 และ L_2 ก็จะถูกใช้เพื่อหา Frequent 3-Itemsets หรือ L_3 เช่นนี้ไปเรื่อย ๆ จนกว่าจะไม่สามารถ Frequent Itemset ได้อีก เพื่อเป็นการเพิ่มประสิทธิภาพของอัลกอริทึมโดยการช่วยลดพื้นที่ที่จะต้องค้นหา Frequent Itemset ในฐานข้อมูล กระทำโดย Itemset ใด ๆ ที่มีค่าสนับสนุนน้อยกว่าค่าสนับสนุนน้อยที่สุดที่ตั้งไว้ ถือว่า Itemset นั้น ๆ ไม่เป็น Frequent Itemset วิธีการดังกล่าวสามารถอธิบายตามขั้นตอนการหา Frequent Itemset ได้ดังนี้

1. ขั้นตอนการเชื่อมความสัมพันธ์ระหว่าง Itemset (Join step) เป็นขั้นตอนการสร้างแคนดิเดตท์ไอเทมเซต (Candidate Itemsets) หรือเป็น Itemset ตัวเลือกที่จะถูกเลือกไปเป็น Frequent Itemset โดย Candidate 1-Itemset แทนด้วนสัญลักษณ์ C_1 และ Candidate 2-Itemsets แทนด้วยสัญลักษณ์ C_2 ไปจนถึง Candidate k-Itemsets แทนด้วนสัญลักษณ์ C_k โดย C_k จะเกิดจากการเชื่อมความสัมพันธ์ของ $L_{(k-1)}$ และ $L_{(k-1)}$ วิธีการเชื่อความสัมพันธ์ทำโดยการพิจารณาว่าจะเชื่อมความสัมพันธ์จากจำนวน Itemset เท่าไหร่ ไปเป็นเท่าไหร่ โดยกำหนดให้หา C_k ซึ่งจะเกิดจากการเชื่อมความสัมพันธ์ระหว่าง $L_{(k-1)}$ และ $L_{(k-1)}$ พิจารณาว่า Itemset ใดสามารถเชื่อมความสัมพันธ์กันได้ ให้พิจารณาที่ Item ที่ 1 ถึง (k-2) ของทั้งสอง Itemset หากเมือนกันก็สามารถเชื่อมความสัมพันธ์กันได้ เช่น {I1, I2} ,{I1, I3}, {I2, I3} Itemset ที่ 1 และ 2 สามารถเชื่อมความสัมพันธ์กันได้ เพราะลำดับแรกของไอเทมเหมือนกัน (I1) เมื่อเชื่อมความสัมพันธ์จะได้ {I1,

I2, I3} และในการเชื่อมความสัมพันธ์ต่อ ๆ ไป Itemset ที่ 1 และ 3 ไม่สามารถเชื่อมความสัมพันธ์ได้เพราะลำดับแรกของ Item ทั้งสองไม่เหมือนกัน

2. ขั้นตอนการคัด Itemset ทิ้ง (Prune step) เป็นขั้นตอนการคัดสมาชิกใน C_k โดยมีกระบวนการการคัดสมาชิก 2 กระบวนการคือ

2.1 การคัดสมาชิกออกด้วยคุณสมบัติของ Apriori คือ C_k ที่ได้จากการเชื่อมความสัมพันธ์ของ $L_{(k-1)}$ และ $L_{(k-1)}$ นั้นจะถูกนำมาหาเซตย่อย (Sub set) หากเซตย่อยไม่ปรากฏใน $L_{(k-1)}$ นั่นคือ Itemset ใน C_k นั้นจะถูกตัดทิ้ง

2.2 การคัดสมาชิกออกด้วยค่าสนับสนุนน้อยที่สุด หลังจากการคัดสมาชิกออกด้วยคุณสมบัติของ Apriori แล้ว ต้องคัดสมาชิกออกอีกครั้งด้วยค่าสนับสนุนน้อยที่สุด โดยคัดสมาชิกใน C_k ที่มีความถี่น้อยกว่าค่าสนับสนุนน้อยที่สุดออก เพื่อสร้างเป็น L_k

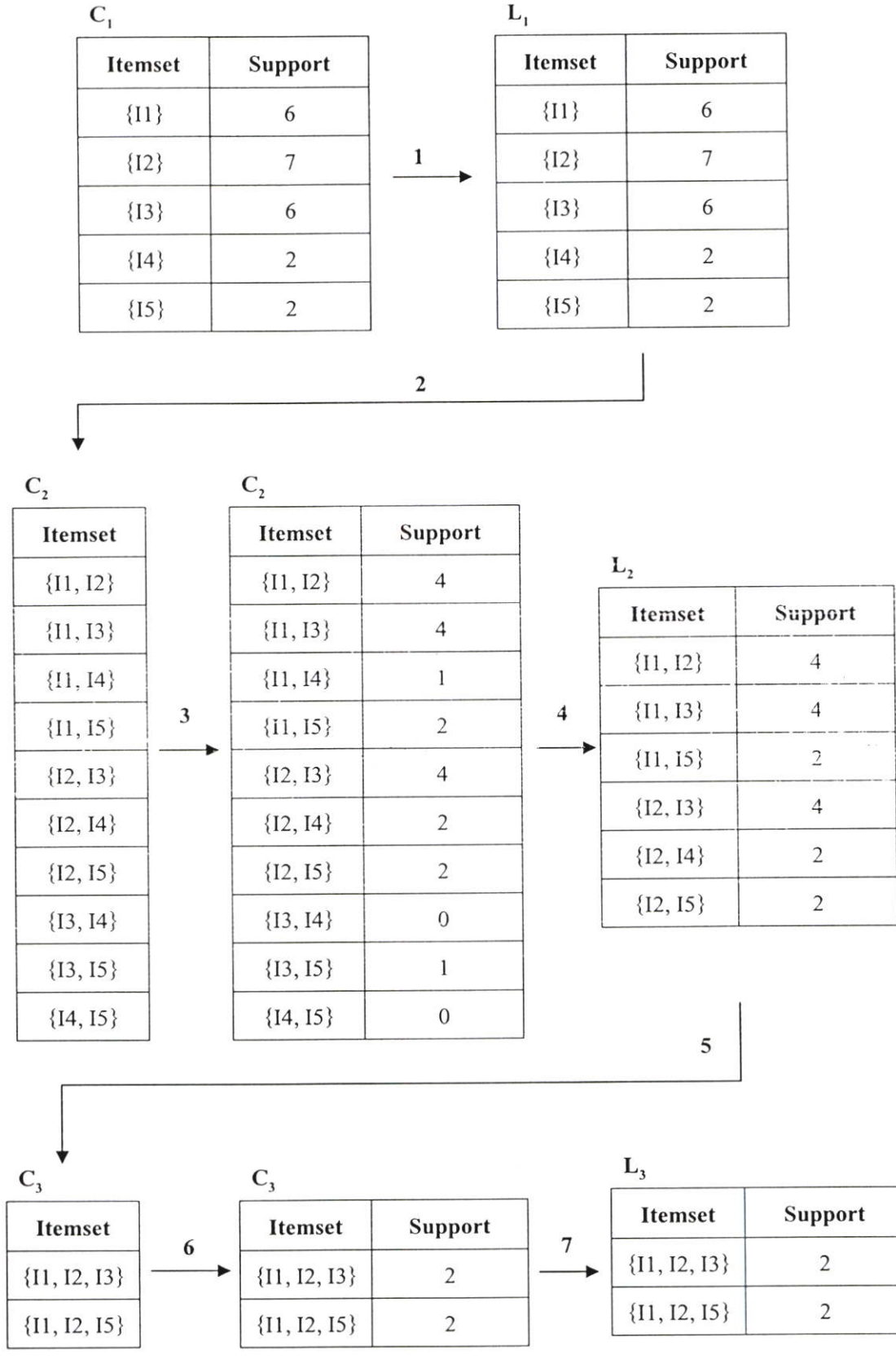
การหา Frequent Itemset จะกระทำวนซ้ำตามขั้นตอนที่ 1 และ 2 ไปเรื่อย ๆ จนกว่าจะไม่สามารถหา L_k ได้อีก จึงหยุดการหา Frequent Itemset วิธีการโดยละเอียดของการหา Frequent Itemset สามารถแสดงได้ดังตัวอย่างต่อไปนี้ [1]

กำหนดให้ฐานข้อมูล D มีจำนวนทรานแซกชัน (Transaction) 9 ทรานแซกชัน แสดงดังรูปที่ 2.3 และค่าสนับสนุนน้อยที่สุดมีค่าเท่ากับ 2 และค่าความเชื่อมั่นน้อยที่สุดมีค่า 70%

ลำดับ	รายการข้อมูล
1	I1, I2, I5
2	I2, I4
3	I2, I3
4	I1, I2, I4
5	I1, I3
6	I2, I3
7	I1, I3
8	I1, I2, I3, I5
9	I1, I2, I3

รูปที่ 2.3 ฐานข้อมูล D

จากฐานข้อมูล D ที่มีอยู่สามารถนำรายการข้อมูลที่มีอยู่มาหากฎความสัมพันธ์ของข้อมูลแต่ละรายการ โดยการหาFrequent Itemset แสดงดังรูปที่ 2.4



รูปที่ 2.4 แสดงวิธีหาFrequent Itemset

จากรูปที่ 2.4 แสดงการหาFrequent Itemsetจากฐานข้อมูล D เริ่มต้นจากหา C_1 โดยการแสกนหา 1- Itemset จากฐานข้อมูล D และนับความถี่ของแต่ละ Itemset ในฐานข้อมูล ขั้นตอนตามรูปที่ 2.5 อธิบายดังนี้

1. ขั้นตอนที่ 1 จาก C_1 เป็น L_1 สามารถหา L_1 ได้จาก Itemset จาก C_1 ที่ผ่านค่าสนับสนุนน้อยที่สุดแล้ว จากตัวอย่างค่าสนันสนุนแต่ละตัวใน Itemset ของ C_1 มีค่ามากกว่าหรือเท่ากับค่าสนับสนุนน้อยที่สุด (ในที่นี้ค่าสนับสนุนน้อยที่สุดมีค่าเป็น 2) ดังนั้นสมาชิกของ C_1 ทุกตัวจึงเป็นสมาชิกของ L_1

2. ขั้นตอนที่ 2 จาก L_1 เป็น C_2 เป็นการหา Candidate 2-Itemsets ซึ่งเกิดจากการเชื่อมความสัมพันธ์ของ L_1 และ L_1 จะได้เป็น C_2 เมื่อได้ Itemset ที่เกิดจากการเชื่อมความสัมพันธ์แล้ว ต้องคัดสมาชิกที่ได้ด้วยการหาเซตย่อยของแต่ละ Itemset ว่ามีอยู่ใน L_1 หรือไม่ เช่น {I1, I2} มีเซตย่อยคือ {I1}, {I2} เมื่อดู Item ใน L_1 แล้วมีทั้ง 2 Itemset ดังนั้นจึงเก็บ {I1, I2} นี้ไว้ และพิจารณาอย่างนี้ทุก Itemset

3. ขั้นตอนที่ 3 เป็นการหาความถี่หรือค่าสนับสนุนของ Itemset ใน C_2 ที่ปรากฏในฐานข้อมูล

4. ขั้นตอนที่ 4 จาก C_2 เป็น L_2 โดย L_2 เป็น Itemset จาก C_2 ที่ผ่านค่าสนับสนุนน้อยที่สุดแล้ว จากรูปที่ 2.5 พบว่ามีทั้งหมด 4 Itemset ที่มีค่าสนับสนุนน้อยกว่าค่าสนับสนุนน้อยที่สุด ได้แก่ {I1, I4}, {I3, I4}, {I3, I5} และ {I4, I5} ดังนั้นจึงตัด Itemset เหล่านี้ทิ้ง

5. ขั้นตอนที่ 5 จาก L_2 เป็น C_3 เป็นการหา Candidate 3-Itemsets ซึ่งเกิดจากการเชื่อมความสัมพันธ์ของ L_2 และ L_2 จะได้เป็น C_3 เมื่อได้ Itemset แล้วต้องคัดสมาชิกที่ได้ด้วยการหาเซตย่อยของแต่ละ Itemset ว่ามีอยู่ใน L_2 หรือไม่ จากรูปที่ 2.5 เมื่อเชื่อมความสัมพันธ์ของ L_2 และ L_2 จะได้ Itemset จำนวน 6 Itemset ดังนี้ {I1, I2, I3} , {I1, I2, I5}, {I1, I3, I5}, {I2, I3, I4}, {I2, I3, I5}, {I2, I4, I5} และนำแต่ละ Itemset มาหาเซตย่อยแล้วตรวจดูว่าเซตย่อยที่ได้ปรากฏอยู่ใน L_2 หรือไม่ เช่น {I1, I2, I3} หาเซตย่อยได้ {I1, I2}, {I1, I3} และ {I2, I3} เมื่อพบว่าเซตย่อยทั้งหมดปรากฏอยู่ใน L_2 จึงเก็บ Itemset นี้ไว้ และในกรณีที่ต้องตัด Itemset นั้น ๆ ทิ้ง เช่น {I1, I3, I5} หาเซตย่อยได้ {I1, I3}, {I1, I5} และ {I3, I5} เมื่อตรวจดูใน L_2 แล้วพบว่าไม่มี {I3, I5} ปรากฏอยู่ดังนั้นจึงตัด {I1, I3, I5} ออก ทำอย่างนี้ไปเรื่อย ๆ จนกว่าจะหมดทุก Itemset จะพบว่ามีจำนวน 2 Itemset เท่านั้นที่มีคุณสมบัติครบคือ {I1, I2, I3} และ {I1, I2, I5}

6. ขั้นตอนที่ 6 เป็นการหาความถี่หรือค่าสนับสนุนของ Itemset ใน C_3 ที่ปรากฏในฐานข้อมูล

7. ขั้นตอนที่ 7 จาก C_3 เป็น L_3 โดย L_3 เป็น Itemset จาก C_3 ที่ผ่านค่าสนับสนุนน้อยที่สุดแล้ว จากรูปที่ 2.4 พบว่าทุก Itemset มีค่าสนับสนุนเท่ากับค่าสนับสนุนน้อยที่สุดจึงเก็บ Itemset เหล่านั้นไว้

8. จาก L_3นำมาเชื่อมความสัมพันธ์เพื่อจะได้ C_4 จะได้ {I1, I2, I3, I5} และหาเซตย่อยของ Itemset นี้จะได้ {I1, I2, I3}, {I1, I2, I5}, {I1, I3, I5} และ {I2, I3, I5} เมื่อดูเซตย่อยนี้ใน L_3 ปรากฏว่าใน L_3 ไม่มี Itemset ที่เป็นเซตย่อยทั้งหมด ดังนั้นจึงไม่มี C_4 เกิดขึ้น และหยุดการหา Frequent Itemset

2.2.1.2 การสร้างกฎความสัมพันธ์

การสร้างกฎความสัมพันธ์จากFrequent Itemset จะนำ L_2 ถึง L_k มาสร้างกฎความสัมพันธ์ และในตัวอย่างจะนำ L_2 และ L_3 มาสร้างกฎความสัมพันธ์

วิธีการสร้างกฎความสัมพันธ์ ทำโดยนำ L_2 ถึง L_k (ในตัวอย่างคือ L_3) มาหาเซตย่อยแต่ละ Itemset และนำเซตย่อยที่ได้มาสร้างเป็นกฎความสัมพันธ์ จาก Frequent Itemset L ทำการหาเซตย่อยของ Frequent Itemset L และทุกเซตย่อย s ของ L ยกเว้นเซตว่างจะได้กฎ "s=>(L-s)" ดังแสดงรายละเอียดดังรูปที่ 2.5

จากรูปที่ 2.6 จาก L_2 ที่ Itemset แรก {I1, I2} นำมาหาเซตย่อยได้ {I1}, {I2} และ {I1, I2} เมื่อนำมาสร้างกฎจะได้ I1 => I2 , I2 =>1 แต่ไม่สามารถสร้าง I1 => I1^ I2 ได้ เพราะทางด้านซ้ายขวาจะมี I1 เหมือนกันไม่ได้ และสำหรับ Itemset อื่น ๆ ก็สร้างกฎด้วยวิธีนี้เช่นกัน ในทำนองเดียวกันนำ L_3 มาสร้างกฎความสัมพันธ์ ที่ Itemset แรก {I1, I2, I3} หาเซตย่อยได้ {I1}, {I2}, {I3}, {I1, I2}, {I1, I3}, {I2, I3} และ {I1, I2, I3} เมื่อนำมาสร้างกฎความสัมพันธ์จะได้ I1 => I2 ^ I3 , I2 => I1 ^ I3 , I3 => I1^I2 , I1^ I2 => I3 , I1 ^ I3 => I2 และ I2 ^ I3 => I1 เมื่อสร้างกฎความสัมพันธ์จาก Frequent Itemset แล้ว กฎความสัมพันธ์ที่ได้ยังไม่สามารถนำไปใช้งานได้ ต้องนำไปหาค่าความเชื่อมั่นก่อน กฎที่สามารถนำไปใช้งานได้ต้องผ่านค่าความเชื่อมั่นน้อยที่สุดก่อน จากตัวอย่างได้กำหนดให้ค่าความเชื่อมั่นน้อยที่สุดมีค่า 70% กฎความสัมพันธ์ที่สามารถนำไปใช้งานได้ต้องมีค่าความเชื่อมั่นของกฎมากกว่าหรือเท่ากับ 70% จากรูปที่ 2.4 จะได้กฎความสัมพันธ์ที่นำไปใช้งานได้ 6 กฎคือ I5 => I1, I4 => I2 , I5 => I2 , I5 => I1^ I2 , I1^ I5 => I2 และ I2 ^ I5 => I1

2.3 งานวิจัยที่เกี่ยวข้อง

ในงานวิจัยของ Osmar R. Zaiane และ Maria-Luiza Antonie [3, 4] ได้นำเสนอวิธีการจำแนกประเภทเอกสารโดยใช้กฎความสัมพันธ์ ซึ่งมีการพัฒนามาอย่างต่อเนื่อง เริ่มต้นด้วยการลดเวลาในการค้นหากฎความสัมพันธ์โดยการตัดข้อมูล (Documents) ที่ไม่ได้ใช้งานแล้วทิ้งเพื่อลดเวลาในการแสกนข้อมูล [3] แล้วนำกฎความสัมพันธ์ที่ได้ไปใช้ในการจำแนกประเภทเอกสาร ซึ่งสามารถจำแนกเอกสารได้ดีหากเอกสารนั้นเป็นเอกสารที่ไม่มีความสัมพันธ์กับกลุ่มใส ต่อมาได้พัฒนาเพิ่มเติม [4] โดยการคัดกฎความสัมพันธ์ทิ้ง (Pruning rule) ซึ่งเป็นการเพิ่มประสิทธิภาพใน

L_2

Itemset	Support
{I1, I2}	4
{I1, I3}	4
{I1, I5}	2
{I2, I3}	4
{I2, I4}	2
{I2, I5}	2

Association rule	Confidence
I1 => I2	66%
I2 => I1	57%
I1 => I3	66%
I3 => I1	66%
I1 => I5	33%
I5 => I1	100%
I2 => I3	57%
I3 => I2	66%
I2 => I4	29%
I4 => I2	100%
I2 => I5	29%
I5 => I2	100%

L_3

Itemset	Support
{I1, I2, I3}	2
{I1, I2, I5}	2

Association rule	Confidence
I1 => I2 ^ I3	33%
I2 => I1 ^ I3	29%
I3 => I1 ^ I2	33%
I1 ^ I2 => I3	50%
I1 ^ I3 => I2	50%
I2 ^ I3 => I1	50%
I1 => I2 ^ I5	33%
I2 => I1 ^ I5	29%
I5 => I1 ^ I2	100%
I1 ^ I2 => I5	50%
I1 ^ I5 => I2	100%
I2 ^ I5 => I1	100%

รูปที่ 2.5 แสดงการสร้างกฎความสัมพันธ์

การจำแนกประเภทเอกสาร แต่อย่างไรก็ตาม การจำแนกประเภทเอกสารก็ยังคงทำได้ไม่ดีหากเอกสารนั้นมีความสัมพันธ์กับข้อมูลกลุ่มอื่น

จากงานวิจัยที่เกี่ยวข้องนี้ได้นำเสนอวิธีการสร้างตัวจำแนกประเภทเอกสารออกเป็น 3 ส่วน คือ การค้นหากฎความสัมพันธ์ การคัดกฎความสัมพันธ์ทิ้ง และการจำแนกประเภทเอกสาร แสดงรายละเอียดของแต่ละส่วนดังต่อไปนี้

2.3.1 การค้นหากฎความสัมพันธ์ของงานวิจัยที่เกี่ยวข้อง

การค้นหากฎความสัมพันธ์ของงานวิจัยที่เกี่ยวข้องนี้ ได้นำเสนออัลกอริทึม Association Rule-based Classifier By Category (ARC-BC) เพื่อใช้ในการค้นหากฎความสัมพันธ์ แสดงรายละเอียดของอัลกอริทึมดังรูปที่ 2.6

จากรูปที่ 2.6 ในขั้นตอนที่ (2) เป็นการสร้าง Frequent 1- Itemset ขั้นตอนที่ (3) –(12) เป็นการสร้างFrequent k- Itemsets โดยมีการตัดข้อมูลที่ไม่ใช้แล้วทิ้งจากฐานข้อมูลในขั้นตอนที่ (6) และในขั้นตอนที่ (15) – (17) เป็นการสร้างกฎความสัมพันธ์จาก Frequent Itemset โดยเลือกเฉพาะกฎที่ทางด้านขวาของกฎบอกชื่อประเภทเอกสารเท่านั้น

เนื่องจากกฎความสัมพันธ์ที่ได้มีจำนวนมาก หากนำไปใช้จำแนกประเภทเอกสารจะทำให้ใช้เวลาในการจำแนกประเภทเอกสารมาก ดังนั้นจึงต้องคัดกฎความสัมพันธ์ทิ้งเป็นบางส่วน ดังจะกล่าวถึงในหัวข้อถัดไป

2.3.2 การ คัด กฎความสัมพันธ์ทิ้งของงานวิจัยที่เกี่ยวข้อง

จากการสร้างกฎความสัมพันธ์ของข้อมูลตามอัลกอริทึม ARC-BC ทำให้ได้กฎความสัมพันธ์จำนวนมาก จึงต้องนำกฎความสัมพันธ์ที่ได้มา คัดทิ้ง (Prune) เพื่อให้เหลือกฎที่ใช้ในการจำแนกเอกสารน้อยลง ซึ่งมีวิธีการคัดกฎทิ้งหลัก ๆ สองขั้นตอนคือ การเก็บเฉพาะกฎที่ สั้นที่สุด (General) และมีค่าความเชื่อมั่นมากที่สุด และการตัดกฎที่ไม่ได้ใช้ในฐานข้อมูล วิธีการคัดกฎความสัมพันธ์ทิ้ง แสดงตามนิยามที่ 1 และนิยามที่ 2 ต่อไปนี้

นิยามที่ 1 เป็นการหากฎที่สั้น มากที่สุด โดยให้กฎความสัมพันธ์ 2 กฎดังนี้ $T_1 \Rightarrow C$ และ $T_2 \Rightarrow C$ จะกล่าวได้ว่ากฎที่ 1 สั้นมากกว่ากฎที่ 2 ก็ต่อเมื่อ $T1 \subset T2$ ให้เก็บกฎที่สั้นมากกว่าไว้

นิยามที่ 2 เป็นการจัดลำดับกฎความสัมพันธ์ที่ได้จากนิยามที่ 1 โดยพิจารณาที่กฎ 2 กฎ คือ R1 และ R2 จะจัดลำดับ R1 ให้เป็นลำดับที่สูงกว่าก็ต่อเมื่อ

1. R1 มีค่าความเชื่อมั่นมากกว่า R2
2. ถ้าทั้ง 2 กฎมีค่าความเชื่อมั่นเท่ากัน ให้พิจารณาที่ค่าสนับสนุน R1 ต้องมีค่าสนับสนุนมากกว่า R2
3. ถ้าทั้ง 2 กฎมีค่าความเชื่อมั่นและค่าสนับสนุนเท่ากัน ให้พิจารณาที่กฎความสัมพันธ์โดยกฎทางด้านซ้ายของ R1 ต้องมีน้อยกว่า R2

Algorithm ARC-BC Find association rules on the training set of the text collection when the text corpora is divided in subsets by category

Input A set of documents (D) of the form D_i : $\{c_i, t_1, t_2, \ldots, t_n\}$ where c_i is the category attached to the document and t_j are the selected terms for the document: A minimum support threshold: A minimum confidence threshold.

Output A set of association rules of the form $t_1 \wedge t_2 \wedge \ldots \wedge t_n \Rightarrow c_i$ where c_i is the category and t_j is a term.

Method:

(1) $C_1 \leftarrow$ {Candidate 1 term-sets and their support}

(2) $F_1 \leftarrow$ {Frequent 1 term-set and their support}

(3) For ($i \leftarrow 2$; $F_{i-1} \neq \phi$; $i \leftarrow i+1$) do

(4) { $C_i \leftarrow (F_{i-1} \quad F_{i-1})$

(5) $C_i \leftarrow C_i - \{c \mid (i-1)$ item-set of $c \notin F_{i-1}\}$

(6) $D_i \leftarrow$ FilterTable(D_{i-1},F_{i-1})

(7) for each document d in D_i do

(8) { for each c in C_i do

(9) { c.support $\leftarrow$ c.support + count(c,d) }

(10) }

(11) $F_i \leftarrow \{c \ C_i \mid$ c.support $> \sigma$ }

(12) }

(13) Sets $\leftarrow U_i\{c \in F_i \mid i > 1\}$

(14) $R = \phi$

(15) For each itemset I in sets do {

(16) R $\leftarrow$ R+{I =>Cat}

(17) }

end

รูปที่ 2.6 อัลกอริทึม ARC-BC [2]

หลังจากหากฎที่สั้นมากที่สุดได้แล้ว กฎเหล่านี้จะเป็นกฎที่ใช้สำหรับจำแนกประเภทเอกสาร และต่อไปจะกล่าวถึงวิธีการจำแนกประเภทเอกสาร

2.3.3 วิธีการจำแนกประเภทเอกสารของงานวิจัยที่เกี่ยวข้อง

ในการจำแนกเอกสารของงานวิจัยนี้ ได้กำหนดค่า Dominant factor (δ) เพื่อใช้ในการจำแนกเอกสาร ในกรณีที่กฎที่ใช้ในการจำแนกบอกว่าเอกสารอยู่มากกว่าสองกลุ่ม โดย Dominant factor จะเป็นตัวระบุว่าควรจะใช้กฎใดในการจำแนกเอกสารนี้ ซึ่งค่า Dominant factor นี้จะอยู่ระหว่าง 0-1 ในงานวิจัยที่เกี่ยวข้องนี้แนะนำว่าหากค่า Dominant factor มีค่ามาก ก็คือเข้าใกล้ 1 ผลการจำแนกประเภทเอกสารก็จะดีมากขึ้นกว่าค่า Dominant factor ที่มีค่าเข้าใกล้ 0 รายละเอียดในการจำแนกเอกสารของงานวิจัยที่เกี่ยวข้องนี้ แสดงดังรูปที่ 2.7

Algorithm Classification of new object

Input A new object to be classified o; The associative classifier (ARC); The dominant factor δ ; The confidence threshold T;

(1) S $\leftarrow \phi$ /* set of rules that match new document (o) */

(2) foreach rule r in ARC (the sorted set of rules)

(3) if (r $\subset$ o) {count++}

(4) if (count == 1)

(5) fr.conf` r.conf /*keep the first rule confidence */

(6) S $\leftarrow$ S $\cup$ r

(7) else if (r.conf > fr.conf-T)

(8) S $\leftarrow$ S $\cup$ r

(9) else exit

(10) divide S in subsets by category : $S_1, S_2, \ldots, S_n$

(11) foreach subset $S_1, S_2, \ldots, S_n$

(12) sum the confidences of rules and divide by the number of rules in Sk

(13) if it is single class classification

(14) put the new document in the class that has the highest confidence sum

(15) else /* multi-class classification */

(16) TakeKClasses(S, δ)

(17) assign these k classes to the new document

รูปที่ 2.7 อัลกอริทึมการจำแนกประเภทเอกสารของงานวิจัยที่เกี่ยวข้อง [2]

จากรูปที่ 2.7 ขั้นตอนที่ (3) เป็นการหากฎที่มีความสัมพันธ์กับเอกสารใหม่ที่ต้องการจำแนกประเภท ขั้นตอนที่ (4) - (9) เป็นการจัดลำดับกฎความสัมพันธ์ตามความสัมพันธ์กับ

เอกสารใหม่ ขั้นตอนที่ (10) – (12) เป็นการหาค่าความเชื่อมั่นของกลุ่มเอกสาร สำหรับการจำแนกประเภทเอกสาร ขั้นตอน (13)-(14) เป็นการจำแนกประเภทเอกสารที่เป็นกลุ่มเดียว และขั้นตอนที่ (15) – (17) เป็นการจำแนกประเภทเอกสารที่เป็นแบบหลายกลุ่มโดยใช้ค่า Dominant factor ช่วยในการจำแนกประเภทเอกสาร

ตัวจำแนกประเภทเอกสารของงานวิจัยนี้ สามารถจำแนกเอกสารได้ดีในกรณีที่ลักษณะข้อมูลของเอกสารนั้นไม่มีการซ้อนทับกัน แต่หากมีข้อมูลที่มีลักษณะเอกสารซ้อนทับกันจะทำให้การจำแนกเอกสารได้ไม่ดีนัก ดังนั้นวิทยานิพนธ์ฉบับนี้จึงได้คิดวิธีการที่จะเพิ่มประสิทธิภาพการจำแนกประเภทเอกสารให้ดีกว่าเดิม โดยจะคิดค้นวิธีการสร้างตัวจำแนกประเภทเอกสาร (Text classifier) ตั้งแต่วิธีการค้นหากฎความสัมพันธ์ การคัดกฎความสัมพันธ์ทิ้ง และวิธีการจำแนกประเภทเอกสาร ดังจะกล่าวรายละเอียดต่าง ๆ ในบทต่อไป

บทที่ 3

อัลกอริทึมใหม่สำหรับการจำแนกเอกสาร โดยใช้กฎความสัมพันธ์

ในบทนี้จะกล่าวถึงอัลกอริทึมใหม่สำหรับจำแนกเอกสารโดยใช้กฎความสัมพันธ์ ซึ่งแบ่งออกเป็น 3 ส่วน คือ การค้นหากฎความสัมพันธ์ การคัดกฎความสัมพันธ์ทิ้ง และการจำแนกเอกสารใหม่ ดังจะกล่าวต่อไปนี้

3.1 การค้นหากฎความสัมพันธ์

ในการสร้างกฎความสัมพันธ์ของอัลกอริทึม Apriori นั้นจะประกอบด้วยเซตข้อมูล 2 เซต คือ Candidate Itemset (C) และ Frequent Itemset (F) หรือ Large Itemset (L) แต่สำหรับงานวิจัยนี้มีเซตข้อมูลมากกว่าที่กล่าวมา คือ T, OL และ M นั่นคือจะมีเซตข้อมูลทั้งหมด 5 เซตข้อมูล ได้แก่ Candidate Itemset, Large Itemset, Temp Itemset, Overlap Itemset และ Main Itemset แต่ละเซตข้อมูลอธิบายได้ดังนี้

1. Candidate Itemset คือกลุ่มข้อมูลที่เป็นตัวเลือกสำหรับการสร้าง Frequent Itemset แทนด้วยสัญลักษณ์ C
2. Frequent Itemset หรือ Large Itemset คือกลุ่มข้อมูลที่ใช้สำหรับการสร้างกฎความสัมพันธ์ โดยแต่ละ Item ที่เป็นสมาชิกของเซตนี้ต้องผ่านค่าสนับสนุนน้อยที่สุดแล้ว แทนด้วยสัญลักษณ์ F หรือ L สำหรับงานวิจัยนี้ใช้ L
3. Temp Itemset คือกลุ่มข้อมูลที่เกิดจากการเชื่อมความสัมพันธ์ (Join) ระหว่าง L_0 และ L_1 โดย L_0 เป็น Itemset ที่ประกอบด้วยชื่อประเภทเอกสารที่ผ่านค่าสนับสนุนที่น้อยที่สุดแล้ว และ L_1 ประกอบด้วย Itemset ที่เป็นคุณลักษณะของเอกสารที่ผ่านค่าสนับสนุนที่น้อยที่สุดแล้ว แทนด้วยสัญญลักษณ์ T
4. Overlap Itemset คือ Itemset ทั้งหมดจากเซต T ที่มีคุณลักษณะของเอกสารซ้ำกัน (Overlap) แทนด้วยสัญญลักษณ์ OL
5. Main Itemset คือ Itemset ทั้งหมดใน L_2 ที่มีชื่อกลุ่มเอกสารเหมือนกับ Itemset ใน OL_2 แทนด้วยสัญญลักษณ์ M

3.1.1. อัลกอริทึมสำหรับค้นหากฎความสัมพันธ์

อัลกอริทึม Association Rule-based Text Classifier (ARTC) เป็นอัลกอริทึมที่ใช้ในการค้นหากฎความสัมพันธ์ของงานวิจัยนี้ แสดงดังรูปที่ 3.1

Algorithm: ARTC Find association rule on the training set of the text collection.

Input: 1. A set of documents of the form $\{c_i,t_1,t_2,\ldots,t_n\}$ where c_i is the category attached to the document and t_j are the selected terms for the document.

2. A minimum support threshold (min_support).

Output: A set of association rules of the form $t_1 \wedge t_2 \wedge \ldots \wedge t_n \Rightarrow c_i$ where c_i is the category and t_j is a term.

Method:

(1) $C_0 \leftarrow$ {Category label 1-itemset and their support}

(2) $L_0 \leftarrow \{c \in C_0 \mid \text{support} > \text{min\_support}\}$

(3) $C_1 \leftarrow$ {Feature 1-itemset and their support}

(4) $L_1 \leftarrow \{c \in C_1 \mid \text{support} > \text{min\_support}\}$

(5) $C_2 \leftarrow$ Gen_2_Itemsets (L_0 , L_1)

(6) $T \leftarrow \{c \in C_2 \mid \text{support} > \text{min\_support}\}$

(7) $OL_2 \leftarrow$ Find_Overlap_Itemset (T)

(8) $L_2 \leftarrow$ T-OL_2

(9) $M \leftarrow$ Select_Itemset(OL_2,L_2)

(10) For ($k=3$; $OL_k \neq \phi$; k++)

(11) {$C_k \leftarrow$ Gen_k_Itemsets (k, $OL_{(k-1)}$, M)

(12) $L_k \leftarrow \{c \in C_k \mid \text{support} > \text{min\_support}\}$

(13) $COL_k \leftarrow$ Gen_Overlap_k_Itemsets (k, $OL_{(k-1)}$)

(14) OLk $\leftarrow \{c \in COL_k \mid \text{support} > \text{min\_support}\}$ }

(15) SetOfRule $\leftarrow \{L_2, L_3, \ldots L_k, OL_2, OL_3, \ldots, OL_k\}$

(16)Rules = ϕ

(17)For each item in SetOfRule do

(18) {Rules $\leftarrow t_1 \wedge t_2 \wedge \ldots t_{k-1} \Rightarrow C_i \mid$ t is term of document and C is category label, $i \in 1,2,\ldots,p$ }

(19)end

รูปที่ 3.1 อัลกอริทึม ARTC

จากรูปที่ 3.1 แสดงอัลกอริทึมสำหรับค้นหากฎความสัมพันธ์ของงานวิจัยนี้ (ARTC Algorithm) เริ่มต้นจากข้อมูลนำเข้าดังนี้

1. กลุ่มเอกสารสำหรับฝึกหัดระบบที่อยู่ในรูปแบบ $\{c_i, t_1, t_2,\ldots, t_n\}$ โดยที่ c_i เป็นชื่อประเภทเอกสาร และ t เป็นคุณลักษณะของเอกสาร มีทั้งหมด n ตัว
2. ค่าสนับสนุนน้อยที่สุด (Minimum support threshold :min_support)

ผลลัพธ์ (Output) ที่ได้ของอัลกอริทึมนี้คือกฎความสัมพันธ์ที่อยู่ในรูปของ $t_1 \wedge t_2 \wedge \ldots \wedge t_n \Rightarrow C_i$ โดยที่ t เป็นคุณลักษณะของเอกสาร และ C เป็นชื่อประเภทเอกสาร

เพื่อความง่ายต่อการเข้าใจวิธีการทำงานของอัลกอริทึมสำหรับค้นหากฎความสัมพันธ์ของงานวิจัยนี้ จะแสดงวิธีการค้นหากฎความสัมพันธ์ในแต่ละขั้นตอนพร้อมยกตัวอย่าง โดยกำหนดให้ฐานข้อมูล Doc เป็นกลุ่มเอกสารสำหรับค้นหากฎความสัมพันธ์ แสดงดังตารางที่ 3.1 โดยที่

Category label = {Category1, Category2, Category3}

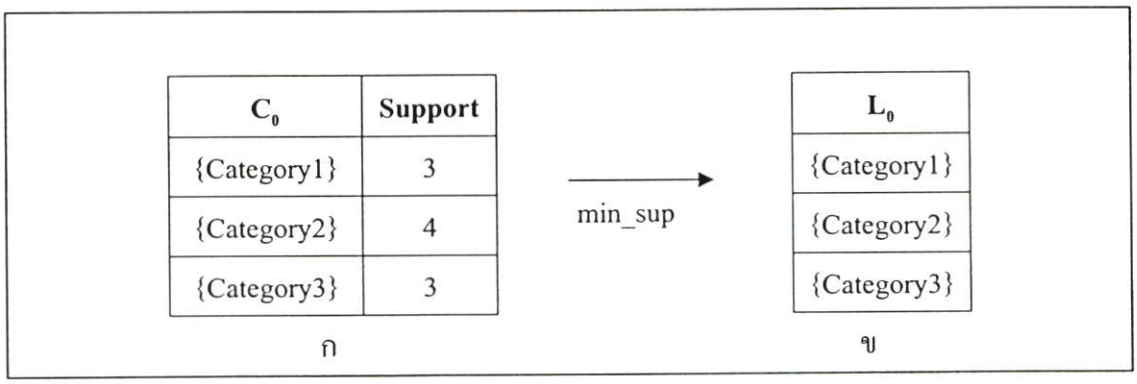
Term = {protocol, algorithm, …, database}

ตารางที่ 3.1 รายการฐานข้อมูล Doc

ลำดับ	รายการข้อมูล
1	{Category2,protocol,algorithm,xml,mail,web,firewall }
2	{Category3,database}
3	{Category2,ftp,algorithm}
4	{Category1,management,money,market}
5	{Category2,protocol,mail,xml,algorithm,web,firewall}
6	{Category1,tour}
7	{Category2,web,algorithm,ftp,firewall}
8	{Category1,money,management}
9	{Category3,xml,oracle,web,protocol,database}
10	{Category3,web,xml,normalization,database}

กำหนดให้ค่าสนับสนุนน้อยที่สุดมีค่าเท่ากับ 2 ทำการค้นหากฎความสัมพันธ์ตามอัลกอริทึมดังนี้

1. ขั้นตอนที่ (1) เป็นการหา C_0 ซึ่งก็คือชื่อกลุ่มเอกสาร พร้อมทั้งค่าสนับสนุนของแต่ละ Itemset แสดงดังรูปที่ 3.2 ก
2. ขั้นตอนที่ (2) เป็นการหาค่า L_0 ซึ่งเป็นสมาชิกของ C_0 ที่ผ่านค่าสนับสนุนน้อยที่สุดแล้ว แสดงดังรูปที่ 3.2 ข



ก ข

รูปที่ 3.2 แสดง C_0 และ L_0

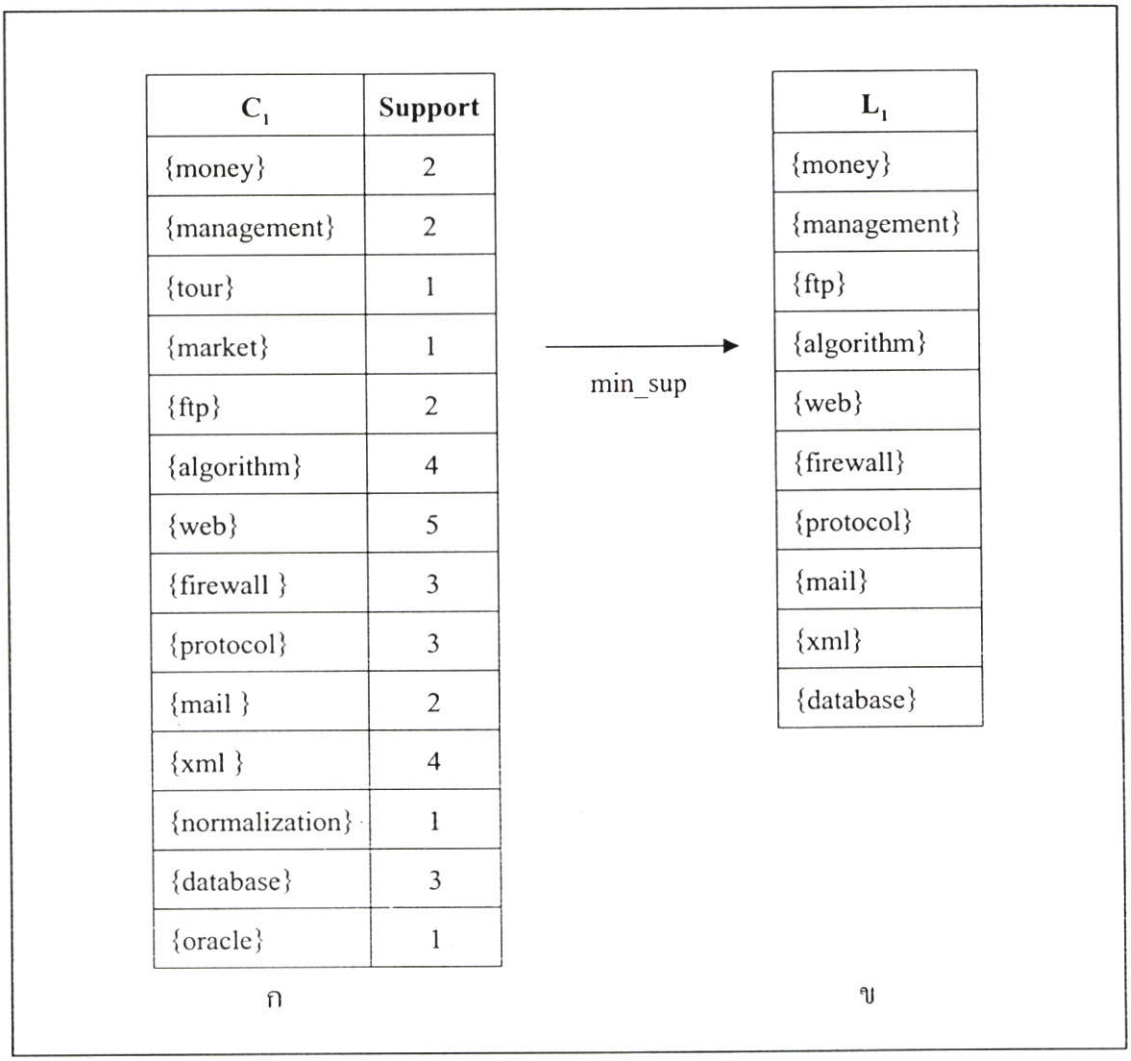
จากรูปที่ 3.2 แสดง C_0 และ L_0 จากฐานข้อมูล Doc หาสมาชิกของ C_0 โดยต้องเป็นชื่อกลุ่มเอกสาร พร้อมทั้งค่าสนับสนุนแต่ละ Itemset และสมาชิกของ L_0 คือสมาชิกของ C_0 ที่มีค่าสนับสนุนมากกว่าค่าสนับสนุนน้อยที่สุด จากตัวอย่างทุก Itemset ของ C_0 มีค่าสนับสนุนมากกว่าค่าสนับสนุนน้อยที่สุด ดังนั้นทุก Itemset ใน C_0 จึงเป็นสมาชิกของ L_0

3. ขั้นตอนที่ (3) เป็นการหา C_1 ซึ่งก็คือคุณลักษณะของเอกสาร พร้อมทั้งค่าสนับสนุนของแต่ละ Itemset แสดงดังรูปที่ 3.3 ก
4. ขั้นตอนที่ (4) เป็นการหาค่า L_1 ซึ่งเป็นสมาชิกของ C_1 ที่ผ่านค่าสนับสนุนน้อยที่สุดแล้ว แสดงดังรูปที่ 3.3 ข

จากรูปที่ 3.3 แสดง C_1 และ L_1 จากฐานข้อมูล Doc หาสมาชิกของ C_1 ซึ่งเป็นคุณลักษณะของเอกสาร พร้อมทั้งค่าสนับสนุนแต่ละ Itemset และสมาชิกของ L_1 คือสมาชิกของ C_1 ที่มีค่าสนับสนุนมากกว่าค่าสนับสนุนน้อยที่สุด จากตัวอย่างตามรูปที่ 3.3 {tour}, {market}, {normalization} และ {oracle} ของ C_1 มีค่าสนับสนุนน้อยกว่าค่าสนับสนุนน้อยที่สุด ดังนั้น Itemset ดังกล่าวจึงไม่เป็นสมาชิกของ L_1

5. ขั้นตอนที่ (5) เป็นการสร้าง Candidate 2-itemsets (C_2) โดยการเรียกใช้ฟังก์ชัน Gen_2_Itemsets การทำงานของฟังก์ชัน Gen_2_Itemsets แสดงดังรูปที่ 3.4

จากรูปที่ 3.4 แสดงวิธีการทำงานของฟังก์ชัน Gen_2_Itemsets ซึ่งเป็นฟังก์ชันในการสร้าง C_2 มีข้อมูลนำเข้าของฟังก์ชันนี้คือ L_0 และ L_1 เชื่อมความสัมพันธ์ (Join) ระหว่าง L_0 และ L_1 โดยทำการเชื่อมความสัมพันธ์ทุก ๆ Itemset ทั้ง L_0 และ L_1 แสดงตัวอย่างการสร้าง C_2 ดังรูปที่ 3.5



C_1	Support
{money}	2
{management}	2
{tour}	1
{market}	1
{ftp}	2
{algorithm}	4
{web}	5
{firewall }	3
{protocol}	3
{mail }	2
{xml }	4
{normalization}	1
{database}	3
{oracle}	1

ก

min_sup

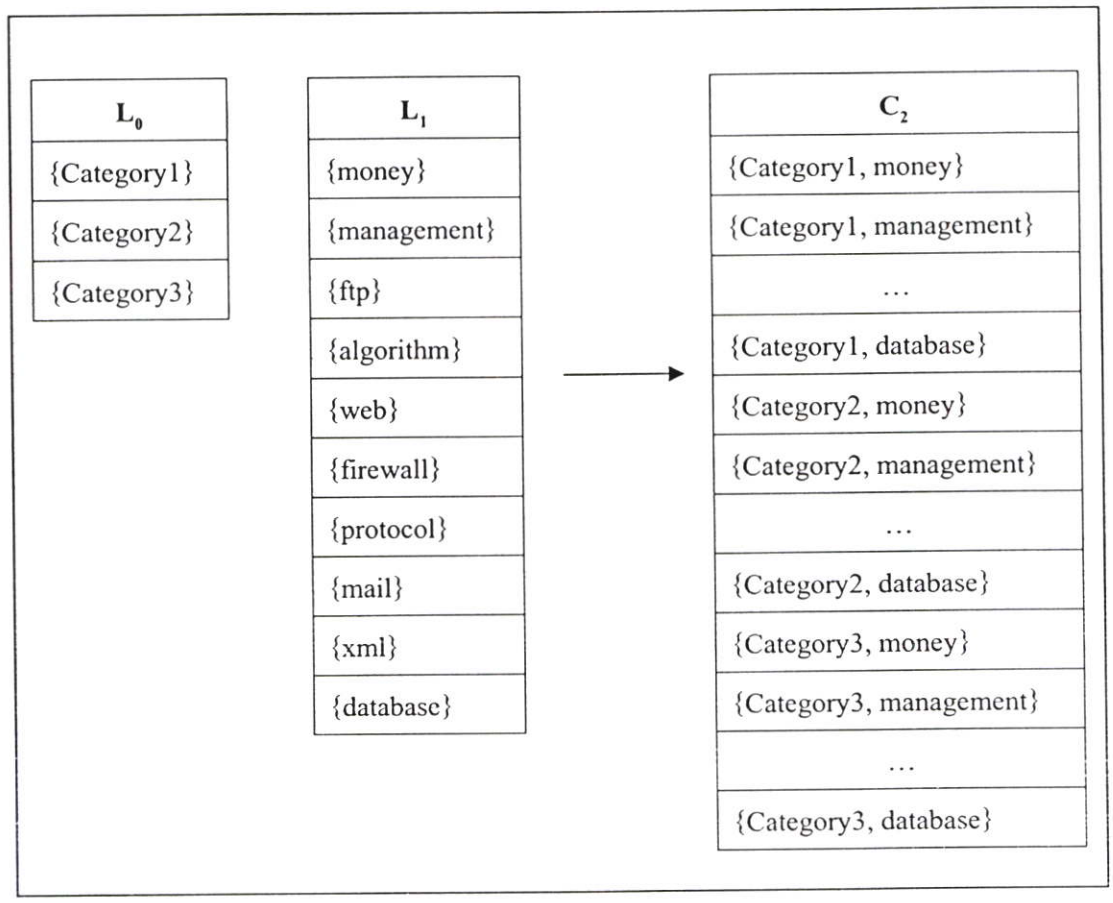
L_1
{money}
{management}
{ftp}
{algorithm}
{web}
{firewall}
{protocol}
{mail}
{xml}
{database}

ข

รูปที่ 3.3 แสดง C_1 และ L_1

```
%Function Gen_2_Itemsets (L0 , L1)
(5.1) for (i=1 ; L0 ≠ φ ;i++)
(5.2)     {for (j=1;L1 φ ;j++)
(5.3)             {Insert into C2
(5.4)             Select L0[i],L1[j]
(5.5)             From L0,L1
(5.6)             }
(5.7)     }
(5.8) Return C2
```

รูปที่ 3.4 ฟังก์ชัน Gen_2_Itemsets



รูปที่ 3.5 การสร้าง C_2

จากรูปที่ 3.5 แสดงตัวอย่างการสร้าง C_2 จากการเชื่อมความสัมพันธ์ระหว่าง Itemset ทุก ๆ Itemset ของ L_0 และ L_1 จากรูปที่ 3.5 {Category1} เชื่อมความสัมพันธ์กับ Itemset ทุก Itemet ใน L_1 จะได้ {Category1, money}, {Category1, management} จนถึง {Category1, database} เมื่อ Itemset ตัวแรกของ L_0 เชื่อมความสัมพันธ์กับ L_1 หมดทุก Itemset แล้ว หา L_0 ยังมี Itemset อื่นที่ยังไม่ได้เชื่อมความสัมพันธ์กับ L_1 ให้เชื่อมความสัมพันธ์จนกว่าทุก ๆ Itemset ของ L_0 จะเชื่อมความสัมพันธ์กับ Itemset ของ L_1 หมดทุกตัว จะได้ C_2 ดังรูปที่ 3.5

6. ขั้นตอนที่ (6) เป็นการสร้างเซต Temp (T) สมาชิกของ T คือสมาชิกของ C_2 ที่ผ่านค่าสนับสนุนน้อยที่สุดแล้ว แสดงดังรูป ที่ 3.6

จากรูปที่ 3.6 แสดงการเกิดของเซต T โดสมาชิกของเซต T ต้องเป็น Itemset ของ C_2 ที่มีค่าสนับสนุนมากกว่าหรือเท่ากับค่าสนับสนุนน้อยที่สุด เช่น {Category1, money} มีค่าสนับสนุนเท่ากับค่าสนับสนุนน้อยที่สุด ดังนี้ Itemset นี้จึงเป็นสมาชิกของเซต T และ {Category1, database} มีค่าสนับสนุนน้อยกว่าค่าสนับสนุนน้อยที่สุด ดังนั้น Itemset นี้จึงไม่เป็นสมาชิกของเซต T

C_2	Support
{Category1, money}	2
{Category1, management}	2
…	…
{Category1, database}	0
{Category2, money}	0
{Category2, management}	0
…	…
{Category2, database}	0
{Category3, money}	0
{Category3, management}	0
…	…
{Category3, database}	3

min_sup

T
{Category1, money}
{Category1, management}
{Category2, ftp}
{Category2, algorithm}
{Category2, web}
{Category2, firewall}
{Category2, protocol}
{Category2, mail}
{Category2, xml}
{Category3, web}
{Category3, xml}
{Category3, database}

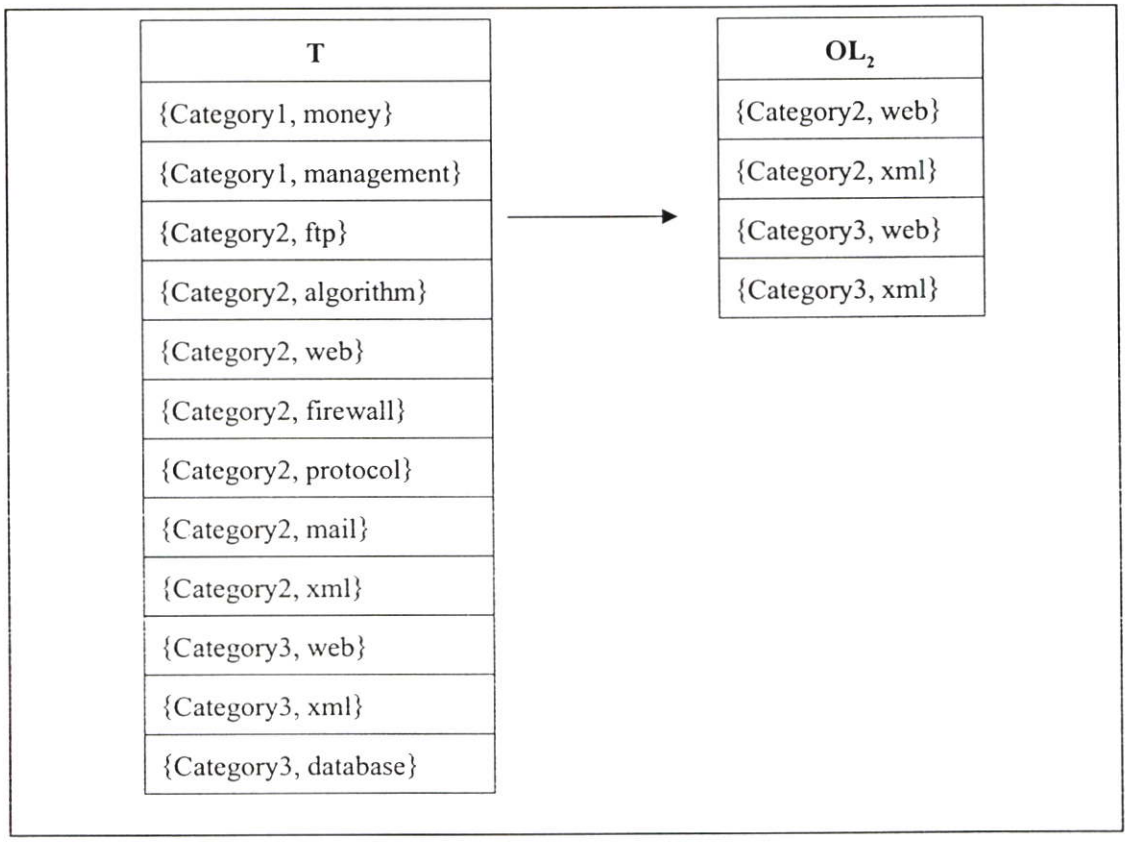
รูปที่ 3.6 การสร้าง T

7. ขั้นตอนที่ (7) เป็นการสร้าง Overlap 2-itemset (OL_2) โดยเรียกใช้ฟังก์ชัน Find_Overlap_Itemset การทำงานของฟังก์ชัน Find_Overlap_Itemset แสดงดังรูปที่ 3.7

```
%Function Find_Overlap_Itemset (T)
(7.1) for(i=1;T[i] ≠ ϕ ;I++)
(7.2)     {for (j=2;T[j] ≠ ϕ ; j++)
(7.3)             {Insert into OL2
(7.4)             Select T[i]
(7.5)             From T
(7.6)             Where T[i,2] = T[j,2]
(7.7)             }
(7.8)     }
(7.9) Return OL2
```

รูปที่ 3.7 ฟังก์ชัน Find_Overlap_Itemset

จากรูปที่ 3.7 แสดงวิธีการทำงานของฟังก์ชัน Find_Overlap_Itemset ซึ่งเป็นฟังก์ชันในการหา term ที่ซ้ำกันของเอกสารแต่ละกลุ่ม โดยพิจารณาที่ Item ตัวที่ 2 ของแต่ละ Itemset (ซึ่งก็คือ term ของเอกสารนั้น ๆ) ว่าเหมือนกันหรือไม่ หากเหมือนกัน Itemset นั้น ๆ จะเป็นสมาชิกของ OL_2 แสดงตัวอย่างดังรูปที่ 3.8



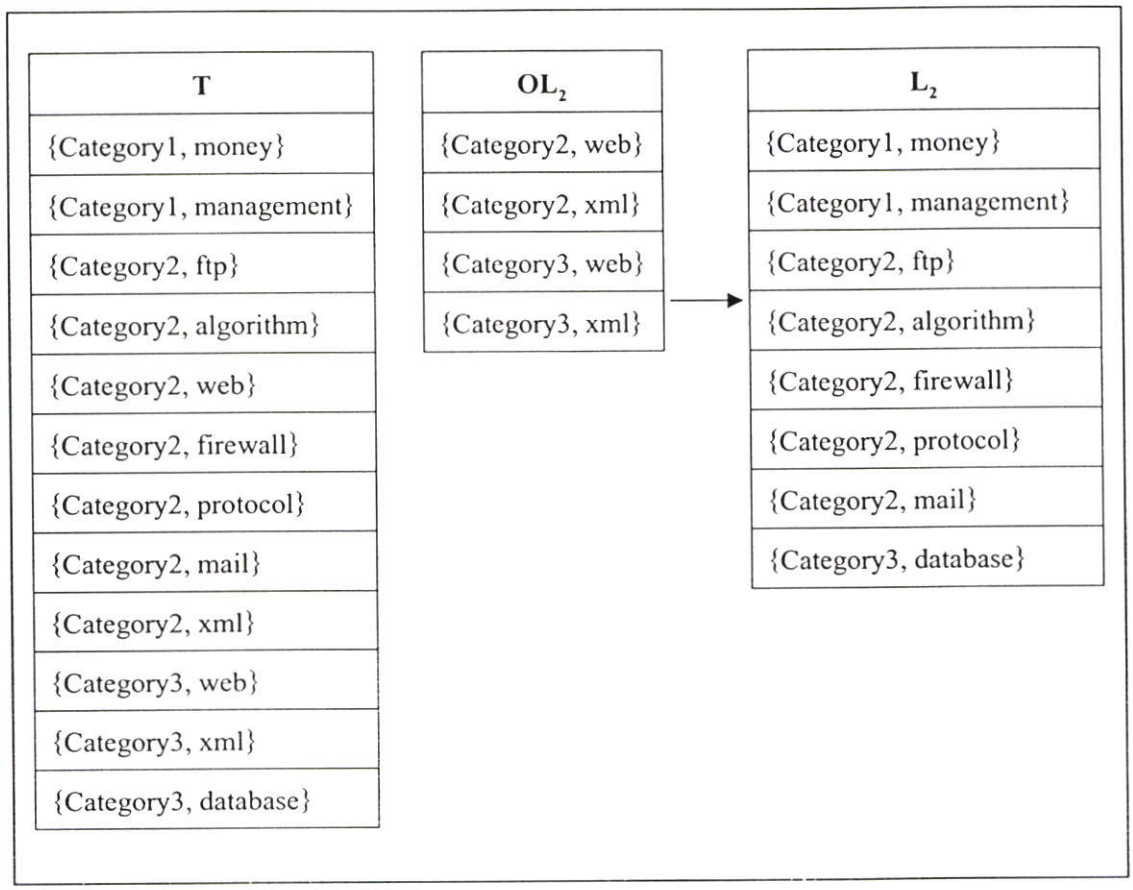
T
{Category1, money}
{Category1, management}
{Category2, ftp}
{Category2, algorithm}
{Category2, web}
{Category2, firewall}
{Category2, protocol}
{Category2, mail}
{Category2, xml}
{Category3, web}
{Category3, xml}
{Category3, database}

OL_2
{Category2, web}
{Category2, xml}
{Category3, web}
{Category3, xml}

รูปที่ 3.8 แสดงการเกิด OL_2

จากรูปที่ 3.8 ตามการทำงานของฟังก์ชัน Find_Overlap_Itemset พิจารณาที่ Item ตัวที่ 2 ว่าเหมือนกันหรือไม่ เริ่มจาก Itemset ตัวที่ 1 คือ {Category1, money} มี Item ตัวที่ 2 คือ money ให้พิจารณากับทุก ๆ Itemset ปรากฏว่าไม่มี Itemset ตัวไหนที่มี Item ตัวที่ 2 เป็น money ดังนั้น Itemset นี้จึงไม่เป็นสมาชิกของ OL_2 พิจารณาที่ Itemset {Category2, web} ที่ Item ตัวที่ 2 คือ web เมื่อพิจารณาแล้วพบว่าซ้ำกับ {Category3, web} ดังนั้น ทั้ง {Category2, web} และ {Category3, web} จึงเป็นสมาชิกของ OL_2

8. ขั้นตอนที่ (8) เป็นการสร้าง Large 2-itemset (L_2) โดยที่สมาชิกของ L_2 คือสมาชิกของเซต T ที่ไม่เป็นสมาชิกของ OL_2 แสดงตัวอย่างดังรูปที่ 3.9



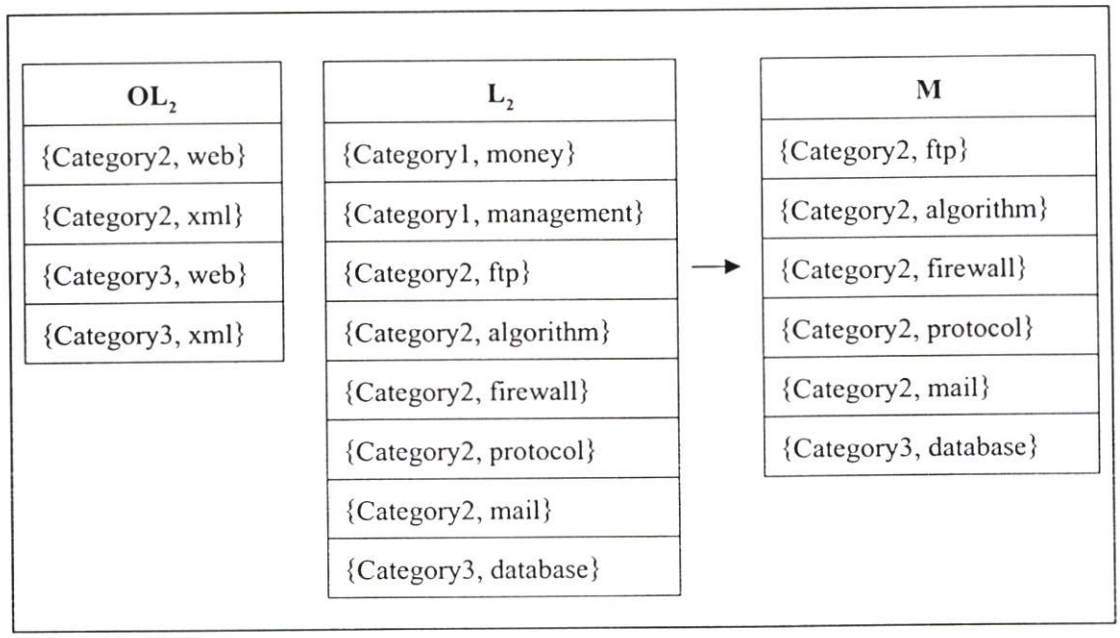
รูปที่ 3.9 แสดงตัวอย่าง L_2

9. ขั้นตอนที่ (9) เป็นการสร้าง Main Itemset (M) โดยการเรียกใช้ฟังก์ชัน Select_Itemset การทำงานของฟังก์ชัน Select_Itemset แสดงดังรูปที่ 3.10

```
%Function Select_Itemset (OL2 , L2)
(9.1) Temp_OL2 ← {Category label from OL2}
(9.2)    for(i=1;Temp_OL2[i] ≠ ϕ ;i++)
(9.3)          {for(j=1;L2[j] ≠ ϕ ;j++)
(9.4)                {Insert into M
(9.5)                Select L2[j]
(9.6)                From L2
(9.7)                Where Temp_OL2[i] = L2[j,1]
(9.8)                }
(9.9)          }
(9.10) Return ML
```

รูปที่ 3.10 ฟังก์ชัน Select_Itemset

จากรูปที่ 3.10 แสดงการทำงานของฟังก์ชัน Select_Itemset ซึ่งเป็นฟังก์ชันในการหา Main Itemset (M) โดยสมาชิกของเซต M คือสมาชิกใน L_2 ที่มีชื่อกลุ่ม (Category label) เหมือนกับ OL_2 แสดงตัวอย่างดังรูปที่ 3.11



รูปที่ 3.11 ตัวอย่าง M

จากรูปที่ 3.11 แสดงตัวอย่างการสร้างเซต M ตามฟังก์ชัน Select_Itemset โดยพิจารณาที่ชื่อกลุ่มจาก OL_2 จากตัวอย่างคือ Category2 และ Category3 แล้วพิจารณาที่ L_2 หาก Itemset ใดใน L_2 ที่มีชื่อกลุ่มเป็น Category2 หรือ Category3 ให้เลือก Itemset นั้นเป็นสมาชิกของเซต M

10. ต่อไปจะเริ่มสู่ขั้นตอนวนซ้ำ (Iteration step) เพื่อหา Large Itemset ตั้งแต่ขั้นตอนที่ (10) – (14) โดยขั้นตอนที่ (10) เป็นการกำหนดค่า k เริ่มต้นและกำหนดเงื่อนไขในการหยุดการหา Large Itemset โดยให้ค่า k เริ่มต้นมีค่าเท่ากับ 3 การทำงานแต่ละรอบจะหยุดการทำงานก็ต่อเมื่อ OL_k เป็นเซ็ตว่าง และจะเพิ่มค่า k อีก 1 หากการทำงานยังไม่หยุดวนซ้ำ
11. ขั้นตอนที่ (11) เป็นการสร้าง Candidate k-itemsets (C_k) โดยเรียกใช้ฟังก์ชัน Gen_k_itemsets การทำงานของฟังก์ชัน Gen_k_itemsets แสดงดังรูปที่ 3.12

```
% Function Gen_k_Itemsets (k, OL(k-1) , M)
(11.1) for (i=1; OLk[i] ≠ φ ; i++)
(11.2)   {for (j=1:M[j] ≠ φ ; j++)
(11.3)          {Insert into Ck
(11.4)           Select OLk[i], M[j][2]
(11.5)           Form OLk, M
(11.6)           Where OLk[1] = M[1]
(11.7)          }
(11.8)   }
(11.9) Retune Ck
```

รูปที่ 3.12 ฟังก์ชัน Gen_k_Itemsets

จากรูปที่ 3.12 แสดงการทำงานของฟังก์ชัน Gen_k_Itemsets เป็นฟังก์ชันในการหา Candidate k-itemsets (C_k) ซึ่งเกิดจากการเชื่อมความสัมพันธ์ของ $OL_{(k-1)}$ และ M ในส่วนนี้เราได้นำเสนอวิธีการเชื่อมความสัมพันธ์แบบใหม่ เรียกว่า ARTC join โดยมีหลักการดังนี้ พิจารณาที่ Item ตัวแรกของแต่ละ Itemset ที่จะเชื่อมความสัมพันธ์กัน หากเหมือนกัน Itemset นั้น ๆ ก็สามารถเชื่อมความสัมพันธ์กันได้ แต่ถ้าไม่เหมือนกันก็จะไม่สามารถเชื่อมความสัมพันธ์กันได้ แสดงตัวอย่างการสร้าง C_k ดังรูปที่ 3.13

OL_2
{Category2, web}
{Category2, xml}
{Category3, web}
{Category3, xml}

M
{Category2, ftp}
{Category2, algorithm}
{Category2, firewall}
{Category2, protocol}
{Category2, mail}
{Category3, database}

→

C_3
{Category2, web, ftp}
{Category2, web, algorithm}
…
{Category2, web, mail}
{Category2, xml, ftp}
{Category2, xml, algorithm}
…
{Category2, xml, mail}
{Category3, web, database}
{Category3, xml, database}

รูปที่ 3.13 แสดงตัวอย่างการสร้าง C_k โดย ARTC join

รูปที่ 3.13 แสดงตัวอย่างการสร้าง C_k ในที่นี้ k มีค่าเท่ากับ 3 ดังนั้นจึงเป็นการหา C_3 ซึ่งเกิดจากการเชื่อมความสัมพันธ์ระหว่าง $OL_{(k-1)}$ ซึ่งก็คือ OL_2 และ M ทำการเชื่อมความสัมพันธ์ระหว่างทุก ๆ Itemset ของ OL_2 ทุก Itemset กับ M ทุก Itemset โดยมีวิธีพิจารณาว่า Itemset คู่ใดเชื่อมความสัมพันธ์กันได้หรือไม่นั้นให้พิจารณาที่ Item ตัวแรกว่าเหมือนกันหรือไม่ หากเหมือนกันจึงเชื่อมความสัมพันธ์กันได้ แต่ถ้าไม่เหมือนกันก็จะเชื่อมความสัมพันธ์กันไม่ได้ เช่น Itemset ตัวแรกของ OL_2 {Category2, web} และ Itemset ตัวแรกของ M {Category2, ftp} พิจารณาที่ Item ตัวแรกของแต่ละ Itemset คือ Category2 และ Category2 เหมือนกัน ดังนั้นทั้งสอง Itemset นี้เชื่อมความสัมพันธ์กันได้ จะได้เป็น {Category2, web, ftp} และในทำนองเดียวกันพิจารณาที่ {Category2, web} และ {Category3, database} เนื่องจาก Category2 และ Category3 ไม่เหมือนกันดังนั้น 2 Itemset นี้ไม่สามารถเชื่อมความสัมพันธ์กันได้

12. ขั้นตอนที่ (12) เป็นการหา Large k-itemset (L_k) ในตัวอย่างนี้ก็คือ L_3 โดยสมาชิกของ L_3 คือสามาชิกของ C_3 ที่มีค่าสนับสนุนมากกว่าหรือเท่ากับค่าสนับสนุนน้อยที่สุด แสดงตัวอย่างดังรูปที่ 3.14

C_3	Support
{Category2, web, ftp}	1
{Category2, web, algorithm}	3
…	…
{Category2, web, mail}	2
{Category2, xml, ftp}	0
{Category2, xml, algorithm}	2
…	…
{Category2, xml, mail}	2
{Category3, web, database}	2
{Category3, xml, database}	2

min_sup

L_3
{Category2, web, algorithm}
{Category2, web, firewall}
{Category2, web, protocol}
{Category2, web, mail}
{Category2, xml, algorithm}
{Category2, xml, firewall}
{Category2, xml, protocol}
{Category2, xml, mail}
{Category3, web, database}
{Category3, xml, database}

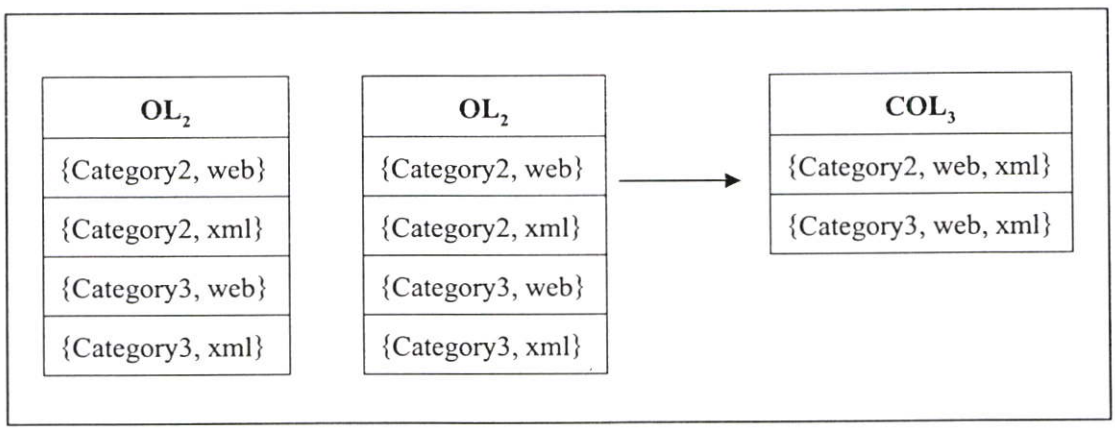
รูปที่ 3.14 ตัวอย่าง L_k

13. ขั้นตอนที่ (13) เป็นการสร้าง Overlap k-itemsets (OL_k) โดยเรียกใช้ฟังก์ชัน Gen_Overlap_k_Itemsets การทำงานของฟังก์ชัน Gen_Overlap_k_Itemsets แสดงดังรูปที่ 3.15

```
%Function Gen_Overlap_k_Itemsets (k, OL(k-1))
(13.1) for (X=1; OL(k-1)X ≠ φ ; X++)
(13.2)   {for (Y=2; OL(k-1)Y ≠ φ ; Y++)
(13.3)          {Insert into COLk
(13.4)           Select OL(k-1)X[1], OL(k-1)X[2],..., OL(k-1)X[k-2], OL(k-1)Y[k-1]
(13.4)           From OL(k-1)
(13.5)           Where OL(k-1)X[1] = OL(k-1)Y[1] and OL(k-1)X[2] = OL(k-1)Y[2] and ... and
OL(k-1)X[k-2] = OL(k-1)Y[k-2]
(13.6)          }
(13.7)   }
(13.8) Return OLk
```

รูปที่ 3.15 ฟังก์ชัน Gen_Overlap_k_Itemsets

จากรูปที่ 3.15 แสดงการทำงานของฟังก์ชัน Gen_Overlap_k_Itemsets เป็นฟังก์ชันในการหา Candidate Overlap k-itemsets (COL_k) ซึ่งเกิดจากการเชื่อมความสัมพันธ์ของ $OL_{(k-1)}$ เอง โดยพิจารณาระหว่าง 2 Itemset ที่ต้องเชื่อมความสัมพันธ์กัน หาก Item ตัวที่ 1 ถึง (k-2) เหมือนกันก็สามารถเชื่อมความสัมพันธ์กันได้ แต่ถ้าไม่เหมือนกันก็จะไม่สามารถเชื่อมความสัมพันธ์กันได้ แสดงตัวอย่างการสร้าง COL_k ดังรูปที่ 3.16



รูปที่ 3.16 การสร้าง COL_k

รูปที่ 3.16 แสดงตัวอย่างการสร้าง COL_k โดยที่ k ในที่นี้คือ 3 ดังนั้นจึงเป็นการสร้าง Candidate Overlap 3-itemset (COL_3) ซึ่งเกิดจากการเชื่อมความสัมพันธ์ของ OL_2 เอง พิจารณาจาก Itemset ที่จะเชื่อมความสัมพันธ์กัน โดย Item ตัวที่ 1 ถึง (k-2) จากตัวอย่างก็คือ Item ตัวที่ 1

ของแต่ละ Itemset หากเหมือนกันก็เชื่อมความสัมพันธ์กันได้ หากไม่เหมือนกันก็จะเชื่อมความสัมพันธ์กันไม่ได้ เช่น {Category2, web} และ {Category2, xml} มี Item ตัวที่ 1 ถึง (k-2) ซึ่งก็คือ 1 เหมือนกัน ดังนั้น Itemset ทั้งสองนี้เชื่อมความสัมพันธ์กันได้เป็น {Category2, web, xml} และเช่นกัน {Category2, web} และ {Category3, xml} ไม่สามารถเชื่อมความสัมพันธ์กันได้ เพราะ Item ตัวแรกไม่เหมือนกัน

14. ขั้นตอนที่ (14) เป็นการหา Overlap k-itemset (OL_k) ในตัวอย่างนี้ก็คือ OL_3 โดยสมาชิกของ OL_3 คือสามาชิกของ COL_3 ที่มีค่าสนับสนุนมากกว่าหรือเท่ากับค่าสนับสนุนน้อยที่สุด แสดงตัวอย่างดังรูปที่ 3.17

COL_3	Support
{Category2, web, xml}	2
{Category3, web, xml}	2

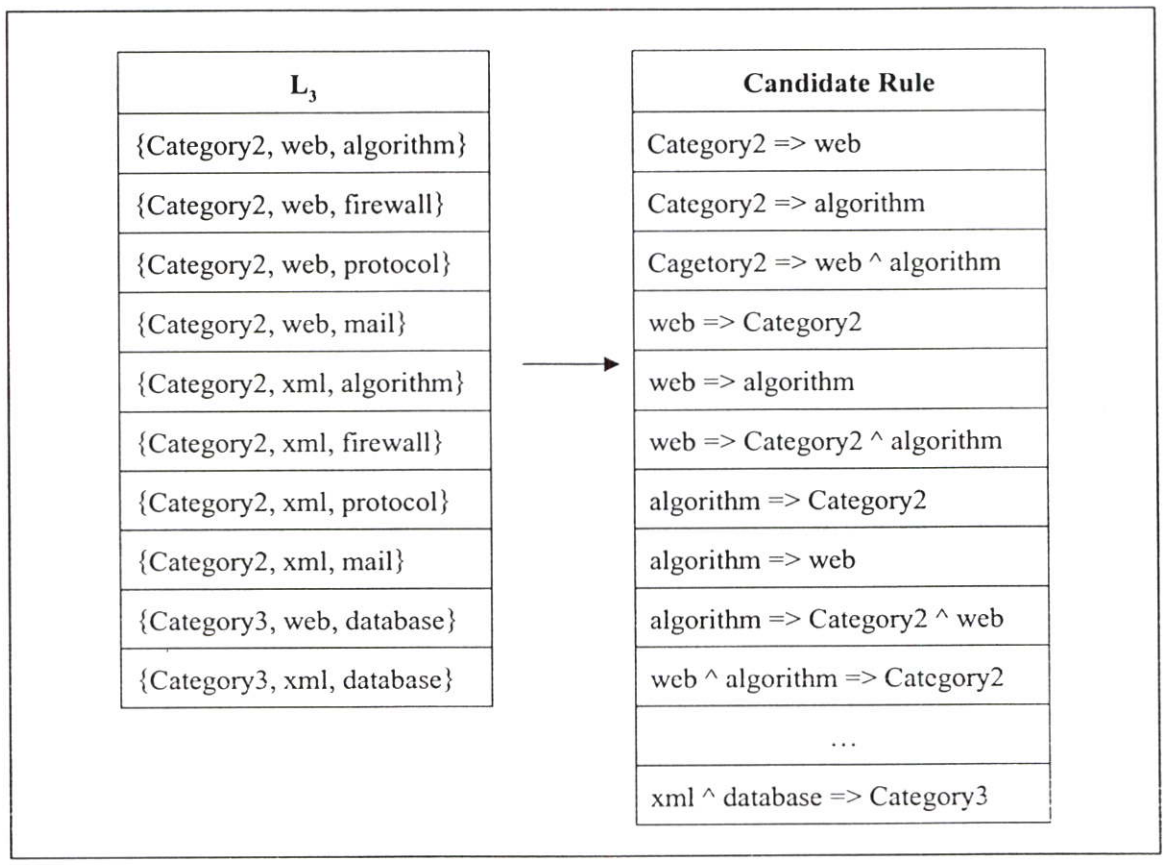
min_sup →

OL_3
{Category2, web, xml}
{Category3, web, xml}

รูปที่ 3.17 ตัวอย่าง OL_k

15. ขั้นตอนที่ (15)-(19) เป็นการนำ Frequet Itemset ได้แก่ L_2, L_3, …, L_k และ OL_2, OL_3, …, OL_k มาสร้างกฎความสัมพันธ์ โดยใช้วิธีการสร้างกฎความสัมพันธ์เช่นเดียวกันอัลกอริทึม Aprioi คือนำ Itemset นั้น ๆ มาหาเซตย่อยแล้วนำมาสร้างเป็นกฎความสัมพันธ์ โดยที่กฎทางด้านขวาจะต้องไม่เป็นซับเซตของกฎทางด้านซ้าย เมื่อได้กฎความสัมพันธ์แล้ว เราจะสนใจแต่กฎที่ทางด้านขวาบอกชื่อกลุ่มเอกสารเท่านั้น แสดงตัวอย่างดังรูปที่ 3.18 และ 3.19

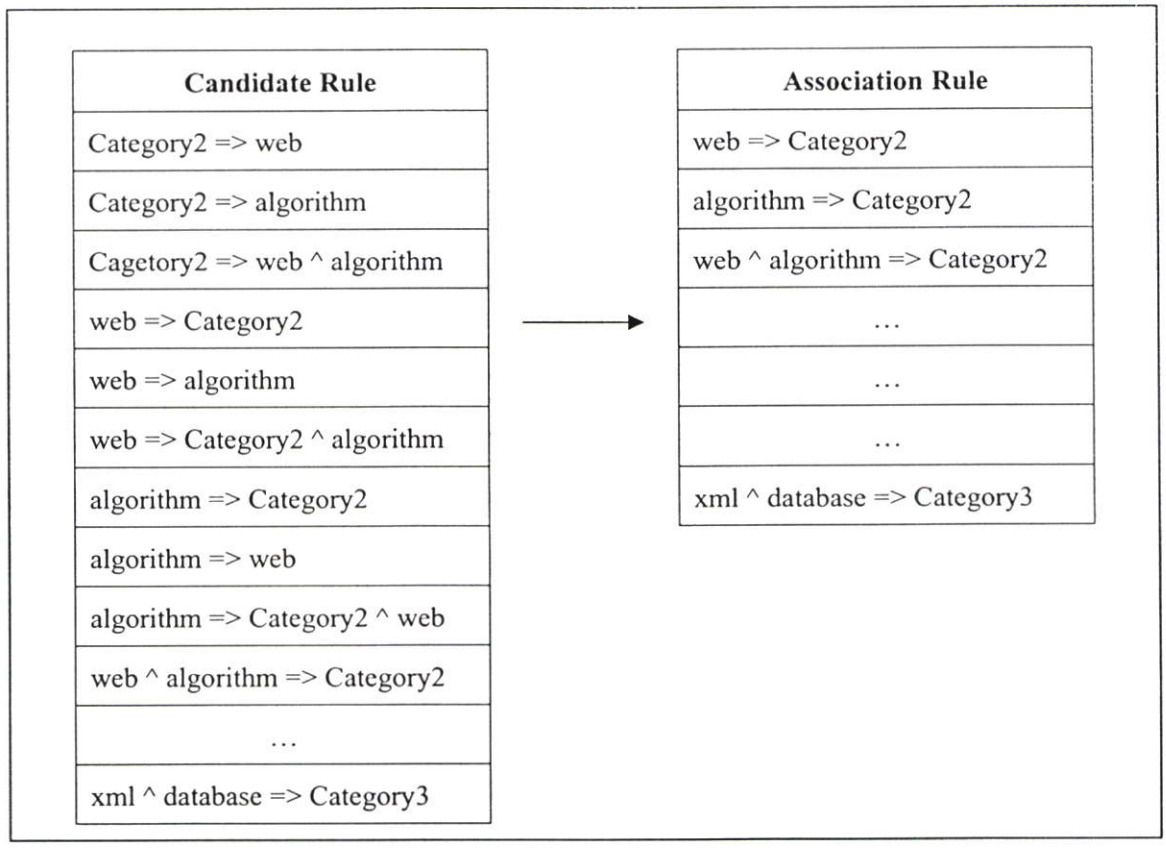
จากรูปที่ 3.18 แสดงตัวอย่างการสร้างกฎความสัมพันธ์ โดยใช้ L_3 นำ Itemset ของ L_3 แต่ละ Itemset มาหาเซตย่อย ตัวอย่างเช่น พิจารณาที่ {Category2, web, algorithm} มีเซตย่อยคือ {Category2}, {web}, {algorithm}, {Category2, web}, {Category2, algorithm} และ {web, algorithm} เมื่อนำมาสร้างกฎความสัมพันธ์ จะได้กฎความสัมพันธ์ดังเซต Candidate Rule และเราสนใจเฉพาะกฎความสัมพันธ์ที่ทางด้านขวาของกฎบอกชื่อกลุ่มเอกสารเท่านั้น ดังนั้นจะได้กฎความสัมพันธ์ดังเซต Rule แสดงดังรูปที่ 3.19



L_3
{Category2, web, algorithm}
{Category2, web, firewall}
{Category2, web, protocol}
{Category2, web, mail}
{Category2, xml, algorithm}
{Category2, xml, firewall}
{Category2, xml, protocol}
{Category2, xml, mail}
{Category3, web, database}
{Category3, xml, database}

Candidate Rule
Category2 => web
Category2 => algorithm
Cagetory2 => web ^ algorithm
web => Category2
web => algorithm
web => Category2 ^ algorithm
algorithm => Category2
algorithm => web
algorithm => Category2 ^ web
web ^ algorithm => Category2
…
xml ^ database => Category3

รูปที่ 3.18 การสร้างกฎความสัมพันธ์โดยใช้ L_3



Candidate Rule
Category2 => web
Category2 => algorithm
Cagetory2 => web ^ algorithm
web => Category2
web => algorithm
web => Category2 ^ algorithm
algorithm => Category2
algorithm => web
algorithm => Category2 ^ web
web ^ algorithm => Category2
…
xml ^ database => Category3

Association Rule
web => Category2
algorithm => Category2
web ^ algorithm => Category2
…
…
…
xml ^ database => Category3

รูปที่ 3.19 การค้นหากฎความสัมพันธ์จาก L_3

เมื่อค้นหาความสัมพันธ์และสร้างกฎความสัมพันธ์ที่จะนำมาใช้จำแนกเอกสารได้แล้ว เนื่องจากกฎความสัมพันธ์ที่ได้มีจำนวนมาก หากนำมาใช้จำแนกเอกสารใหม่ ต้องเสียเวลาในการจำแนกเอกสารเป็นเวลามาก เพื่อเป็นการเพิ่มประสิทธิภาพของกฎความสัมพันธ์ที่ค้นพบ งานวิจัยนี้จึงได้คิดวิธีคัดกฎความสัมพันธ์ทิ้ง โดยพยายามคงไว้ซึ่งกฎที่จำเป็นที่ต้องใช้ในการจำแนกเอกสารเท่านั้น ดังจะกล่าวถึงวิธีการคัดกฎความสัมพันธ์ทิ้งในหัวข้อถัดไป

3.2 การคัดกฎความสัมพันธ์ทิ้ง

เนื่องจากความพยายามที่ต้องการคงไว้ซึ่งกฎความสัมพันธ์ที่ไม่สั้นมากเกินไป (General) และกฎความสัมพันธ์นั้นยังคงระดับความน่าเชื่อถืออยู่ เราจึงนำกฎนั้นมาสร้างเป็นต้นไม้กฎความสัมพันธ์ เพื่อจัดระดับของกฎความสัมพันธ์ (Hierarchy association rule) แล้วคัดกฎความสัมพันธ์ทิ้ง (Prune) โดยใช้ค่า Limit confidence เป็นตัวชี้ว่าจะเก็บกฎนั้นไว้หรือตัดกฎนั้นทิ้ง ซึ่งค่า Limit confidence นี้มีค่าระหว่าง 0-1

ในส่วนนี้จะกล่าวถึงวิธีการคัดกฎความสัมพันธ์ทิ้งของงานวิจัยนี้ โดยแบ่งออกเป็น 2 ขั้นตอนคือ วิธีการสร้างต้นไม้กฎความสัมพันธ์ และวิธีการคัดกฎความสัมพันธ์ทิ้ง

3.2.1 วิธีสร้างต้นไม้กฎความสัมพันธ์

วิธีสร้างต้นไม้กฎความสัมพันธ์ใด ๆ หลังจากได้กฎความสัมพันธ์แล้ว มีหลักการดังนี้

1. Root ของต้นไม้กฎความสัมพันธ์แต่ละต้นจะเป็นชื่อกลุ่มเอกสาร
2. โหนด (Node) 1 โหนดคือกฎความสัมพันธ์ 1 กฎ
3. โหนดแม่ (Parent node) ต้องเป็นกฎที่สั้นกว่า (General rule) โหนดลูก (Child node)
4. โหนดแม่มีโหนดลูกได้ตั้งแต่ 1 โหนดเป็นต้นไป
5. โหนดลูกต้องเป็น Super set ของโหนดแม่ โดยให้ความสำคัญกับลำดับของ Term ในกฎความสัมพันธ์

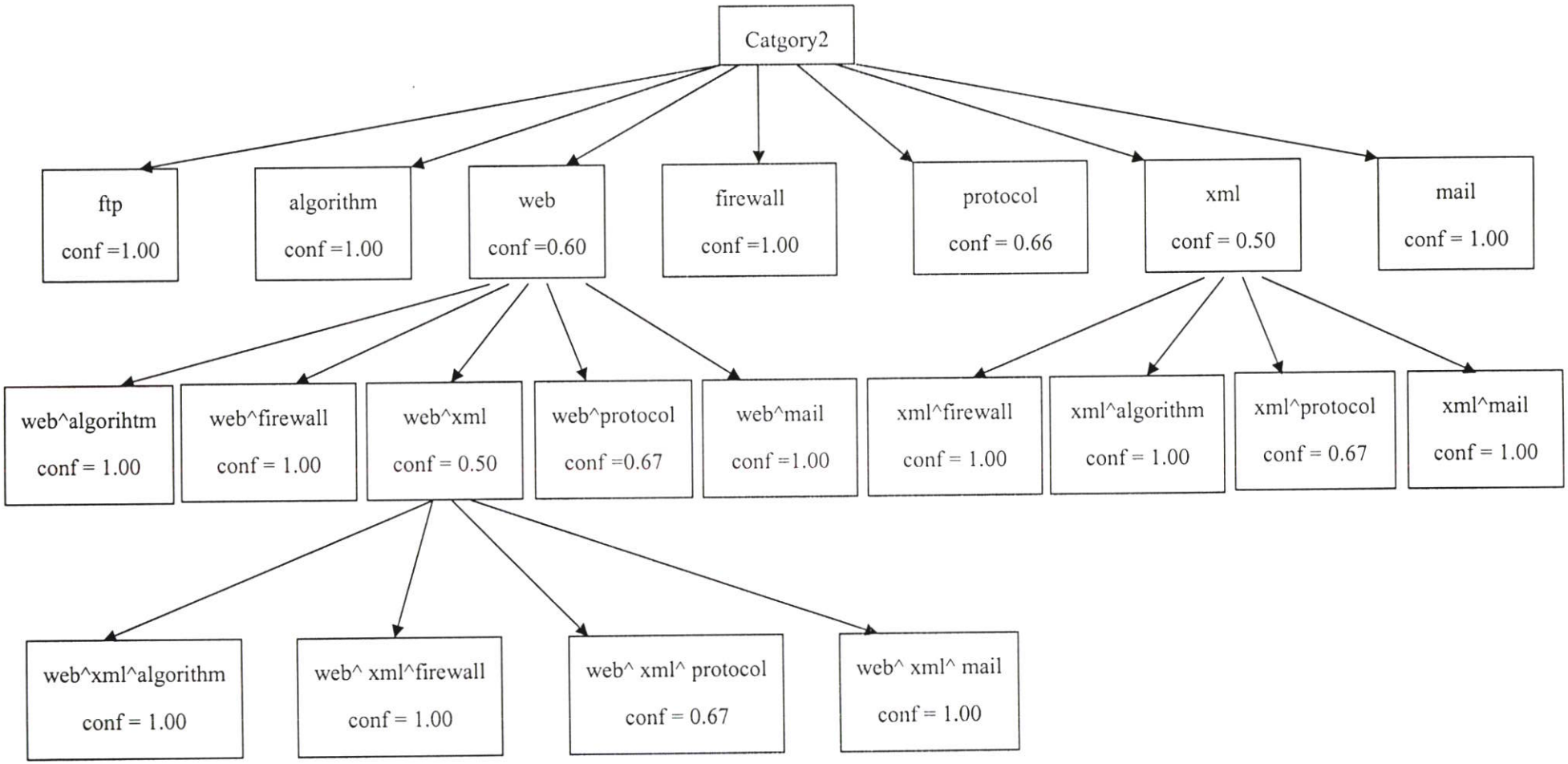
จากหลักการสร้างต้นไม้กฎความสัมพันธ์ เพื่อความเข้าใจมากขึ้น จึงแสดงตัวอย่างวิธีการสร้างต้นไม้กฎความสัมพันธ์ โดยใช้ตัวอย่างกฎความสัมพันธ์จากตารางที่ 3.2

จากตารางที่ 3.2 แสดงกฎความสัมพันธ์และค่าความเชื่อมันของแต่ละกฎความสัมพันธ์ นำมาสร้างต้นไม้กฎความสัมพันธ์ตามหลักการที่กล่าวมาแล้ว จากหลักการสร้างต้นไม้กฎความสัมพันธ์ Root ของต้นไม้คือชื่อกลุ่มเอกสาร จากตารางที่ 3.2 กฎทางด้านขวามีชื่อกลุ่มเอกสารปรากฏดังนี้ Category1, Category2 และ Category3 ดังนั้นจากตารางที่ 3.2 จะมีต้นไม้ด้วยกัน 3 ต้น โดยมี Root คือ Caegory1, Category2 และ Category3 หลังจากนั้นหาโหนดในระดับถัดไป (โหนดที่ถัดจาก Root) ต้องเป็นกฎที่สั้นที่สุดของกลุ่มนั้น แล้วดูว่าโหนดนั้นมีโหนดลูกหรือไม่ หากมีก็ใส่โหนดลูกเข้าไป โดยที่โหนดลูกต้องเป็น Super set ของโหนดแม่ โดยให้

ความสำคัญกับลำดับของ Term ในกฎความสัมพันธ์ แสดงตัวอย่างการสร้างต้นไม้กฎความสัมพันธ์ดังรูปที่ 3.20

ตารางที่ 3.2 ตัวอย่างกฎความสัมพันธ์

กฎความสัมพันธ์	ค่าความเชื่อมั่น
money => Caegory1	1.00
management => Category1	1.00
ftp => Category2	1.00
algorithm => Category2	1.00
web =>Category2	0.60
firewall => Category2	1.00
protocol => Category2	0.66
xml =>Category2	0.50
mail =>Category2	1.00
web^algorithm => Category2	1.00
web^firewall => Category2	1.00
web^protocol => Category2	0.67
web^mail => Category2	1.00
web^xml => Category2	0.50
xml^algorithm => Category2	1.00
xml^firewall => Category2	1.00
xml^protocol => Category2	0.67
xml^mail => Category2	1.00
web^xml^algorithm => Category2	1.00
web^xml^firewall => Category2	1.00
web^xml^protocol => Category2	0.67
web^xml^mail => Category2	1.00
database => Caegory3	1.00
...	...
xml^database => Category3	1.00
web^database => Category3	1.00
web^xml^database => Category3	1.00



รูปที่ 3.20 ต้นไม้กฎความสัมพันธ์

จากรูปที่ 3.20 แสดงตัวอย่างต้นไม้กฎความสัมพันธ์ที่สร้างจากรูปที่ 3.21 โดยเลือกสร้างเพียงต้นเดียวที่มี Root คือ Category2 ดังนั้นกฎความสัมพันธ์ที่จะนำมาสร้างต้นไม้ต้นนี้ได้ต้องเป็นกฎที่ทางด้านขวาคือ Caegory2 โดยให้ Root คือ Category2 หลังจากนั้นหากฎความสัมพันธ์ที่จะเป็นโหนดถัดจาก Root ซึ่งโหนดที่ถัดจาก Root ต้องเป็นกฎความสัมพันธ์ที่สั้นที่สุดซึ่งก็คือ ftp => Category2, algorithm => Category2, web =>Category2, firewall => Category2, protocol => Category2, xml => Category2, mail => Category2 เมื่อสร้างโหนดในระดับนี้หมดแล้ว พิจารณากฎความสัมพันธ์ว่ายังมีกฎความสัมพันธ์เหลืออยู่หรือไม่ พบว่ายังมีอยู่ดังนั้นจึงต้องสร้างโหนดในระดับถัดไป โดยโหนดลูกต้องเป็น Super set ของโหนดแม่ หรือโหนดแม่ต้องเป็น Sub set ของโหนดลูกนั่นเอง เช่น โหนดแม่มีกฎความสัมพันธ์คือ web => Category2 โหนดลูกต้องเป็น web ^ ? => Category2 แสดงดังระดับที่ 2 ของรูปที่ 3.22 และในระดับถัดไปก็เช่นกัน ถ้าโหนดแม่เป็น web^xml => Category2 แล้วโหนดลูกของโหนดนี้ต้องเป็น web ^ xml ^? => Category2

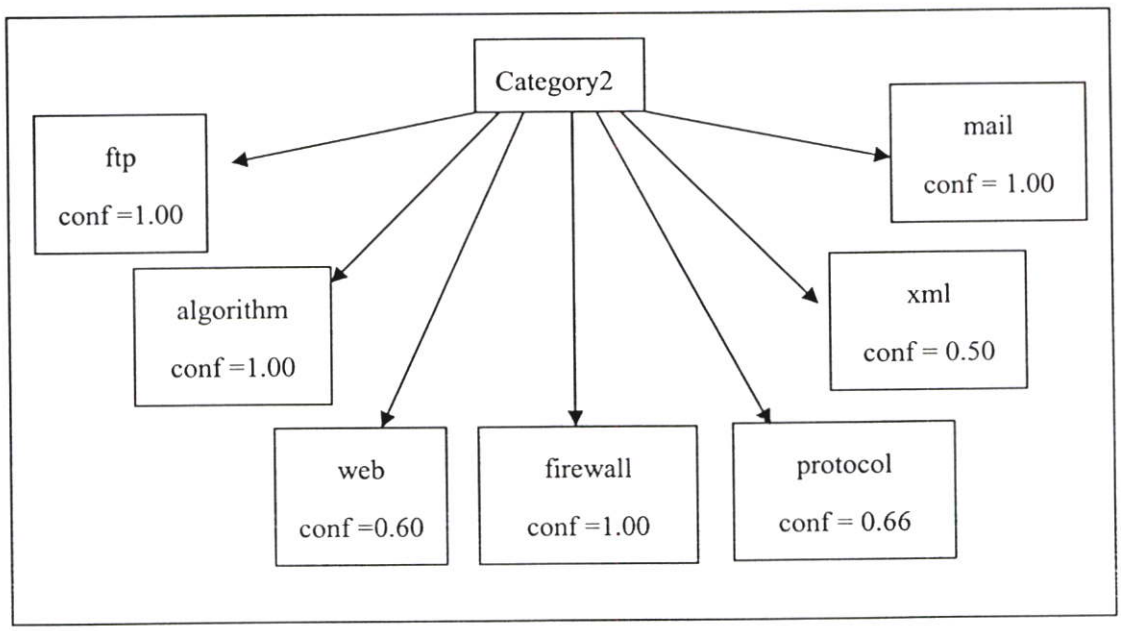
3.2.2 วิธีคัดกฎความสัมพันธ์ทิ้ง

ก่อนคัดกฎความสัมพันธ์ทิ้งต้องกำหนดค่า Limit confidence ซึ่งเป็นค่าที่เรากำหนดขึ้นมา โดยจะมีค่าระหว่าง 0-1 เพื่อบอกว่ากฎความสัมพันธ์ที่ได้ต้องเป็นกฎที่มีค่าความเชื่อมั่นของกฎความสัมพันธ์นั้นมากกว่าหรือเท่ากับค่า Limit confidence จึงจะเป็นกฎที่งานวิจัยนี้ยอมรับในการใช้จำแนกเอกสาร

หลักการในการคัดกฎความสัมพันธ์ทิ้งจากต้นไม้กฎความสัมพันธ์ที่ได้นั้นจะใช้วิธีพิจารณาจากบนลงล่าง โดยพิจารณาจาก Root จนถึงโหนดสุดท้าย (Left node) ดังหลักการดังนี้

1. โหนดแม่มีค่าความเชื่อมั่นมากกว่าหรือเท่ากับค่า Limit confidence ให้เก็บกฎจากโหนดแม่ไว้และตัดโหนดลูกทิ้งให้หมด
2. โหนดแม่มีค่าความเชื่อมั่นน้อยกว่าค่า Limit confidence ให้พิจารณาโหนดลูกทุก ๆ โหนดของโหนดแม่ ถ้ามีค่าความเชื่อมั่นมากกว่าหรือเท่ากับค่า Limit confidence ให้เก็บกฎนั้นไว้และตัดโหนดลูกของโหนดนั้นทิ้ง แต่หากค่าความเชื่อมั่นของกฎนั้นน้อยกว่าค่า Limit confidence ก็จะต้องดูว่าโหนดนั้นมีลูกหรือไม่ แล้วพิจารณาต่อไปว่ามีค่าความเชื่อมั่นมากกว่า หรือน้อยกว่าค่า Limit confidence

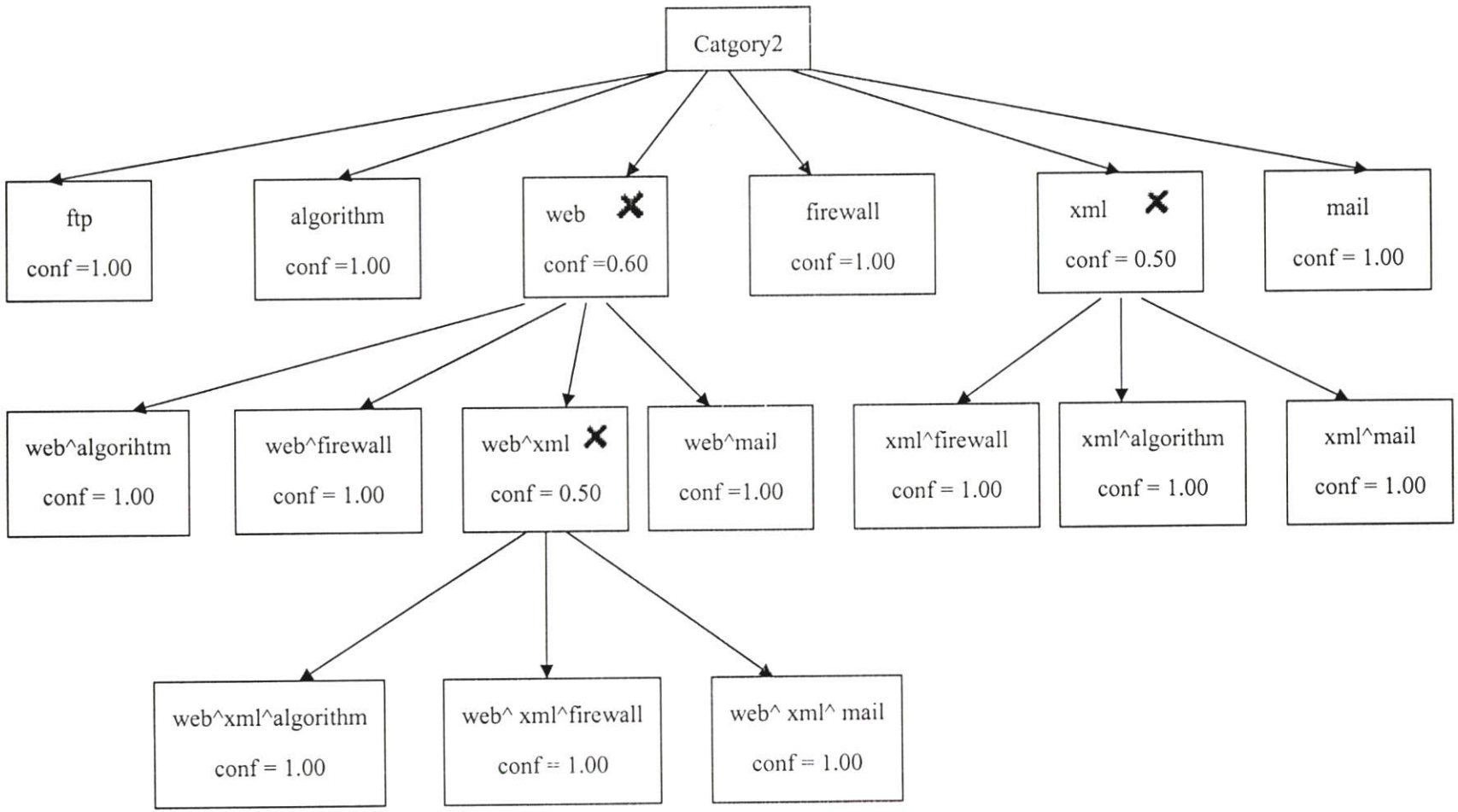
แสดงตัวอย่างการคัดกฎความสัมพันธ์ทิ้งดังรูปที่ 3.21 โดยกำหนดให้ค่า Limit confidence มีค่า 0.50



รูปที่ 3.21 โครงสร้างต้นไม้กฎความสัมพันธ์ หลังจากคัดกฎความสัมพันธ์ทิ้งที่ Limit confidence = 0.50

จากรูปที่ 3.21 แสดงโครงสร้างต้นไม้กฎความสัมพันธ์หลังจากคัดกฎความสัมพันธ์ทิ้งที่ Limit confidence มีค่า 0.50 โดยอ้างอิงจากรูปที่ 3.20 พิจารณาที่ระดับที่ 1 เริ่มต้นที่ ftp => Category2 มีค่าความเชื่อมั่น 1.00 เปรียบเทียบกับค่า Limit confidence พบว่าค่าความเชื่อมั่นของกฎนี้มีค่ามากกว่าค่า Limit confidence ดังนั้นให้เก็บกฎจากโหนดนี้แล้วตัดโหนดลูกที่มีของโหนดนี้ทิ้ง จากรูปที่ 3.20 พบว่าโหนดนี้ไม่มีโหนดลูกก็ไม่ต้องทำอะไร พิจารณาเช่นนี้จะครบทุกโหนดในระดับเดียวกัน และทำนองเดียวกัน ที่โหนด web => Category2 มีค่าความเชื่อมั่น 0.60 ซึ่งมากกว่าค่า Limit confidence ดังนั้นเก็บกฎจากโหนดนี้และตัดโหนดลูกของโหนดนี้ทั้งหมด เมื่อพิจารณาครบทุกโหนดในระดับเดียวกันแล้ว พบว่าทุกโหนดในระดับนี้มีค่าความเชื่อมั่นมากกว่าหรือเท่ากับค่า Limit confidence ทั้งหมด ดังนั้นจึงได้กฎดังรูปที่ 3.21 นั่นก็คือกฎที่จะนำมาใช้ในการจำแนกเอกสารที่ Limit confidence มีค่า 0.50 ได้แก่ ftp => Category2, algorithm=> Category2, web => Category2, firewall => Category2, protocol => Category2, xml =>Category2 และ mail =>Category2

แสดงตัวอย่างอีกตัวอย่างในการคัดกฎความสัมพันธ์ทิ้ง ที่ค่า Limit confidence มีค่า 0.70 ดังรูปที่ 3.22



รูปที่ 3.22 โครงสร้างต้นไม้กฎความสัมพันธ์หลังจากคัดกฎความสัมพันธ์ทิ้งที่ Limit confidence = 0.70

จากรูปที่ 3.22 แสดงโครงสร้างต้นไม้กฎความสัมพันธ์หลังจากคัดกฎความสัมพันธ์ทิ้งที่ Limit confidence มีค่า 0.70 โดยอ้างอิงจากตารางที่ 3.2 พิจารณาที่ระดับที่ 1 พบว่ามี 3 กฎที่มีค่าความเชื่อมั่นไม่ผ่านค่า Limit confidence คือ web => Category2, protocol => Category2 และ xml => Category2 ดังนั้นให้พิจารณาโหนดลูกของทั้ง 3 โหนดนี้ต่อไป สำหรับโหนด protocol => Category2 นั้นไม่มีโหนดลูกแล้วแต่เนื่องจากกฎของโหนดนี้ก็ไม่ผ่านค่า Limit confidence ดังนั้นจึงตัดโหนดนี้ทิ้ง พิจารณาอีก 2 โหนด คือ web => Category2 และ xml => Category2 ให้มาร์ค (Mark) ไว้ว่ากฎของโหนดนี้ไม่ได้นำไปใช้ในการจำแนกเอกสาร แล้วพิจารณาโหนดลูกของทั้งสองโหนดนี้ต่อไป หากโหนดไหนที่มีค่าความเชื่อมั่นมากกว่าหรือเท่ากับค่า Limit confidence ก็เก็บกฎในโหนดนั้นไว้และตัดโหนดลูกของโหนดนั้นทิ้ง หากไม่ใช่ก็จะมาร์คโหนดแม่ไว้แล้วหาโหนดลูกต่อไปจนถึงโหนดสุดท้าย ก็จะพบว่าที่โหนดสุดท้าย (Leaf node) จะเก็บกฎความสัมพันธ์ที่มีค่าความเชื่อมั่นมากกว่าหรือเท่ากับค่า Limit confidence ดังรูปที่ 3.23

ftp => Category2
algorithm => Category2
firewall => Category2
mail =>Category2
web^algorithm => Category2
web^firewall => Category2
web^mail => Category2
xml^algorithm => Category2
xml^firewall => Category2
xml^mail => Category2
web^xml^algorithm => Category2
web^xml^firewall => Category2
web^xml^mail => Category2

รูปที่ 3.23 กฎความสัมพันธ์ที่ Limit confidence = 0.70

รูปที่ 3.23 แสดงกฎความสัมพันธ์ที่ได้หลังจากผ่านการคัดกฎความสัมพันธ์ทิ้งที่ค่า Limit confidence = 0.70

3.3 การจำแนกประเภทเอกสาร

ขั้นตอนการจำแนกเอกสารของงานวิจัยนี้แสดงได้ดังรูปที่ 3.24 โดยมีข้อมูลนำเข้าดังนี้

1. กลุ่มเอกสารอยู่ในรูปแบบ $\{t_1,t_2,\ldots,t_n\}$ โดยที่ t เป็นคุณลักษณะของเอกสาร มีทั้งหมด n ตัว
2. กฎความสัมพันธ์

ผลลัพธ์ที่ได้คือเอกสารถูกจำแนกว่าอยู่กลุ่มใด ในการอธิบายขั้นตอนการจำแนกเอกสารตามรูปที่ 3.24 ขอยกตัวอย่างประกอบอัลกอริทึมเพื่อให้เข้าใจมากขึ้น โดยกำหนดให้เอกสารที่เข้ามาในระบบเพื่อใช้จำแนกกลุ่มคือ เอกสาร D ที่มีคุณลักษณะของเอกสารดังนี้

D = {web, xml, firewall}

กำหนดให้ Set of Rule หรือกลุ่มของกฎความสัมพันธ์แสดดังตารางที่ 3.3 และอธิบายขั้นตอนการทำงานของอัลกอริทึมการจำแนกเอกสารดังนี้

1. ขั้นตอนที่ (1) เป็นการนำเอกสาร D ที่ต้องการจำแนกกลุ่มเข้าสู่ระบบ
2. ขั้นตอนที่ (2) เป็นการเรียกใช้ฟังก์ชัน Find_rule_activate ซึ่งทำหน้าที่ในการหากฎความสัมพันธ์ว่ามีกฎความสัมพันธ์ใดบ้างที่สามารถนำมาใช้จำแนกเอกสารนี้ได้ โดยกฎที่ใช้จำแนกเอกสารนี้ได้นั้นคือกฎที่มี Term ทุก ๆ Term เหมือนกับคุณลักษณะของเอกสารที่ต้องการนำมาจำแนกกลุ่ม จากตัวอย่าง Set of Rule ในตารางที่ 3.3 นำมาหากฎความสัมพันธ์ที่จะใช้จำแนกเอกสาร D พิจารณาที่ ftp => Category2 เนื่องจากกฎทางด้านซ้ายของกฎนี้ไม่มี Term ใดที่เหมือนกับคุณลักษณะของเอกสาร D ดังนั้นกฎนี้จึงไม่ถูกเลือกมาใช้ในการจำแนกเอกสาร D กฎต่อไป firewall => Category2 เป็นกฎที่ต้องใช้ในการจำแนกเอกสาร D เพราะกฎทางด้านซ้ายมี term ที่เหมือนกับคุณลักษณะของเอกสาร D และถัดไปกรณี web^algorithm => Category2 ไม่ถูกเลือกให้เป็นกฎที่ใช้ในการจำแนกเอกสาร D เพราะกฎทางด้านซ้ายมี Term ที่ไม่เหมือนกับคุณลักษณะของเอกสาร D กฎที่ถูกเลือกเพื่อนำมาใช้ในการจำแนกเอกสาร D แสดงดังตารางที่ 3.4
3. ขั้นตอนที่ (3)-(4) เป็นการเริ่มการจำแนกกลุ่มเอกสาร โดยเป็นเงื่อนไขแรกในการพิจารณาว่าเอกสารควรอยู่กลุ่มใด โดยในขั้นตอนนี้จะหาก่อนว่ากฎความสัมพันธ์ที่ใช้ในการจำแนกเอกสาร D นี้ มีกฎใดที่ได้มาเหมือนกับคุณลักษณะของเอกสาร D หรือไม่ โดยกฎที่เหมือนกับคุณลักษณะของเอกสาร D ได้แก่ web^xml^firewall => {Category label} ซึ่งจากกฎความสัมพันธ์ที่ได้ไม่มีกฎใดมีลักษณะเช่นนี้ ดังนั้นจึงต้องพิจารณาในขั้นตอนต่อไป

Algorithm: Classifying new documents.

Input: 1.A set of documents of the form $\{t_1, t_2, \ldots, t_n\}$ where t_j are document terms.

2. A set of association rule (Rule set)

Output: 1. A set of documents of the form $\{c_i, t_1, t_2, \ldots, t_n\}$ where c_i is the category label to the document and t_j are document terms.

Method:

```
(1) For each new documents
(2) A_rule ◄── Find_rule_activate(document,Rule)
(3) If A_rule = = matching document
(4)       Classifying new document
(5) Else
(6)       if number of A_rule > 1
(7)             Group nA_rule by category label {C1, C2,...,Cn}
(8)             Find sum of confidence
(9)             Find sum of support
(10)            If no_of_rule_C1 > no_of_rule_C2
(11)                  Classifying new document by C1
(12)            Else
(13)                  If sum_conf_C1 > sum_conf_C2
(14)                        Classifying new document by C1
(15)                  Else
(16)                        if sum_support_C1 > sum_support_C2
(17)                              Classifying new document by C1
(18)                        End
(19)                  End
(20)            End
(21)      End
(22)  End
(23)next document
```

รูปที่ 3.24 อัลกอริทึมสำหรับจำแนกเอกสาร

ตารางที่ 3.3 ตัวอย่าง Set of rule

กฎความสัมพันธ์	ค่าสนับสนุน	ค่าความเชื่อมั่น
ftp => Category2	2	0.70
algorithm => Category2	2	0.80
firewall => Category2	3	1.00
mail =>Category2	4	1.00
web^algorithm => Category2	2	1.00
web^firewall => Category2	3	0.80
web^mail => Category2	3	0.90
xml^algorithm => Category2	4	1.00
xml^firewall => Category2	2	1.00
xml^mail => Category2	2	0.95
web^xml^mail => Category2	3	1.00
web => Category3	5	1.00
web^algorithm => Category3	2	1.00
web^firewall => Category3	2	0.70

ตารางที่ 3.4 กฎที่ใช้ในการจำแนกเอกสาร D

กฎความสัมพันธ์	ค่าสนับสนุน	ค่าความเชื่อมั่น
firewall => Category2	3	1.00
web^firewall => Category2	3	0.80
xml^firewall => Category2	2	1.00
xml => Category3	2	1.00
web^firewall => Category3	2	0.70

4. ขั้นตอนที่ (6) เป็นการเช็คเงื่อนไขต่อไปของการจำแนกเอกสารหลังจากที่เงื่อนไขไม่ตรงกับขั้นตอนที่ (3) นั่นคือกฎความสัมพันธ์ที่ใช้ในการจำแนกเอกสารมีมากกว่า 1 กฎ
5. ขั้นตอนที่ (7) เป็นการนำกฎความสัมพันธ์ที่ได้มาจัดรวมเป็นกลุ่ม โดยรวมกฎที่มีกฎทางด้านขวาเหมือนกันไว้กลุ่มเดียวกัน แสดงการจัดกลุ่มกฎความสัมพันธ์ดังรูปที่ 3.25

กฎความสัมพันธ์	ค่าสนับสนุน	ค่าความเชื่อมั่น	
firewall => Category2	3	1.00	C1
web^firewall => Category2	3	0.80	
xml^firewall => Category2	2	1.00	
xml => Category3	2	1.00	C2
web^firewall => Category3	2	0.70	

รูปที่ 3.25 แสดงการแบ่งกลุ่มของกฎความสัมพันธ์

จากรูปที่ 3.25 จะแบ่งกลุ่มกฎความสัมพันธ์ได้ 2 กลุ่มคือ C1 และ C2 โดยที่ C1 เก็บกลุ่มของ Category2 และ C2 เก็บกลุ่มของ Category3

6. ขั้นตอนที่ (8) เป็นการหาค่าผลรวมค่าความเชื่อมั่นของกฎความสัมพันธ์ ซึ่งหาได้ดังสมการที่ (3.1)

$$\text{ผลรวมค่าความเชื่อมั่น} = \frac{\text{ผลบวกค่าความเชื่อมั่นของแต่ละกฎในกลุ่มเดียวกัน}}{\text{จำนวนกฎที่อยู่ในกลุ่มเดียวกัน}} \quad (3.1)$$

จากรูปที่ 3.25 แสดงวิธีการหาผลรวมค่าความเชื่อมั่นของแต่ละกลุ่มดังนี้

$$C1 = (1.00+0.80+1.00)/3$$

$$= 9.33$$

$$C2 = (1.00 +0.70)/2$$

$$= 0.85$$

7. ขั้นตอนที่ (9) เป็นการหาค่าผลรวมค่าสนับสนุนของกฎความสัมพันธ์ ซึ่งหาได้ดังสมการที่ (3.2)

$$\text{ผลรวมค่าสนันสนุน} = \frac{\text{ผลบวกค่าสนับสนุนของแต่ละกฎในกลุ่มเดียวกัน}}{\text{จำนวนกฎที่อยู่ในกลุ่มเดียวกัน}} \quad (3.2)$$

จากรูปที่ 3.25 แสดงวิธีการหาผลรวมค่าสนับสนุนของแต่ละกลุ่มดังนี้

$$C1 = (3+3+2)/3$$

$$= 2.66$$

$$C2 = (2+2)/2$$

$$= 2$$

8. ขั้นตอนที่ (10)-(23) เป็นการจำแนกเอกสารโดยการเช็คเงื่อนไขต่าง ๆ ตามลำดับคือ หากกฎความสัมพันธ์ที่ได้ กลุ่มใดมีจำนวนกฎมากกว่าให้จำแนกว่าเอกสารอยู่กลุ่มนั้น แต่ถ้ากลุ่มใด ๆ มีจำนวนกฎเท่ากันให้พิจารณาที่ค่าผลรวมของค่าความเชื่อมั่นกฎ หากกลุ่มใดมีค่าผลรวมค่าความเชื่อมั่นกฎมากกว่าให้จำแนกว่าเอกสารอยู่กลุ่มนั้น แต่ถ้าผลรวมค่าความเชื่อมั่นกฎเท่ากันให้พิจารณาที่ผลรวมค่าสนับสนุน ถ้ากลุ่มใดมีค่ามากกว่าให้จำแนกว่าเอกสารอยู่กลุ่มนั้น จากตัวอย่างพบว่าตรงตามเงื่อนไขแรกคือจำนวนกฎของกลุ่ม C1 มีมากกว่ากลุ่ม C2 ดังนั้นเอกสารนี้บอกว่าอยู่ Category2 ตามกฎทางด้านขวาของ C1

ในบทต่อไปจะกล่าวถึงการทดลองและผลการทดลองกับชุดข้อมูลที่นำมาทดลองกับอัลกอริทึมงานวิจัยนี้ และงานวิจัยที่เกี่ยวข้อง

บทที่ 4

การทดลอง และผลการทดลอง

ในบทนี้จะกล่าวถึงชุดข้อมูลที่ใช้ในการทดลอง การทดลอง และผลการทดลอง ซึ่งเป็นการเปรียบเทียบประสิทธิภาพในเชิงความถูกต้องในการจำแนกประเภทเอกสารระหว่างอัลกอริทึม ARC-BC และอัลกอริทึม ARTC ซึ่งเป็นผลงานของงานวิจัยฉบับนี้

ในการสร้าง Model ต่าง ๆ ในงานวิจัยนี้ใช้โปรแกรม MATLAB 6.5 ในการทดลองทั้งหมด และใช้โปรแกรม MATLAB 6.5, PHP 4.3.8, IIS 5.1และ Copernic Summarizer ในการเตรียมข้อมูลก่อนการทดลอง

4.1 ชุดข้อมูลที่ใช้ในการทดลอง

ในส่วนนี้จะกล่าวถึงชุดข้อมูลที่ใช้ในการทดลอง โดยแบ่งออก 2 ส่วนคือ ลักษณะของชุดข้อมูลที่ใช้ในการทดลอง และการเตรียมข้อมูลในการทดลอง

4.1.1 ชุดข้อมูล

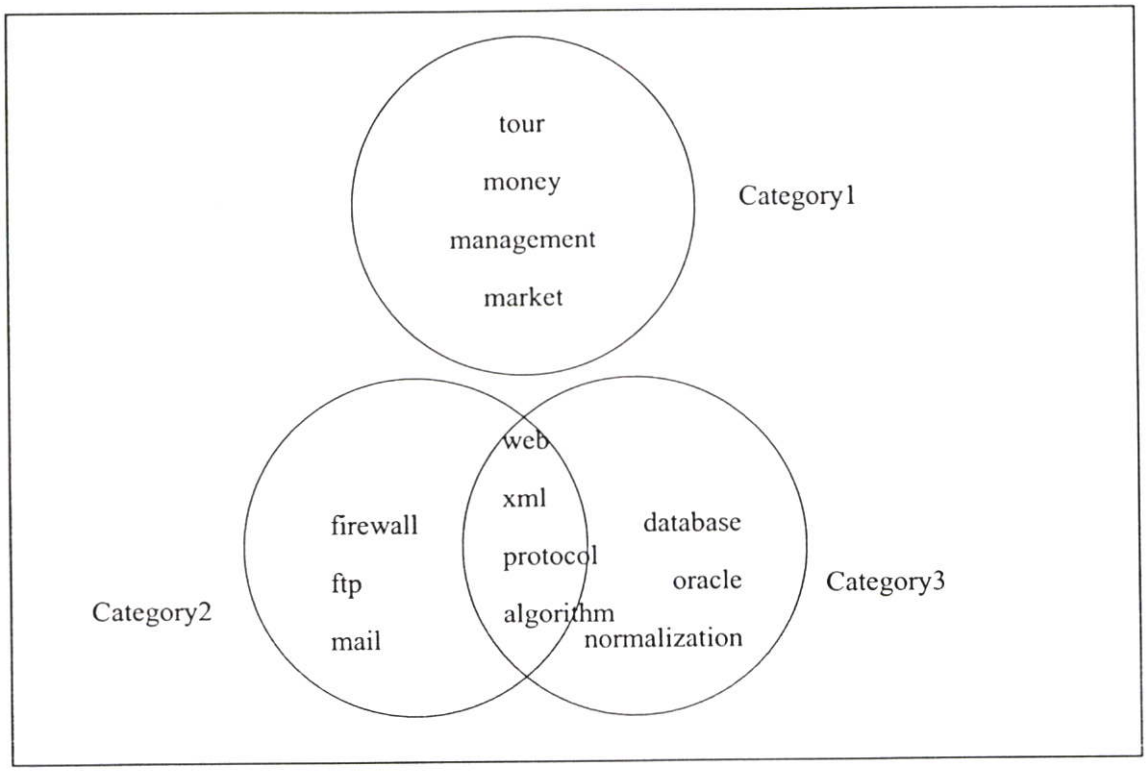
ชุดข้อมูลที่ใช้ในการทดลองประกอบไปด้วยชุดข้อมูล 4 ชุดคือ ชุดข้อมูลข้อความ ชุดข้อมูล CSTR ชุดข้อมูล K-dataset และชุดข้อมูล Reuters-Top10 ดังอธิบายชุดข้อมูลแต่ละชุดต่อไปนี้

4.1.1.1 ชุดข้อมูลข้อความ

ชุดข้อมูลข้อความอักษร เป็นชุดข้อมูลที่สร้างขึ้นด้วยข้อความภาษาอังกฤษ โดยแบ่งข้อมูลที่สร้างขึ้นเป็น 3 กลุ่ม แสดงลักษณะของชุดข้อมูลดังรูปที่ 4.1

จากรูปที่ 4.1 เป็นลักษณะข้อมูลของชุดข้อมูลข้อความอักษรที่แบ่งข้อมูลเป็น 3 กลุ่ม คือ Category1, Category2 และ Category3 โดยที่ข้อความที่ปรากฏอยู่ในแต่ละกลุ่มคือคำสำคัญ (Keyword) ของเอกสารแต่ละกลุ่ม จากลักษณะของชุดข้อมูลที่สร้างขึ้น เราได้จำลองลักษณะของเอกสารที่มีอยู่จริงทั่ว ๆ ไปกล่าวคือ เอกสารที่ไม่มี Keyword ไปเกี่ยวข้องกับกลุ่มอื่นๆ ซึ่งก็คือ Category1 จะกล่าวอีกนัยก็คือเอกสารที่สามารถบอกได้ชัดเจน 100% ว่าควรอยู่กลุ่มใด และลักษณะเอกสารที่มี Keyword ซ้อนทับกับกลุ่มอื่น ซึ่งก็คือ Category2 และ Category3 ซึ่งอาจจะมีเอกสารในบางส่วนที่ไม่สามารถบอกได้ชัดเจน 100% ว่าเอกสารควรอยู่กลุ่มใด

การสร้างข้อมูลชุดนี้ ใช้วิธีการสุ่มข้อความจากลักษณะข้อมูลในรูปที่ 4.1 เป็นจำนวน 2000 เอกสาร แสดงรายละเอียดข้อมูลดังตารงที่ 4.1 และตัวอย่างเอกสารที่สร้างขึ้นดังตารางที่ 4.2



รูปที่ 4.1 ลักษณะของชุดข้อมูลข้อความ

ตารางที่ 4.1 รายละเอียดชุดข้อมูลข้อความ

ลำดับ	ชื่อกลุ่ม	จำนวน
1	Category1	684
2	Category2	682
3	Category3	634
รวม		2000

ตารางที่ 4.2 ตัวอย่างเอกสารชุดข้อมูลข้อความ

ลำดับ	คำสำคัญ	กลุ่มเอกสาร
1	firewall, ftp, web, xml	Category2
2	database, oracle, wet, protocol	Categoyr3
3	web, xml, protocol, mail, ftp	Categoy2
4	normalization, oracle, database, xml	Categoy3
5	database, oracle, normalization	Categoy3
6	tour, money, management, market, hotel	Categoy1
7	management, hotel, tour, money	Categoy1
8	hotel, tour, money	Categoy1

4.1.1.2 ชุดข้อมูล CSTR

ชุดข้อมูล Computer Science Technical Report (CSTR) [5] เป็นชุดข้อมูลที่เป็นบทคัดย่องานวิจัยของคณะวิทยาศาสตร์มหาวิทยาลัย Rochester ที่เก็บอยู่ในรูปเอกสาร Html file โดยข้อมูลที่นำมาใช้ในการทดลองงานวิจัยนี้เป็นข้อมูลที่อยู่ระหว่างปี ค.ศ. 1978 – 2005 จำนวน 542 เอกสาร แบ่งออกเป็น 4 กลุ่ม คือ AI, Robotics and Vision, Systems และ Theory แสดงรายละเอียดชุดข้อมูล CSTR ดังตารางที่ 4.3 และตัวอย่างข้อมูลดังรูปที่ 4.2

ตารางที่ 4.3 รายละเอียดชุดข้อมูล CSTR

ลำดับ	ชื่อกลุ่ม	จำนวนเอกสาร
1	AI	112
2	Robotics and Vision	88
3	Systems	195
4	Theory	147
รวม		542

<LI>Ahn, David D., "The Role of Presuppositions and Situations in the Interpretation of Adverbial Quantifiers", TR793, Computer Science Dept., U. Rochester, November 2002.
<Ahref="ftp://ftp.cs.rochester.edu/pub/papers/ai/02.tr793.Presupp_and_situations_in_interp_of_adverbial_quantifiers.ps.gz">02.tr793.Presupp_and_situations_in_interp_of_adverbial_quantifiers.ps.gz</A>

<P>Keywords: computational semantics; presuppositions; situation theory; quantificational adverbs.

<BLOCKQUOTE>This paper describes a method for computing the domain of quantification of an adverbially quantified sentence. This method relies on the accommodation of presuppositions in the scope of a quantificational adverb and on the resolution of the domain in context. Situations form the link between adverbial quantifiers and presuppositions, as adverbial quantifiers are taken to quantify over situations and presuppositions are taken to be constraints on resource situations. This paper also briefly describes a computational system for processing such sentences based on this method. </BLOCKQUOTE>

รูปที่ 4.2 ตัวอย่างเอกสารชุดข้อมูล CSTR

4.1.1.3 ชุดข้อมูล K-dataset

ชุดข้อมูล K-dataset [6] เป็นข้อมูลที่เกิดขึ้นจากโครงการ WebACE [7] เป็นเอกสารข่าว Reuters ที่ให้บริการบนเครือข่ายอินเตอร์เน็ตในเดือนตุลาคม ปี ค.ศ. 1997 เก็บอยู่ในรูป Html file ประกอบด้วยเอกสาร 2340 เอกสาร จำนวน 20 กลุ่มเอกสาร แสดงรายละเอียดชุดข้อมูล K-dataset ดังตารางที่ 4.4 และตัวอย่างข้อมูลดังรูปที่ 4.3

ตารางที่ 4.4 รายละเอียดชุดข้อมูล K-dataset

ลำดับ	ชื่อกลุ่ม	จำนวนเอกสาร
1	Business	142
2	Entertainment	9
3	Art	24
4	Cable	44
5	Culture	74
6	Film	278
7	Industry	70
8	Media	21
9	Multimedia	14
10	Music	125
11	Online	65
12	People	248
13	Review	158
14	Stage	18
15	Television	187
16	Variety	54
17	Health	494
18	Politics	114
19	Sports	141
20	Technology	60
รวม		2340

```
<!-- Yahoo TimeStamp: 874292400 -->
Sunday September 14 11:00 PM EDT
</strong>
<h2>NBC Leads Field At 1997 Emmy Awards</h2>
<!-- TextStart -->
<p>  PASADENA, Calif., Sept 14 (Reuter) - NBC won the battle of the television networks Sunday, grabbing 24 awards at the annual Emmy Awards ceremony.
<p>  Led by a powerful lineup of comedies such as &quot;Seinfeld&quot;, &quot;Frasier&quot; &quot;3rd Rock From the Sun&quot; and &quot;Mad About You&quot;, NBC beat out cable television network HBO which had 19 awards, CBS, which had 12, and ABC which had 10.
<p>  Reuters/Variety
<!-- TextEnd -->
<!-- StartRelated -->
<!-- EndRelated -->
```

รูปที่ 4.2 ตัวอย่างเอกสารชุดข้อมูล K-dataset

4.1.1.4 ชุดข้อมูล Reuters-Top10

ชุดข้อมูล Reuters-Top10 [8] เป็นส่วนหนึ่งของชุดข้อมูล Reuters-21578 [9] ชุดข้อมูลมาตรฐานที่ใช้ในการจำแนกประเภทเอกสารที่มี 21578 เอกสาร 135 กลุ่มข้อมูล แต่สำหรับ Reuters-Top10 จะนำข้อมูลมาใช้จำนวน 10 กลุ่มข้อมูลที่มีการใช้งานมากที่สุดประกอบด้วยเอกสาร 2775 เอกสาร เก็บอยู่ในรูป Xml file แสดงรายละเอียดชุดข้อมูลดังตารางที่ 4.4 และตัวอย่างข้อมูลดังรูปที่ 4.4

ตารางที่ 4.5 รายละเอียดชุดข้อมูล Reuters-Top10

ลำดับ	ชื่อกลุ่ม	จำนวนเอกสาร
1	Acq	719
2	Corn	56
3	Crude	189
4	Earn	1085
5	Grain	148
6	Interest	129
7	Money-fx	176
8	Ship	89
9	Trade	113
10	Wheat	71
รวม		2775

<?xml version="1.0" ?>

<MESSAGE>PHOENIX FINANCIAL <PHFC> BUYS DATA ACCESS STAKE BLACKWOOD, N.J., April 9 - Data Access Systems Inc said chairman David Cohen has sold 1,800,000 common shares to Phoenix Financial corp for undisclosed terms and resigned as chairman and chief executive officer. The company said Phoenix Financial now has a 27 pct interest in Data Access and effective control. Data Access said Phoenix chairman Martin S. Ackerman has been named chairman of Data Access as well and two other Phoenix representatives have been named to the Data Access board. It said four directors other than Cohen have resigned from the board. Reuter</MESSAGE>

รูปที่ 4.4 ตัวอย่างเอกสารชุดข้อมูล Reuters-Top10

4.1.2 การเตรียมข้อมูล

ในงานวิจัยนี้ใช้ข้อมูล 1 มิติ (Dimension) ในการทดลอง ซึ่งก็คือคำสำคัญ (Keyword) ของเอกสาร ดังนั้นในการเตรียมข้อมูลจึงเลือกเฉพาะ Keyword ซึ่งมีหลักการดังนี้

1. เลือกข้อมูลที่เป็นเฉพาะข้อความของเอกสารแต่ละชุด เนื่องจากข้อมูลที่เลือกมาใช้ในการทดลองของงานวิจัยนี้จะเก็บอยู่ในรูป Html และ Xml file ดังนั้นข้อมูลที่ได้มาจะมี Tag ต่าง ๆ ติดมากับข้อมูลด้วย สิ่งที่ต้องทำในขั้นตอนนี้คือ ตัด Tag ต่าง ๆ ทิ้งเพื่อให้เหลือแต่ตัวข้อความ

หรือเนื้อข่าวของเอกสารแต่ละชุด ในรูปตัวอย่างที่ 4.1 จะต้องตัด Tag ต่าง ๆ และข้อความอื่นๆ ที่ไม่เกี่ยวข้องเช่นชื่อเรื่อง (Title) ชื่อผู้แต่ง (Author) ทิ้งให้เหลือแต่เนื้อความที่ต้องการ คือข้อความตั้งแต่ <BLOCKQUOTE> Tag จนถึง </BLOCKQUOTE> Tag และเช่นกันในรูปที่ 4.3 ก็ต้องใช้ข้อความตั้งแต่ <!-- TextStart --> จนถึง <!-- TextEnd --> และในรูปที่ 4.4 ต้องใช้ข้อความตั้งแต่ <MESSAGE> จนถึง </MESSAGE>

2. นำเอกสารที่ตัด Tag ต่าง ๆ ทิ้งแล้วมาหา Keyword ของเอกสารนั้น ๆ โดยใช้โปรแกรม Copernic Summarizer โดยให้หา Keyword ของแต่ละเอกสารให้มากที่สุดเท่าที่โปรแกรมจะหาได้ แสดงตัวอย่างเอกสารที่ หา Keyword โดยใช้ โปรแกรม Copernic Summarizer ดังรูปที่ 4.5

5.txt

file://C:\MATLAB6p5\work\yuki\DocumentPreprocessing\k-dataset\k-dataset\category\E\5.txt

Concepts:

legion, awards, Chevalier, Jean-Marie Poire, Nana Mouskouri, Officiers, singer Charles Aznavour, Writer/director Lars Schmidt, Commandeur, violinist Stephane Grappelli, jazz violinist Stephane, distributing legion, entertainment business, President Jacques Chirac, French President Jacques, Variety, PARIS, Michael Williams.

Summary:

- PARIS (Variety) - French President Jacques Chirac honored the entertainment business on Thursday, distributing Legion d'Honneur awards to some top international names.

Summarized by Copernic Summarizer

รูปที่ 4.5 แสดงตัวอย่างการหา Keyword ของเอกสารจากโปรแกรม Copernic Summarizer

จากรูปที่ 4.5 แสดงตัวอย่างการหา Keyword ของเอกสารจากโปรแกรม Copernic Summarizer โดยไฟล์ที่ได้จะเป็นไฟล์ข้อความ (Text file) โดย Keyword ของเอกสารนี้คือส่วนของข้อความใน Concepts ซึ่ง Keyword แต่ละ Keyword จะถูกขั้นด้วยเครื่องหมาย Comma (,) และในส่วน Summary คือการสรุปเนื้อหาของเอกสารนี้

3. ทำการเตรียม Keyword ที่ได้ก่อนโดยการวิเคราะห์ Keyword ดังนี้ [10]

- ทำอักษรตัวใหญ่ให้เป็นตัวเล็กทั้งหมด เช่น Officers เป็น officers เป็นต้น

- ตัดเครื่องหมาย Hyphens แล้วแยกคำนั้นออก เช่น jean-marie เป็น jean และ marie เป็นต้น
- ตัดช่องว่างระหว่างคำ (Space) แล้วแยกคำนั้นออก เช่น entertainment business เป็น entertainment และ business เป็นต้น
- ตัด Keyword คำที่เป็นตัวเลขทั้งหมดทิ้ง เช่น 1998 แต่ถ้าเป็นคำที่มีอักษรปนกับตัวเลขไม่ต้องตัดทิ้ง เช่น 510B.C. เป็นต้น

4. นำ Keyword ของแต่ละเอกสารมาตัด Stop word ทิ้ง

5. นำกลุ่ม Keyword ที่ผ่านการตัด Stop word แล้วมาตัด Stemming word โดยใช้ Porter's Algorithm [10] ในการตัด Stemming word

4.2 การทดลอง และผลการทดลอง

การทดลองของงานวิจัยนี้ ได้ทำการทดลองกับชุดข้อมูลทั้งหมด 4 ชุดข้อมูลดังที่กล่าวมาแล้วข้างต้น แต่ละชุดข้อมูลนำมาทดลองกับอัลกอริทึม ARC-BC และอัลกอริทึม ARTC โดยการทดลองของชุดข้อมูลแต่ละชุดและแต่ละอัลกอริทึมนั้นได้กำหนดค่าพารามิเตอร์ต่าง ๆ ดังนี้

1. ทดลองกับข้อมูลแต่ละชุดด้วยค่าสนับสนุนน้อยที่สุดจำนวน 5 ค่า ซึ่งได้กำหนดในแต่ละชุดข้อมูล
2. ค่า Dominant factor ของอัลกอริทึม ARC-BC ที่ใช้ในการจำแนกเอกสารใหม่ (Classifying new document) ได้ทำการทดลอง 21 ครั้งในแต่ละชุดข้อมูลและในแต่ละค่าสนับสนุนน้อยที่สุดโดยกำหนดค่าที่ใช้ทดลองดังนี้

$$\text{Dominant factor} = \{0, 0.05, 0.10, 0.15, 0.20, 0.25, \ldots, 1.00\}$$

3. ค่า Limit confidence ของอัลกอริทึม ARTC ที่ใช้ในการคัดกฎความสัมพันธ์ทิ้ง ได้ทำการทดลอง 21 ครั้งแต่ละชุดข้อมูลและในแต่ละค่าสนับสนุนน้อยที่สุดโดยกำหนดค่าที่ใช้ทดลองดังนี้

$$\text{Limit confidence} = \{0, 0.05, 0.10, 0.15, 0.20, 0.25, \ldots, 1.00\}$$

ในการทดลองและเก็บค่าผลการทดลองของแต่ละอัลกอริทึมจะทำเหมือนกันในทุก ๆ ชุดข้อมูล ซึ่งอธิบายได้ดังนี้

อัลกอริทึม ARC-BC

ในแต่ละค่าสนับสนุนน้อยที่สุด ได้ทดลองในแต่ละค่า Dominant factor ทั้งหมด 21 ค่า ตั้งแต่ 0-1 โดยเพิ่มค่าเข้าไปครั้งละ 0.05 ในแต่ละค่าสนับสนุนน้อยที่สุดจะเก็บจำนวน Frequent Itemset และจำนวนกฎที่ใช้ในการจำแนกเอกสาร (Applicable rule) ของการใช้ค่าสนับสนุนน้อยที่สุดนั้น และในแต่ละการทดลองของแต่ละค่า Dominant factor จะเก็บผลการจำแนกเอกสาร โดยเก็บจำนวนเอกสารที่จำแนกประเภทได้ถูกต้อง (Correct) เก็บจำนวนเอกสารที่จำแนกไม่ถูกต้อง (Incorrect) โดยแบ่งเป็นเอกสารที่จำแนกไม่ถูกกลุ่ม (Miss classification) และเอกสารที่ไม่สามารถจำแนกได้ว่าอยู่กลุ่มใด (Unclassification) และสุดท้ายเก็บค่าความถูกต้อง (Accuracy) ในการจำแนกประเภทเอกสารของแต่ละค่า Dominant factor

อัลกอริทึม ARTC

ในแต่ละค่าสนับสนุนน้อยที่สุด ได้ทดลองในแต่ละค่า Limit confidence ทั้งหมด 21 ค่า ตั้งแต่ 0-1 โดยเพิ่มค่าเข้าไปครั้งละ 0.05 ในแต่ละค่าสนับสนุนน้อยที่สุดจะเก็บจำนวน Frequent Itemset และในแต่ละการทดลองของแต่ละค่า Limit confidence จะเก็บผลการจำแนกเอกสาร โดยเก็บจำนวนเอกสารที่จำแนกประเภทได้ถูกต้อง (Correct) เก็บจำนวนเอกสารที่จำแนกไม่ถูกต้อง (Incorrect) โดยแบ่งเป็นเอกสารที่จำแนกไม่ถูกกลุ่ม (Miss classification) และเอกสารที่ไม่สามารถจำแนกได้ว่าอยู่กลุ่มใด (Unclassification) เก็บค่าความถูกต้อง (Accuracy) ในการจำแนกประเภทเอกสารของแต่ละค่า Dominant factor และสุดท้ายเก็บกฎที่ใช้ในการจำแนกเอกสาร (Applicable rule) ในแต่ละค่า Dominant factor

เมื่อได้ผลการทดลองในแต่ละค่าสนับสนุนน้อยที่สุดแล้ว นำผลที่ได้มาเปรียบเทียบกันระหว่าง 2 อัลกอริทึมนี้ โดยเปรียบเทียบส่วนของจำนวน Frequent Itemset จำนวนกฎที่ใช้ในการจำแนกเอกสาร (Applicable rule) และประสิทธิภาพ (Accuracy rate) ในการจำแนกเอกสาร

4.2.1 ชุดข้อมูลข้อความ

ชุดข้อมูลข้อความที่สร้างขึ้นมีจำนวน 2000 เอกสาร แบ่งข้อมูลสำหรับการเรียนรู้ถึงแบบแผนของข้อมูล (Training) และทดสอบระบบ (Testing) แสดงดังตารางที่ 4.6

ตารางที่ 4.6 รายละเอียดข้อมูล Training และ Testing ของชุดข้อมูลข้อความ

ลำดับ	ชื่อกลุ่ม	Training	Testing
1	Category1	547	137
2	Category2	545	137
3	Category3	507	127
รวม		1599	401

ในการทดลองชุดข้อมูลข้อความ กับอัลกอริทึม ARC-BC และ ARTC นั้น ได้ทำการทดลอง 5 ครั้ง โดยกำหนดค่าสนับสนุนน้อยที่สุดคือ 100 120 130 140 และ 50 ตามลำดับ แสดงผลการทดลองของแต่ละอัลกอริทึมและแต่ละค่าพารามิเตอร์ดังต่อไปนี้

4.2.1.1 ผลการทดลองชุดข้อมูลข้อความ โดยของอัลกอริทึม ARC-BC

ตารางที่ 4.7 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 100

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	180	221	0	221	0.4489
0.05	180	221	0	221	0.4489
0.10	180	221	0	221	0.4489
0.15	180	221	0	221	0.4489
0.20	180	221	0	221	0.4489
0.25	180	221	0	221	0.4489
0.30	180	221	0	221	0.4489
0.35	180	221	0	221	0.4489
0.40	180	221	0	221	0.4489
0.45	180	221	0	221	0.4489
0.50	298	103	0	103	0.7431
0.55	400	0	1	1	0.9975
0.60	395	0	6	6	0.9850
0.65	362	0	39	39	0.9027
0.70	309	0	92	92	0.7706
0.75	237	0	164	164	0.5910
0.80	211	0	190	190	0.5262
0.85	182	0	219	219	0.4539
0.90	180	0	221	221	0.4489
0.95	180	0	221	221	0.4489
1.00	180	0	221	221	0.4489

ตารางที่ 4.8 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 120

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	180	221	0	221	0.4489
0.05	180	221	0	221	0.4489
0.10	180	221	0	221	0.4489
0.15	180	221	0	221	0.4489
0.20	180	221	0	221	0.4489
0.25	180	221	0	221	0.4489
0.30	180	221	0	221	0.4489
0.35	180	221	0	221	0.4489
0.40	180	221	0	221	0.4489
0.45	180	221	0	221	0.4489
0.50	298	103	0	103	0.7431
0.55	400	0	1	1	0.9975
0.60	395	0	6	6	0.9850
0.65	362	0	39	39	0.9027
0.70	309	0	92	92	0.7706
0.75	237	0	164	164	0.5910
0.80	211	0	190	190	0.5262
0.85	182	0	219	219	0.4539
0.90	180	0	221	221	0.4489
0.95	180	0	221	221	0.4489
1.00	180	0	221	221	0.4489

ตารางที่ 4.9 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 140

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	180	221	0	221	0.4489
0.05	180	221	0	221	0.4489
0.10	180	221	0	221	0.4489
0.15	180	221	0	221	0.4489
0.20	180	221	0	221	0.4489
0.25	180	221	0	221	0.4489
0.30	180	221	0	221	0.4489
0.35	180	221	0	221	0.4489
0.40	180	221	0	221	0.4489
0.45	180	221	0	221	0.4489
0.50	298	103	0	103	0.7431
0.55	400	0	1	1	0.9975
0.60	395	0	6	6	0.9850
0.65	362	0	39	39	0.9027
0.70	309	0	92	92	0.7706
0.75	237	0	164	164	0.5910
0.80	211	0	190	190	0.5262
0.85	182	0	219	219	0.4539
0.90	180	0	221	221	0.4489
0.95	180	0	221	221	0.4489
1.00	180	0	221	221	0.4489

ตารางที่ 4.10 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 160

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	180	221	0	221	0.4489
0.05	180	221	0	221	0.4489
0.10	180	221	0	221	0.4489
0.15	180	221	0	221	0.4489
0.20	180	221	0	221	0.4489
0.25	180	221	0	221	0.4489
0.30	180	221	0	221	0.4489
0.35	180	221	0	221	0.4489
0.40	180	221	0	221	0.4489
0.45	180	221	0	221	0.4489
0.50	298	103	0	103	0.7431
0.55	400	0	1	1	0.9975
0.60	395	0	6	6	0.9850
0.65	362	0	39	39	0.9027
0.70	309	0	92	92	0.7706
0.75	237	0	164	164	0.5910
0.80	211	0	190	190	0.5262
0.85	182	0	219	219	0.4539
0.90	180	0	221	221	0.4489
0.95	180	0	221	221	0.4489
1.00	180	0	221	221	0.4489

ตารางที่ 4.11 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 180

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	180	221	0	221	0.4489
0.05	180	221	0	221	0.4489
0.10	180	221	0	221	0.4489
0.15	180	221	0	221	0.4489
0.20	180	221	0	221	0.4489
0.25	180	221	0	221	0.4489
0.30	180	221	0	221	0.4489
0.35	180	221	0	221	0.4489
0.40	180	221	0	221	0.4489
0.45	180	221	0	221	0.4489
0.50	298	103	0	103	0.7431
0.55	400	0	1	1	0.9975
0.60	395	0	6	6	0.9850
0.65	362	0	39	39	0.9027
0.70	309	0	92	92	0.7706
0.75	237	0	164	164	0.5910
0.80	211	0	190	190	0.5262
0.85	182	0	219	219	0.4539
0.90	180	0	221	221	0.4489
0.95	180	0	221	221	0.4489
1.00	180	0	221	221	0.4489

4.2.1.2 ผลการทดลองชุดข้อมูลข้อความ โดยของอัลกอริทึม ARTC

ตารางที่ 4.12 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 100

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	401	0	0	0	1.0000	18
0.05	401	0	0	0	1.0000	18
0.10	401	0	0	0	1.0000	18
0.15	401	0	0	0	1.0000	18
0.20	401	0	0	0	1.0000	18
0.25	401	0	0	0	1.0000	18
0.30	401	0	0	0	1.0000	18
0.35	401	0	0	0	1.0000	18
0.40	401	0	0	0	1.0000	18
0.45	401	0	0	0	1.0000	18
0.50	283	118	0	118	0.7057	58
0.55	400	1	0	1	0.9975	97
0.60	400	1	0	1	0.9975	99
0.65	400	1	0	1	0.9975	99
0.70	400	1	0	1	0.9975	99
0.75	400	1	0	1	0.9975	99
0.80	400	1	0	1	0.9975	99
0.85	400	1	0	1	0.9975	99
0.90	400	1	0	1	0.9975	99
0.95	400	1	0	1	0.9975	99
1.00	0	0	401	401	0.0000	0

ตารางที่ 4.13 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 120

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	401	0	0	0	1.0000	18
0.05	401	0	0	0	1.0000	18
0.10	401	0	0	0	1.0000	18
0.15	401	0	0	0	1.0000	18
0.20	401	0	0	0	1.0000	18
0.25	401	0	0	0	1.0000	18
0.30	401	0	0	0	1.0000	18
0.35	401	0	0	0	1.0000	18
0.40	401	0	0	0	1.0000	18
0.45	401	0	0	0	1.0000	18
0.50	283	118	0	118	0.7057	51
0.55	362	39	0	39	0.9027	88
0.60	351	50	0	50	0.8753	90
0.65	351	50	0	50	0.8753	90
0.70	351	50	0	50	0.8753	90
0.75	351	50	0	50	0.8753	90
0.80	351	50	0	50	0.8753	90
0.85	351	50	0	50	0.8753	90
0.90	351	50	0	50	0.8753	90
0.95	351	50	0	50	0.8753	90
1.00	0	0	401	401	0.0000	0

ตารางที่ 4.14 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 140

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	401	0	0	0	1.0000	18
0.05	401	0	0	0	1.0000	18
0.10	401	0	0	0	1.0000	18
0.15	401	0	0	0	1.0000	18
0.20	401	0	0	0	1.0000	18
0.25	401	0	0	0	1.0000	18
0.30	401	0	0	0	1.0000	18
0.35	401	0	0	0	1.0000	18
0.40	401	0	0	0	1.0000	18
0.45	401	0	0	0	1.0000	18
0.50	283	118	0	118	0.7057	39
0.55	329	72	0	72	0.8204	70
0.60	329	72	0	72	0.8204	70
0.65	329	72	0	72	0.8204	70
0.70	329	72	0	72	0.8204	70
0.75	329	72	0	72	0.8204	70
0.80	329	72	0	72	0.8204	70
0.85	329	72	0	72	0.8204	70
0.90	329	72	0	72	0.8204	70
0.95	329	72	0	72	0.8204	70
1.00	0	0	401	401	0.0000	0

ตารางที่ 4.15 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 160

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	401	0	0	0	1.0000	18
0.05	401	0	0	0	1.0000	18
0.10	401	0	0	0	1.0000	18
0.15	401	0	0	0	1.0000	18
0.20	401	0	0	0	1.0000	18
0.25	401	0	0	0	1.0000	18
0.30	401	0	0	0	1.0000	18
0.35	401	0	0	0	1.0000	18
0.40	401	0	0	0	1.0000	18
0.45	401	0	0	0	1.0000	18
0.50	284	117	0	117	0.7082	28
0.55	322	79	0	79	0.8030	51
0.60	322	79	0	79	0.8030	50
0.65	322	79	0	79	0.8030	50
0.70	322	79	0	79	0.8030	50
0.75	322	79	0	79	0.8030	50
0.80	322	79	0	79	0.8030	50
0.85	322	79	0	79	0.8030	50
0.90	322	79	0	79	0.8030	50
0.95	322	79	0	79	0.8030	50
1.00	0	0	401	401	0.0000	0

ตารางที่ 4.16 ผลการทดลองชุดข้อมูลข้อความ โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 180

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	401	0	0	0	1.0000	18
0.05	401	0	0	0	1.0000	18
0.10	401	0	0	0	1.0000	18
0.15	401	0	0	0	1.0000	18
0.20	401	0	0	0	1.0000	18
0.25	401	0	0	0	1.0000	18
0.30	401	0	0	0	1.0000	18
0.35	401	0	0	0	1.0000	18
0.40	401	0	0	0	1.0000	18
0.45	401	0	0	0	1.0000	18
0.50	289	112	0	112	0.7207	24
0.55	358	43	0	43	0.8928	34
0.60	358	43	0	43	0.8928	34
0.65	358	43	0	43	0.8928	34
0.70	358	43	0	43	0.8928	34
0.75	358	43	0	43	0.8928	34
0.80	358	43	0	43	0.8928	34
0.85	358	43	0	43	0.8928	34
0.90	358	43	0	43	0.8928	34
0.95	358	43	0	43	0.8928	34
1.00	0	0	401	401	0.0000	0

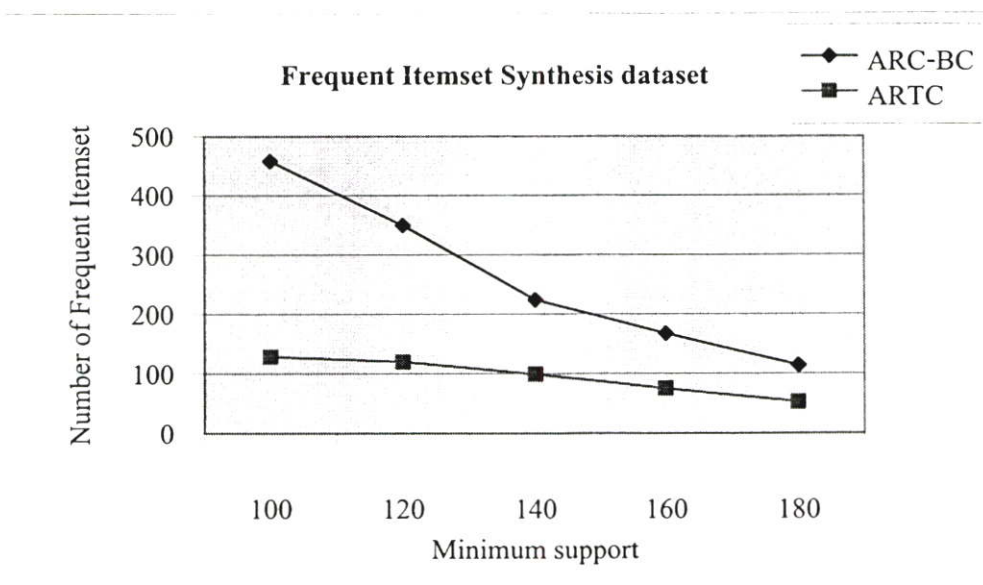
4.2.1.3 สรุปผลการทดลองชุดข้อมูลข้อความ

จากผลการทดลองของชุดข้อมูลข้อความ เลือกผลการทดลองที่ดีที่สุดของแต่ละอัลกอริทึมในแต่ละค่าสนับสนุนน้อยที่สุดที่ได้ทำการทดลอง โดยผลการทดลองที่ดีที่สุดคือผลการทดลองที่มีค่าความถูกต้องในการจำแนกประเภทเอกสาร (Accuracy rate) สูงที่สุด และกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร (Applicable rule) มีจำนวนน้อย นำผลการทดลองที่ดีที่สุดแสดงผลในตารางที่ 4.17

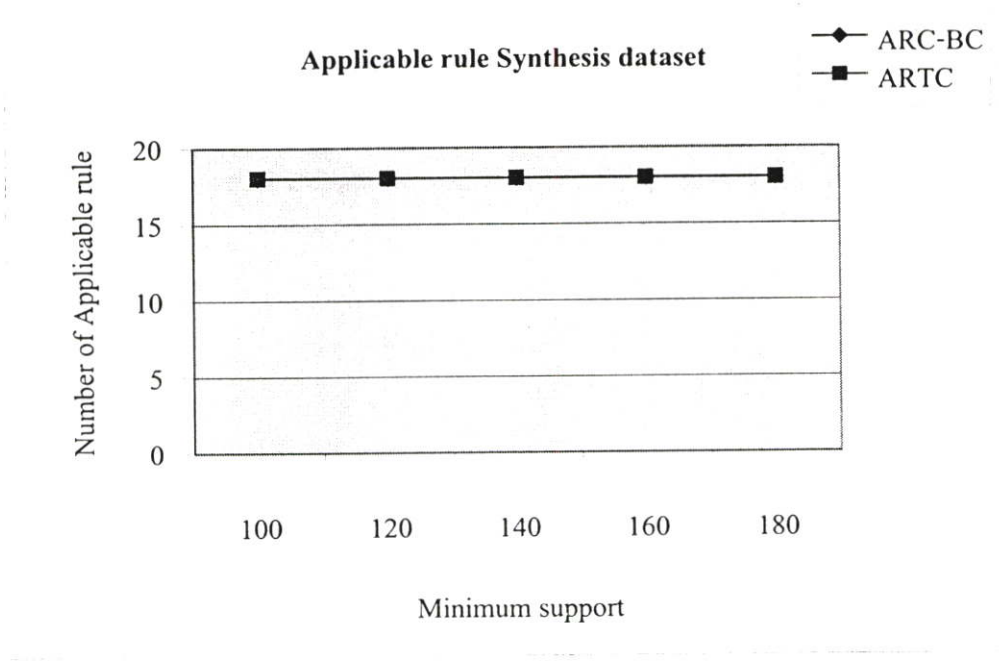
ตารางที่ 4.17 ผลการทดลองชุดข้อมูลข้อความ

Minimum	ARC-BC			ARTC		
support	Frequent Itemset	Applicable rule	Accuracy	Frequent Itemset	Applicable rule	Accuracy
100	457	18	0.9975	129	18	1.0000
120	349	18	0.9975	120	18	1.0000
140	224	18	0.9975	99	18	1.0000
160	168	18	0.9975	75	18	1.0000
180	114	18	0.9975	53	18	1.0000

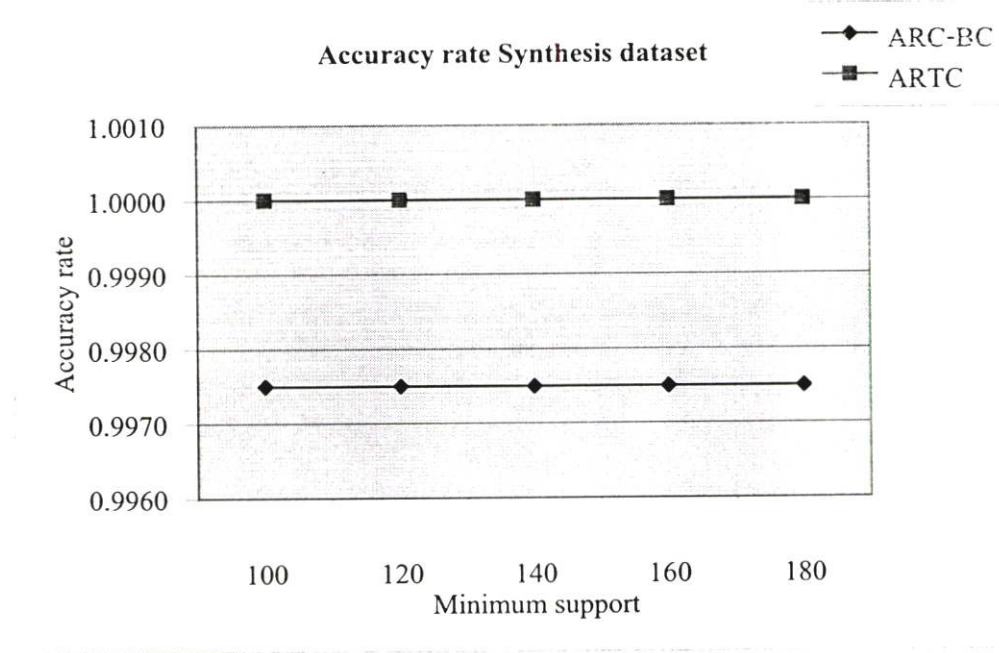
จากตารางที่ 4.17 นำข้อมูลที่ได้แสดงผลเป็นกราฟจำนวน Frequent Itemset ในรูปที่ 4.6 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารในรูปที่ 4.7 และรูปที่ 4.8 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร



รูปที่ 4.6 เปรียบเทียบจำนวน Frequent Itemset แต่ละค่าสนับสนุนน้อยที่สุดของชุดข้อมูลข้อความ



รูปที่ 4.7 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารแต่ละค่าสนับสนุนน้อยที่สุด ของชุดข้อมูลข้อความ



รูปที่ 4.8 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสารแต่ละค่าสนับสนุนน้อยที่สุด ของชุดข้อมูลข้อความ

จากการเปรียบเทียบผลการทดลองที่ได้ พบว่าในทุก ๆ ค่าสนับสนุนน้อยที่สุด อัลกอริทึม ARTC มีจำนวน Frequent Itemset น้อยกว่าอัลกอริทึม ARC-BC แสดงว่าอัลกอริทึม ARTC ค้นหาความสัมพันธ์ได้น้อยกว่าอัลกอริทึม ARC-BC เมื่อพิจารณาที่ประสิทธิภาพการจำแนกประเภทเอกสารพบว่าอัลกอริทึม ARTC มีประสิทธิภาพการจำแนกประเภทเอกสารได้ดีกว่าอัลกอริทึม ARC-BC ในจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารเท่ากัน

4.2.2 ชุดข้อมูล CSTR

ชุดข้อมูล CSTR มีจำนวน 542 เอกสาร แบ่งข้อมูล Training และ Testing แสดงดังตารางที่ 4.18

ตารางที่ 4.18 รายละเอียดข้อมูล Trainingและ Testing ของชุดข้อมูล CSTR

ลำดับ	ชื่อกลุ่ม	Training	Testing
1	AI	87	25
2	Robotics and Vision	68	20
3	Systems	152	43
4	Theory	114	33
รวม		421	121

ในการทดลองชุดข้อมูล CSTR กับอัลกอริทึม ARC-BC และ ARTC นั้น ได้ทำการทดลอง 5 ครั้ง โดยกำหนดค่าสนับสนุนน้อยที่สุดคือ 10 20 30 40 และ 50 ตามลำดับ แสดงผลการทดลองของแต่ละอัลกอริทึมและแต่ละค่าพารามิเตอร์ดังต่อไปนี้

4.2.2.1 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARC-BC

ตารางที่ 4.19 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ10

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	0	121	0	121	0.0000
0.05	0	121	0	121	0.0000
0.10	0	121	0	121	0.0000
0.15	0	121	0	121	0.0000
0.20	0	121	0	121	0.0000
0.25	0	121	0	121	0.0000
0.30	9	112	0	112	0.0744
0.35	24	97	0	97	0.1983
0.40	44	77	0	77	0.3636
0.45	70	51	0	51	0.5785
0.50	83	29	9	38	0.6860
0.55	85	15	21	36	0.7025
0.60	81	4	36	40	0.6694
0.65	64	2	55	57	0.5289
0.70	36	2	83	85	0.2975
0.75	12	0	109	109	0.0992
0.80	3	0	118	118	0.0248
0.85	0	0	121	121	0.0000
0.90	0	0	121	121	0.0000
0.95	0	0	121	121	0.0000
1.00	0	0	121	121	0.0000

ตารางที่ 4.20 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ20

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	0	121	0	121	0.0000
0.05	0	121	0	121	0.0000
0.10	0	121	0	121	0.0000
0.15	0	121	0	121	0.0000
0.20	0	121	0	121	0.0000
0.25	0	121	0	121	0.0000
0.30	4	117	0	117	0.0331
0.35	12	109	0	109	0.0992
0.40	28	93	0	93	0.2314
0.45	44	75	2	77	0.3636
0.50	57	61	3	64	0.4711
0.55	71	35	15	50	0.5868
0.60	77	20	24	44	0.6364
0.65	77	8	36	44	0.6364
0.70	48	3	70	73	0.3967
0.75	21	1	99	100	0.1736
0.80	6	0	115	115	0.0496
0.85	0	0	121	121	0.0000
0.90	0	0	121	121	0.0000
0.95	0	0	121	121	0.0000
1.00	0	0	121	121	0.0000

ตารางที่ 4.21 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ30

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	5	115	1	116	0.0413
0.05	5	115	1	116	0.0413
0.10	5	115	1	116	0.0413
0.15	5	115	1	116	0.0413
0.20	5	115	1	116	0.0413
0.25	5	115	1	116	0.0413
0.30	5	115	1	116	0.0413
0.35	7	113	1	114	0.0579
0.40	20	100	1	101	0.1653
0.45	33	87	1	88	0.2727
0.50	57	60	4	64	0.4711
0.55	69	44	8	52	0.5702
0.60	72	32	17	49	0.5950
0.65	72	24	25	49	0.5950
0.70	54	18	49	67	0.4463
0.75	36	12	73	85	0.2975
0.80	20	10	91	101	0.1653
0.85	15	10	96	106	0.1240
0.90	15	9	97	106	0.1240
0.95	5	8	108	116	0.0413
1.00	5	8	108	116	0.0413

ตารางที่ 4.22 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ40

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	27	92	2	94	0.2231
0.05	27	92	2	94	0.2231
0.10	27	92	2	94	0.2231
0.15	27	92	2	94	0.2231
0.20	27	92	2	94	0.2231
0.25	27	92	2	94	0.2231
0.30	27	92	2	94	0.2231
0.35	27	92	2	94	0.2231
0.40	27	92	2	94	0.2231
0.45	29	90	2	92	0.2397
0.50	34	85	2	87	0.2810
0.55	51	65	5	70	0.4215
0.60	69	47	5	52	0.5702
0.65	80	30	11	41	0.6612
0.70	70	23	28	51	0.5785
0.75	51	20	50	70	0.4215
0.80	43	19	59	78	0.3554
0.85	41	19	61	80	0.3388
0.90	41	18	62	80	0.3388
0.95	40	17	64	81	0.3306
1.00	27	17	77	94	0.2231

ตารางที่ 4.23 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ50

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	38	75	8	83	0.3140
0.05	38	75	8	83	0.3140
0.10	38	75	8	83	0.3140
0.15	38	75	8	83	0.3140
0.20	38	75	8	83	0.3140
0.25	38	75	8	83	0.3140
0.30	38	75	8	83	0.3140
0.35	38	75	8	83	0.3140
0.40	38	75	8	83	0.3140
0.45	40	73	8	81	0.3306
0.50	41	72	8	80	0.3388
0.55	55	53	13	66	0.4545
0.60	63	42	16	58	0.5207
0.65	67	31	23	54	0.5537
0.70	66	23	32	55	0.5455
0.75	49	23	49	72	0.4050
0.80	42	23	56	79	0.3471
0.85	38	23	60	83	0.3140
0.90	38	23	60	83	0.3140
0.95	38	23	60	83	0.3140
1.00	38	23	60	83	0.3140

4.2.2.2 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARTC

ตารางที่ 4.24 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ10

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	102	19	0	19	0.8430	394
0.05	102	19	0	19	0.8430	394
0.10	102	19	0	19	0.8430	393
0.15	104	17	0	17	0.8595	388
0.20	104	17	0	17	0.8595	383
0.25	104	17	0	17	0.8595	405
0.30	95	26	0	26	0.7851	421
0.35	90	31	0	31	0.7438	467
0.40	69	52	0	52	0.5702	542
0.45	69	52	0	52	0.5702	616
0.50	79	42	0	42	0.6529	712
0.55	70	51	0	51	0.5785	781
0.60	71	50	0	50	0.5868	893
0.65	75	46	0	46	0.6198	977
0.70	75	46	0	46	0.6198	1092
0.75	73	48	0	48	0.6033	1210
0.80	73	48	0	48	0.6033	1280
0.85	72	49	0	49	0.5950	1289
0.90	72	49	0	49	0.5950	1290
0.95	70	51	0	51	0.5785	1189
1.00	0	0	121	121	0.0000	0

ตารางที่ 4.25 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ20

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	91	30	0	30	0.7521	127
0.05	91	30	0	30	0.7521	127
0.10	91	30	0	30	0.7521	127
0.15	91	30	0	30	0.7521	127
0.20	90	31	0	31	0.7438	123
0.25	89	32	0	32	0.7355	119
0.30	88	33	0	33	0.7273	116
0.35	87	34	0	34	0.7190	112
0.40	85	36	0	36	0.7025	119
0.45	84	37	0	37	0.6942	124
0.50	78	43	0	43	0.6446	130
0.55	78	43	0	43	0.6446	135
0.60	77	44	0	44	0.6364	149
0.65	76	45	0	45	0.6281	154
0.70	78	43	0	43	0.6446	144
0.75	76	45	0	45	0.6281	134
0.80	77	43	1	44	0.6364	113
0.85	73	45	3	48	0.6033	99
0.90	74	44	3	47	0.6116	89
0.95	65	53	3	56	0.5372	72
1.00	0	0	121	121	0.0000	0

ตารางที่ 4.26 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ30

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	78	41	2	43	0.6446	58
0.05	78	41	2	43	0.6446	58
0.10	78	41	2	43	0.6446	58
0.15	78	41	2	43	0.6446	58
0.20	78	41	2	43	0.6446	58
0.25	78	41	2	43	0.6446	58
0.30	78	41	2	43	0.6446	58
0.35	77	42	2	44	0.6364	56
0.40	78	40	3	43	0.6446	55
0.45	79	38	4	42	0.6529	51
0.50	79	38	4	42	0.6529	49
0.55	80	37	4	41	0.6612	48
0.60	81	36	4	40	0.6694	47
0.65	76	41	4	45	0.6281	50
0.70	77	39	5	44	0.6364	46
0.75	77	38	6	44	0.6364	39
0.80	81	28	12	40	0.6694	32
0.85	81	28	12	40	0.6694	31
0.90	81	25	15	40	0.6694	29
0.95	78	16	27	43	0.6446	21
1.00	0	0	121	121	0.0000	0

ตารางที่ 4.27 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ40

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	74	44	3	47	0.6116	30
0.05	74	44	3	47	0.6116	30
0.10	74	44	3	47	0.6116	30
0.15	74	44	3	47	0.6116	30
0.20	74	44	3	47	0.6116	30
0.25	74	44	3	47	0.6116	30
0.30	74	44	3	47	0.6116	30
0.35	74	44	3	47	0.6116	30
0.40	74	44	3	47	0.6116	30
0.45	76	40	5	45	0.6281	29
0.50	78	33	10	43	0.6446	27
0.55	77	30	14	44	0.6364	26
0.60	84	23	14	37	0.6942	22
0.65	82	24	15	39	0.6777	20
0.70	82	19	20	39	0.6777	17
0.75	80	16	25	41	0.6612	13
0.80	71	9	41	50	0.5868	9
0.85	63	10	48	58	0.5207	8
0.90	59	9	53	62	0.4876	7
0.95	55	10	56	66	0.4545	5
1.00	0	0	121	121	0.0000	0

ตารางที่ 4.28 ผลการทดลองชุดข้อมูล CSTR โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ50

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	72	41	8	49	0.5950	16
0.05	72	41	8	49	0.5950	16
0.10	72	41	8	49	0.5950	16
0.15	72	41	8	49	0.5950	16
0.20	72	41	8	49	0.5950	16
0.25	72	41	8	49	0.5950	16
0.30	72	41	8	49	0.5950	16
0.35	72	41	8	49	0.5950	16
0.40	72	41	8	49	0.5950	16
0.45	74	36	11	47	0.6116	15
0.50	74	28	19	47	0.6116	14
0.55	70	25	26	51	0.5785	13
0.60	72	21	28	49	0.5950	12
0.65	68	19	34	53	0.5620	10
0.70	57	12	52	64	0.4711	7
0.75	48	11	62	73	0.3967	5
0.80	38	11	72	83	0.3140	4
0.85	22	10	89	99	0.1818	3
0.90	22	10	89	99	0.1818	3
0.95	22	10	89	99	0.1818	3
1.00	0	0	121	121	0.0000	0

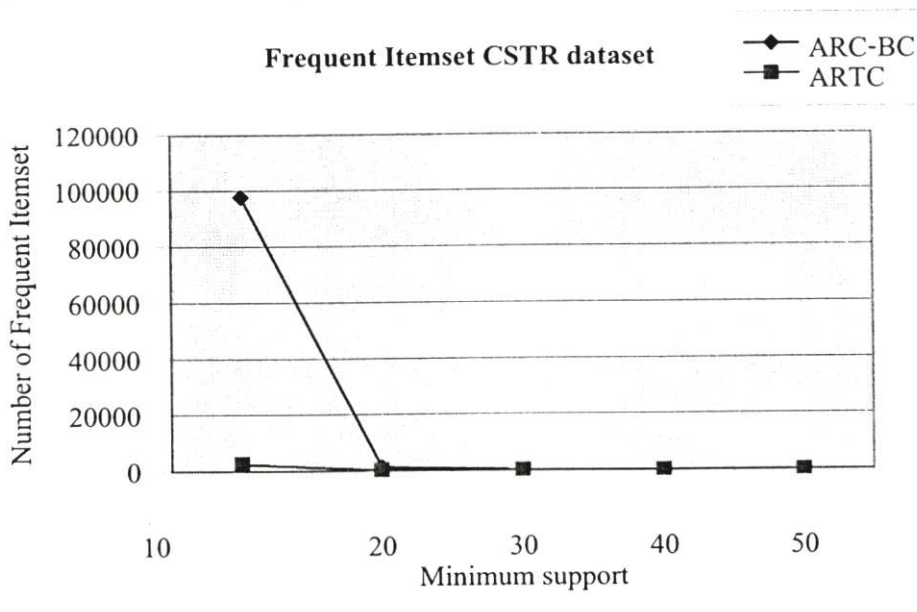
4.2.2.3 สรุปผลการทดลอง ชุดข้อมูล CSTR

จากผลการทดลองของชุดข้อมูล CSTR เลือกผลการทดลองที่ดีที่สุดของแต่ละอัลกอริทึมในแต่ละค่าสนับสนุนน้อยที่สุดที่ได้ทำการทดลอง โดยผลการทดลองที่ดีที่สุดคือผลการทดลองที่มีค่าความถูกต้องในการจำแนกประเภทเอกสาร (Accuracy rate) สูงที่สุด และกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร (Applicable rule) มีจำนวนน้อย นำผลการทดลองที่ดีที่สุดแสดงผลในตารางที่ 4.29

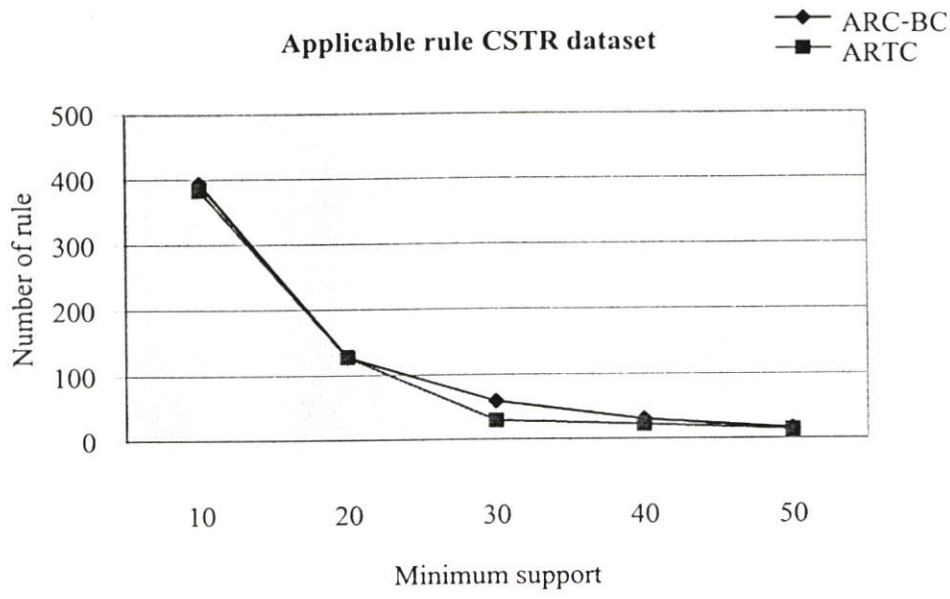
ตารางที่ 4.29 ผลการทดลองชุดข้อมูล CSTR

Minimum	ARC-BC			ARTC		
support	Frequent Itemset	Applicable rule	Accuracy	Frequent Itemset	Applicable rule	Accuracy
10	97664	394	0.7024	2546	383	0.8595
20	1242	127	0.6364	224	127	0.7521
30	169	59	0.5950	73	29	0.6694
40	44	30	0.6616	30	22	0.6942
50	18	16	0.5537	16	14	0.6116

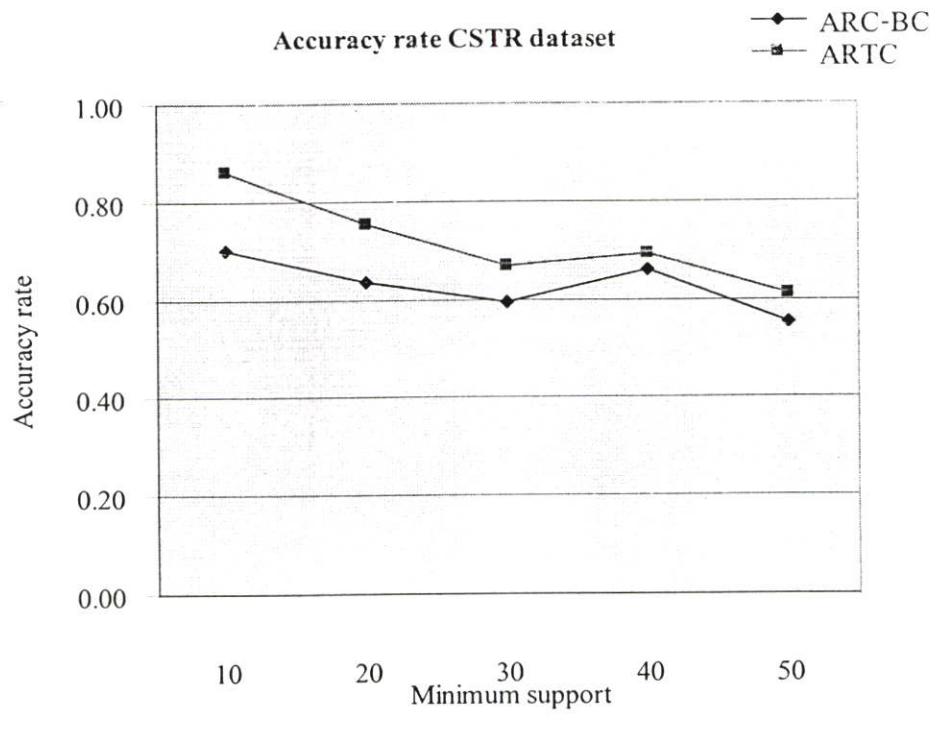
จากตารางที่ 4.29 นำข้อมูลที่ได้แสดงผลเป็นกราฟจำนวน Frequent Itemsetในรูปที่ 4.9 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารในรูปที่ 4.10 และรูปที่ 4.11 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร



รูปที่ 4.9 เปรียบเทียบจำนวน Frequent Itemset แต่ละค่าสนับสนุนน้อยที่สุดของชุดข้อมูล CSTR



รูปที่ 4.10 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร แต่ละค่าสนับสนุนน้อยที่สุด ของชุดข้อมูล CSTR



รูปที่ 4.11 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร แต่ละค่าสนับสนุนน้อยที่สุด ของชุดข้อมูล CSTR

จากการเปรียบเทียบผลการทดลองที่ได้ พบว่าในทุก ๆ ค่าสนับสนุนน้อยที่สุด อัลกอริทึม ARTC มีจำนวน Frequent Itemset น้อยกว่าอัลกอริทึม ARC-BC แสดงว่าอัลกอริทึม ARTC ค้นหาความสัมพันธ์ได้น้อยกว่าอัลกอริทึม ARC-BC เมื่อพิจารณาที่ประสิทธิภาพการจำแนกประเภทเอกสารพบว่าอัลกอริทึม ARTC มีประสิทธิภาพการจำแนกประเภทเอกสารได้ดีกว่าอัลกอริทึม ARC-BC ในจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารน้อยกว่าอัลกอริทึม ARC-BC ในทุก ๆ ค่าสนับสนุนน้อยที่สุด ยกเว้นที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 20 ที่ใช้จำนวนกฎความสัมพันธ์เท่ากัน แต่อย่างไรก็ตามประสิทธิภาพการจำแนกประเภทเอกสารของอัลกอริทึม ARTC ก็ยังดีกว่าอัลกอริทึม ARC-BC

4.2.3 ชุดข้อมูล K-dataset

ชุดข้อมูล K-dataset มีจำนวน 2340 เอกสาร แบ่งข้อมูล Training และ Testing แสดงดังตารางที่ 4.30

ในการทดลองชุดข้อมูล K-dataset กับอัลกอริทึม ARTC และ ARC-BC นั้น ได้ทำการทดลอง 5 ครั้ง โดยกำหนดค่าสนับสนุนน้อยที่สุดคือ 20 30 40 50 และ 60 ตามลำดับ แสดงผลการทดลองของแต่ละอัลกอริทึมและแต่ละค่าพารามิเตอร์ดังต่อไปนี้

ตารางที่ 4.30 รายละเอียดข้อมูล Training และ Testing ของชุดข้อมูล K-dataset

ลำดับ	ชื่อกลุ่ม	Training	Testing
1	Business	113	29
2	Entertainment	7	2
3	Art	19	5
4	Cable	35	9
5	Culture	59	15
6	Film	222	56
7	Industry	56	14
8	Media	16	5
9	Multimedia	11	3
10	Music	100	25
11	Online	52	13
12	People	198	50
13	Review	126	32
14	Stage	14	4
15	Television	149	38
16	Variety	43	11
17	Health	395	99
18	Politics	91	23
19	Sports	112	29
20	Technology	48	12
รวม		1866	474

4.2.3.1 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARC-BC

ตารางที่ 4.31 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARC-BC ค่าสนับสนุนน้อยที่สุด เท่ากับ 20

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	2	472	0	472	0.00421941
0.05	2	472	0	472	0.00421941
0.10	2	472	0	472	0.00421941
0.15	7	467	0	467	0.01476793
0.20	16	458	0	458	0.03375527
0.25	34	438	2	440	0.07172996
0.30	74	387	13	400	0.15611814
0.35	129	314	31	345	0.2721519
0.40	164	252	58	310	0.34599156
0.45	186	187	101	288	0.39240506
0.50	188	143	143	286	0.39662447
0.55	161	109	204	313	0.33966245
0.60	138	75	261	336	0.29113924
0.65	117	57	300	357	0.24683544
0.70	93	35	346	381	0.19620253
0.75	53	23	398	421	0.11181435
0.80	28	15	431	446	0.05907173
0.85	8	11	455	466	0.01687764
0.90	5	3	466	469	0.01054852
0.95	3	0	471	471	0.00632911
1.00	2	0	472	472	0.00421941

ตารางที่ 4.32 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARC-BC ค่าสนับสนุนน้อยที่สุด เท่ากับ 30

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	16	458	0	458	0.0338
0.05	16	458	0	458	0.0338
0.10	16	458	0	458	0.0338
0.15	16	458	0	458	0.0338
0.20	31	443	0	443	0.0654
0.25	62	409	3	412	0.1308
0.30	100	365	9	374	0.2110
0.35	138	309	27	336	0.2911
0.40	174	248	52	300	0.3671
0.45	181	203	90	293	0.3819
0.50	182	158	134	292	0.3840
0.55	175	112	187	299	0.3692
0.60	163	83	228	311	0.3439
0.65	147	67	260	327	0.3101
0.70	119	42	313	355	0.2511
0.75	80	29	365	394	0.1688
0.80	55	20	399	419	0.1160
0.85	35	13	426	439	0.0738
0.90	25	7	442	449	0.0527
0.95	24	4	446	450	0.0506
1.00	16	2	456	458	0.0338

ตารางที่ 4.33 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARC-BC ค่าสนับสนุนน้อยที่สุด เท่ากับ 40

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	54	415	5	420	0.1139
0.05	54	415	5	420	0.1139
0.10	54	415	5	420	0.1139
0.15	54	415	5	420	0.1139
0.20	73	395	6	401	0.1540
0.25	94	368	12	380	0.1983
0.30	120	341	13	354	0.2532
0.35	160	285	29	314	0.3376
0.40	192	210	72	282	0.4051
0.45	198	174	102	276	0.4177
0.50	188	151	135	286	0.3966
0.55	181	124	169	293	0.3819
0.60	177	94	203	297	0.3734
0.65	160	81	233	314	0.3376
0.70	141	58	275	333	0.2975
0.75	112	41	321	362	0.2363
0.80	92	30	352	382	0.1941
0.85	78	21	375	396	0.1646
0.90	74	10	390	400	0.1561
0.95	72	7	395	402	0.1519
1.00	54	5	415	420	0.1139

ตารางที่ 4.34 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARC-BC ค่าสนับสนุนน้อยที่สุดเท่ากับ 50

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	93	368	13	381	0.1962
0.05	93	368	13	381	0.1962
0.10	93	368	13	381	0.1962
0.15	93	368	13	381	0.1962
0.20	104	357	13	370	0.2194
0.25	130	328	16	344	0.2743
0.30	153	303	18	321	0.3228
0.35	171	270	33	303	0.3608
0.40	207	197	70	267	0.4367
0.45	210	177	87	264	0.4430
0.50	203	162	109	271	0.4283
0.55	188	149	137	286	0.3966
0.60	176	122	176	298	0.3713
0.65	170	111	193	304	0.3586
0.70	168	96	210	306	0.3544
0.75	174	79	221	300	0.3671
0.80	148	63	263	326	0.3122
0.85	133	53	288	341	0.2806
0.90	130	41	303	344	0.2743
0.95	109	38	327	365	0.2300
1.00	93	36	345	381	0.1962

ตารางที่ 4.35 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARC-BC ค่าสนับสนุนน้อยที่สุดเท่ากับ 60

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	117	328	29	357	0.2468
0.05	117	328	29	357	0.2468
0.10	117	328	29	357	0.2468
0.15	117	328	29	357	0.2468
0.20	129	316	29	345	0.2722
0.25	162	280	32	312	0.3418
0.30	163	279	32	311	0.3439
0.35	176	258	40	298	0.3713
0.40	194	189	91	280	0.4093
0.45	194	166	114	280	0.4093
0.50	195	155	124	279	0.4114
0.55	173	141	160	301	0.3650
0.60	169	134	171	305	0.3565
0.65	170	131	173	304	0.3586
0.70	168	122	184	306	0.3544
0.75	175	105	194	299	0.3692
0.80	155	93	226	319	0.3270
0.85	154	84	236	320	0.3249
0.90	150	74	250	324	0.3165
0.95	136	72	266	338	0.2869
1.00	117	71	286	357	0.2468

4.2.3.2 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARTC

ตารางที่ 4.36 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 20

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	359	115	0	115	0.7574	481
0.05	359	115	0	115	0.7574	481
0.10	369	105	0	105	0.7785	472
0.15	373	101	0	101	0.7869	462
0.20	347	127	0	127	0.7321	448
0.25	297	177	0	177	0.6266	456
0.30	293	181	0	181	0.6181	456
0.35	300	174	0	174	0.6329	460
0.40	280	194	0	194	0.5907	470
0.45	253	221	0	221	0.5338	493
0.50	257	217	0	217	0.5422	466
0.55	245	229	0	229	0.5169	457
0.60	245	229	0	229	0.5169	435
0.65	246	227	1	228	0.5190	413
0.70	241	232	1	233	0.5084	387
0.75	242	231	1	232	0.5105	359
0.80	240	232	2	234	0.5063	323
0.85	241	231	2	233	0.5084	289
0.90	237	235	2	237	0.5000	241
0.95	241	230	3	233	0.5084	193
1.00	0	0	474	474	0.0000	0

ตารางที่ 4.37 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 30

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	293	180	1	181	0.6181	200
0.05	293	180	1	181	0.6181	200
0.10	303	170	1	171	0.6392	199
0.15	298	175	1	176	0.6287	196
0.20	297	176	1	177	0.6266	185
0.25	298	174	2	176	0.6287	175
0.30	300	171	3	174	0.6329	164
0.35	274	197	3	200	0.5781	159
0.40	264	206	4	210	0.5570	159
0.45	242	227	5	232	0.5105	163
0.50	243	225	6	231	0.5127	152
0.55	241	222	11	233	0.5084	141
0.60	242	217	15	232	0.5105	128
0.65	245	213	16	229	0.5169	119
0.70	239	215	20	235	0.5042	110
0.75	232	210	32	242	0.4895	99
0.80	225	208	41	249	0.4747	90
0.85	224	207	43	250	0.4726	80
0.90	222	207	45	252	0.4684	65
0.95	207	205	62	267	0.4367	42
1.00	0	0	474	474	0.0000	0

ตารางที่ 4.38 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 40

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	272	190	12	202	0.5738	108
0.05	272	190	12	202	0.5738	108
0.10	272	190	12	202	0.5738	108
0.15	270	192	12	204	0.5696	107
0.20	260	201	13	214	0.5485	102
0.25	249	206	19	225	0.5253	97
0.30	245	206	23	229	0.5169	89
0.35	249	201	24	225	0.5253	85
0.40	230	214	30	244	0.4852	84
0.45	218	229	27	256	0.4599	86
0.50	216	222	36	258	0.4557	82
0.55	207	219	48	267	0.4367	77
0.60	219	186	69	255	0.4620	69
0.65	219	183	72	255	0.4620	65
0.70	214	176	84	260	0.4515	58
0.75	209	173	92	265	0.4409	54
0.80	205	168	101	269	0.4325	50
0.85	201	162	111	273	0.4241	44
0.90	213	138	123	261	0.4494	34
0.95	189	86	199	285	0.3987	18
1.00	0	0	474	474	0.0000	0

ตารางที่ 4.39 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 50

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	214	230	30	260	0.4515	64
0.05	214	230	30	260	0.4515	64
0.10	214	230	30	260	0.4515	64
0.15	214	230	30	260	0.4515	64
0.20	203	229	42	271	0.4283	62
0.25	215	217	42	259	0.4536	59
0.30	217	214	43	257	0.4578	56
0.35	224	203	47	250	0.4726	54
0.40	222	186	66	252	0.4684	50
0.45	221	192	61	253	0.4662	52
0.50	222	190	62	252	0.4684	51
0.55	223	177	74	251	0.4705	50
0.60	244	137	93	230	0.5148	45
0.65	247	132	95	227	0.5211	43
0.70	238	128	108	236	0.5021	39
0.75	236	124	114	238	0.4979	37
0.80	229	115	130	245	0.4831	33
0.85	226	104	144	248	0.4768	28
0.90	211	103	160	263	0.4451	22
0.95	185	53	236	289	0.3903	12
1.00	0	0	474	474	0.0000	0

ตารางที่ 4.40 ผลการทดลองชุดข้อมูล K-dataset โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุด เท่ากับ 60

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	178	238	58	296	0.3755	42
0.05	178	238	58	296	0.3755	42
0.10	178	238	58	296	0.3755	42
0.15	178	238	58	296	0.3755	42
0.20	174	222	78	300	0.3671	41
0.25	191	205	78	283	0.4030	38
0.30	191	205	78	283	0.4030	38
0.35	194	193	87	280	0.4093	36
0.40	193	173	108	281	0.4072	34
0.45	194	179	101	280	0.4093	35
0.50	196	171	107	278	0.4135	34
0.55	197	156	121	277	0.4156	33
0.60	196	138	140	278	0.4135	31
0.65	195	109	170	279	0.4114	30
0.70	194	110	170	280	0.4093	29
0.75	193	104	177	281	0.4072	27
0.80	189	98	187	285	0.3987	23
0.85	172	89	213	302	0.3629	18
0.90	167	76	231	307	0.3523	14
0.95	115	89	270	359	0.2426	8
1.00	0	0	474	474	0.0000	0

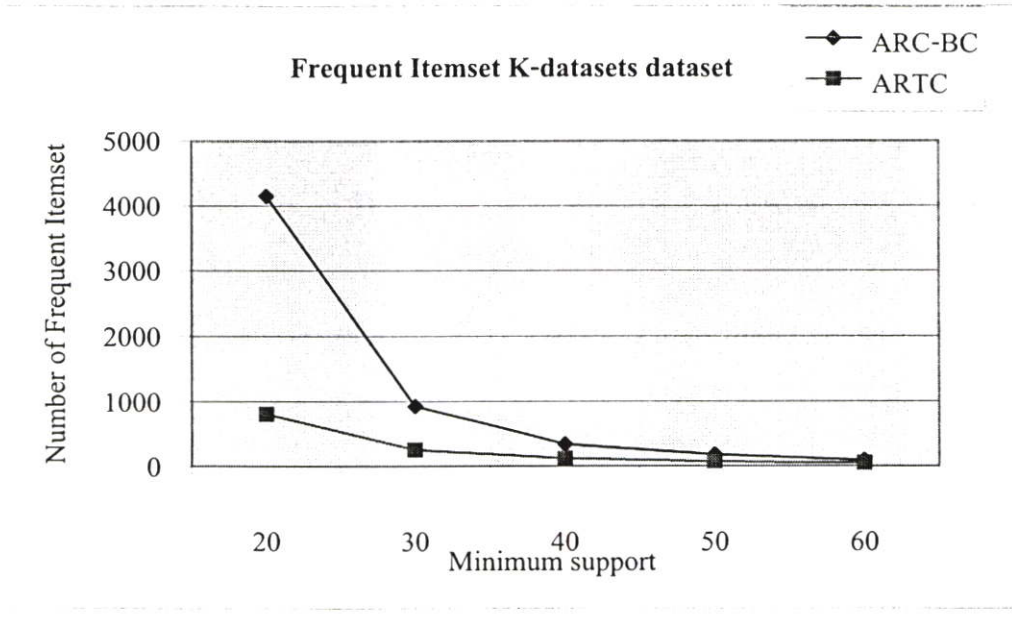
4.2.3.3 สรุปผลการทดลองชุดข้อมูล K-dataset

จากผลการทดลองของชุดข้อมูล K-dataset เลือกผลการทดลองที่ดีที่สุดของแต่ละอัลกอริทึมในแต่ละค่าสนับสนุนน้อยที่สุดที่ได้ทำการทดลอง โดยผลการทดลองที่ดีที่สุดคือผลการทดลองที่มีค่าความถูกต้องในการจำแนกประเภทเอกสาร (Accuracy rate) สูงที่สุด และกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร (Applicable rule) มีจำนวนน้อย นำผลการทดลองที่ดีที่สุดแสดงผลในตารางที่ 4.41

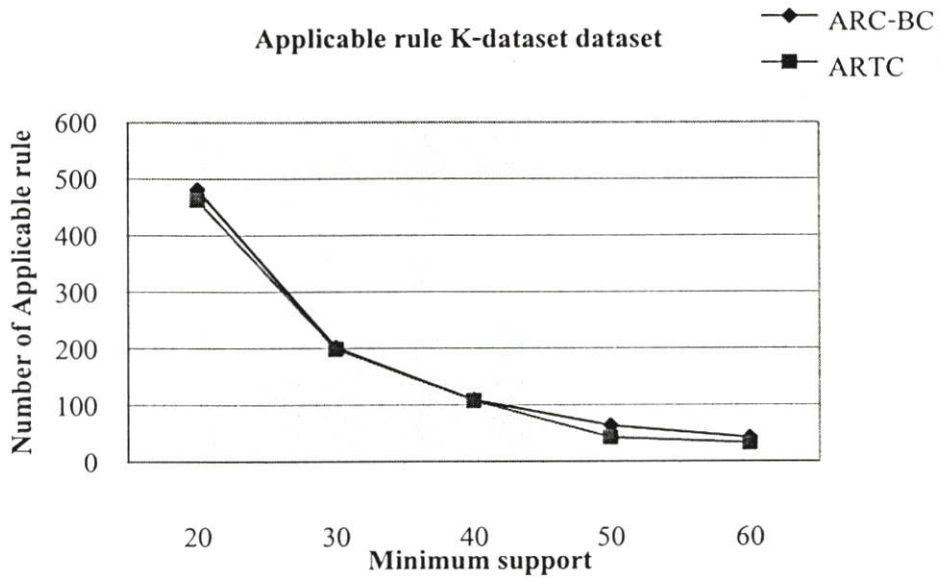
ตารางที่ 4.41 แสดงผลการทดลองชุดข้อมูล K-dataset

Minimum	ARC-BC			ARTC		
support	Frequent Itemset	Applicable rule	Accuracy	Frequent Itemset	Applicable rule	Accuracy
20	4149	481	0.3966	801	462	0.7869
30	926	202	0.3840	256	199	0.6392
40	342	109	0.4177	126	108	0.5738
50	181	64	0.4430	72	43	0.5211
60	92	42	0.4114	47	33	0.4156

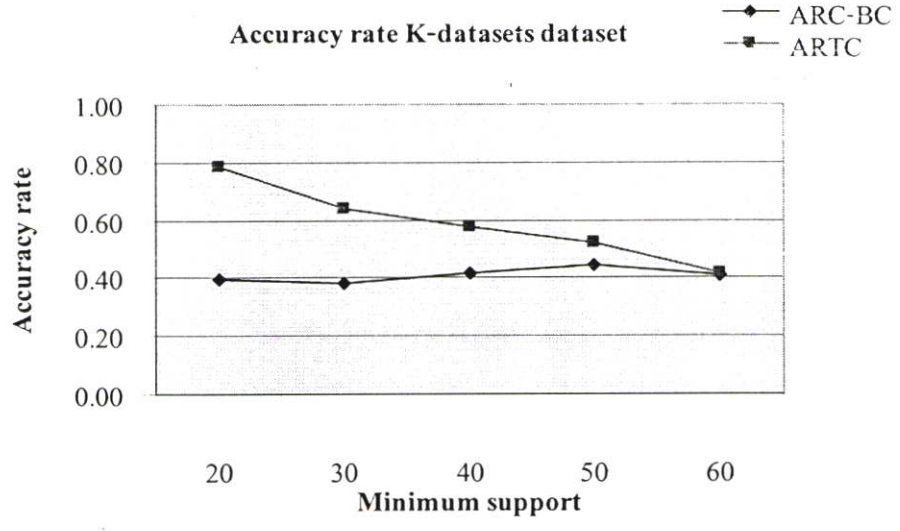
จากตารางที่ 4.41 นำข้อมูลที่ได้แสดงผลเป็นกราฟจำนวน Frequent Itemsetในรูปที่ 4.12 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารในรูปที่ 4.13 และรูปที่ 4.14 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร



รูปที่ 4.12 เปรียบเทียบจำนวน Frequent Itemset แต่ละค่าสนับสนุนน้อยที่สุดของชุดข้อมูล K-dataset



รูปที่ 4.13 จำนวนกฎที่ใช้ในการจำแนกเอกสารแต่ละค่าสนับสนุนน้อยที่สุด ของชุดข้อมูล K-dataset



รูปที่ 4.14 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร แต่ละค่าสนับสนุนน้อยที่สุด ของชุดข้อมูล K-dataset

จากการเปรียบเทียบผลการทดลองที่ได้ พบว่าในทุก ๆ ค่าสนับสนุนน้อยที่สุด อัลกอริทึม ARTC มีจำนวน Frequent Itemset น้อยกว่าอัลกอริทึม ARC-BC แสดงว่าอัลกอริทึม ARTC ค้นหาความสัมพันธ์ได้น้อยกว่าอัลกอริทึม ARC-BC เมื่อพิจารณาที่ประสิทธิภาพการจำแนกประเภทเอกสารพบว่าอัลกอริทึม ARTC มีประสิทธิภาพการจำแนกประเภทเอกสารได้ดีกว่าอัลกอริทึม ARC-BC ในจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารน้อยกว่าอัลกอริทึม ARC-BC ในทุก ๆ ค่าสนับสนุนน้อยที่สุด

4.2.4 ชุดข้อมูล Reuters-Top10

ชุดข้อมูล Reuters-Top10 มีจำนวน 2775 เอกสาร แบ่งข้อมูล Training และ Testing แสดงดังตารางที่ 4.42

ตารางที่ 4.42 รายละเอียดข้อมูล Trainingและ Testing ของชุดข้อมูล Reuters-Top10

ลำดับ	ชื่อกลุ่ม	Training	Testing
1	Acq	575	144
2	Corn	44	12
3	Crude	151	38
4	Earn	868	217
5	Grain	118	30
6	Interest	103	26
7	Money-fx	140	36
8	Ship	71	18
9	Trade	90	23
10	Wheat	56	15
รวม		2216	559

ในการทดลองชุดข้อมูล Reuters-Top10 กับอัลกอริทึม ARTC และ ARC-BC นั้น ได้ทำการทดลอง 5 ครั้ง โดยกำหนดค่าสนับสนุนน้อยที่สุดคือ 30 40 50 60 และ 70 ตามลำดับ แสดงผลการทดลองของแต่ละอัลกอริทึมและแต่ละค่าพารามิเตอร์ดังต่อไปนี้

4.2.4.1 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARC-BC

ตารางที่ 4.43 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 30

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	0	559	0	559	0.0000
0.05	0	559	0	559	0.0000
0.10	0	559	0	559	0.0000
0.15	0	559	0	559	0.0000
0.20	4	555	0	555	0.0072
0.25	36	523	0	523	0.0644
0.30	104	454	1	455	0.1860
0.35	208	316	35	351	0.3721
0.40	295	148	116	264	0.5277
0.45	335	42	182	224	0.5993
0.50	302	8	249	257	0.5403
0.55	246	0	313	313	0.4401
0.60	188	0	371	371	0.3363
0.65	98	0	461	461	0.1753
0.70	22	0	537	537	0.0394
0.75	2	0	557	557	0.0036
0.80	0	0	559	559	0.0000
0.85	0	0	559	559	0.0000
0.90	0	0	559	559	0.0000
0.95	0	0	559	559	0.0000
1.00	0	0	559	559	0.0000

ตารางที่ 4.44 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 40

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	0	559	0	559	0.0000
0.05	0	559	0	559	0.0000
0.10	0	559	0	559	0.0000
0.15	0	559	0	559	0.0000
0.20	0	559	0	559	0.0000
0.25	3	556	0	556	0.0054
0.30	34	518	7	525	0.0608
0.35	132	400	27	427	0.2361
0.40	249	214	96	310	0.4454
0.45	315	70	174	244	0.5635
0.50	298	15	246	261	0.5331
0.55	256	0	303	303	0.4580
0.60	200	0	359	359	0.3578
0.65	115	0	444	444	0.2057
0.70	25	0	534	534	0.0447
0.75	1	0	558	558	0.0018
0.80	0	0	559	559	0.0000
0.85	0	0	559	559	0.0000
0.90	0	0	559	559	0.0000
0.95	0	0	559	559	0.0000
1.00	0	0	559	559	0.0000

ตารางที่ 4.45 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 50

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	0	559	0	559	0.0000
0.05	0	559	0	559	0.0000
0.10	0	559	0	559	0.0000
0.15	0	559	0	559	0.0000
0.20	0	559	0	559	0.0000
0.25	1	558	0	558	0.0018
0.30	25	531	3	534	0.0447
0.35	120	415	24	439	0.2147
0.40	231	244	84	328	0.4132
0.45	306	102	151	253	0.5474
0.50	303	28	228	256	0.5420
0.55	259	0	300	300	0.4633
0.60	201	0	358	358	0.3596
0.65	133	0	426	426	0.2379
0.70	31	0	528	528	0.0555
0.75	2	0	557	557	0.0036
0.80	0	0	559	559	0.0000
0.85	0	0	559	559	0.0000
0.90	0	0	559	559	0.0000
0.95	0	0	559	559	0.0000
1.00	0	0	559	559	0.0000

ตารางที่ 4.46 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 60

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	0	559	0	559	0.0000
0.05	0	559	0	559	0.0000
0.10	0	559	0	559	0.0000
0.15	0	559	0	559	0.0000
0.20	0	559	0	559	0.0000
0.25	0	559	0	559	0.0000
0.30	22	535	2	537	0.0394
0.35	116	431	12	443	0.2075
0.40	213	274	72	346	0.3810
0.45	286	119	154	273	0.5116
0.50	304	36	219	255	0.5438
0.55	264	7	288	295	0.4723
0.60	207	2	350	352	0.3703
0.65	135	1	423	424	0.2415
0.70	33	1	525	526	0.0590
0.75	2	0	557	557	0.0036
0.80	0	0	559	559	0.0000
0.85	0	0	559	559	0.0000
0.90	0	0	559	559	0.0000
0.95	0	0	559	559	0.0000
1.00	0	0	559	559	0.0000

ตารางที่ 4.47 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARC-BC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 70

Dominant factor	Classifying new documents				
	Correct	Miss classification	Unclassification	Incorrect	Accuracy
0.00	0	559	0	559	0.0000
0.05	0	559	0	559	0.0000
0.10	0	559	0	559	0.0000
0.15	0	559	0	559	0.0000
0.20	0	559	0	559	0.0000
0.25	0	559	0	559	0.0000
0.30	17	541	1	542	0.0304
0.35	116	428	15	443	0.2075
0.40	211	276	72	348	0.3775
0.45	288	131	140	271	0.5152
0.50	297	45	217	262	0.5313
0.55	252	13	294	307	0.4508
0.60	205	3	351	354	0.3667
0.65	140	1	418	419	0.2504
0.70	37	1	521	522	0.0662
0.75	2	0	557	557	0.0036
0.80	0	0	559	559	0.0000
0.85	0	0	559	559	0.0000
0.90	0	0	559	559	0.0000
0.95	0	0	559	559	0.0000
1.00	0	0	559	559	0.0000

4.2.4.2 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARTC

ตารางที่ 4.48 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 30

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	394	165	0	165	0.7048	525
0.05	392	167	0	167	0.7013	535
0.10	306	253	0	253	0.5474	599
0.15	101	458	0	458	0.1807	614
0.20	239	320	0	320	0.4275	579
0.25	253	306	0	306	0.4526	565
0.30	186	373	0	373	0.3327	609
0.35	224	335	0	335	0.4007	602
0.40	220	339	0	339	0.3936	609
0.45	227	332	0	332	0.4061	811
0.50	220	339	0	339	0.3936	977
0.55	221	338	0	338	0.3953	1034
0.60	219	340	0	340	0.3918	1183
0.65	218	341	0	341	0.3900	1352
0.70	218	341	0	341	0.3900	1404
0.75	218	341	0	341	0.3900	1519
0.80	217	342	0	342	0.3882	1493
0.85	217	342	0	342	0.3882	1519
0.90	217	342	0	342	0.3882	1662
0.95	217	342	0	342	0.3882	2330
1.00	0	0	559	559	0.0000	0

ตารางที่ 4.49 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 40

Limit	Classifying new documents					
confidence	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	376	183	0	183	0.6726	331
0.05	381	178	0	178	0.6816	337
0.10	332	227	0	227	0.5939	352
0.15	255	304	0	304	0.4562	355
0.20	360	199	0	199	0.6440	325
0.25	343	216	0	216	0.6136	315
0.30	149	410	0	410	0.2665	439
0.35	149	410	0	410	0.2665	436
0.40	146	413	0	413	0.2612	561
0.45	307	252	0	252	0.5492	732
0.50	230	329	0	329	0.4114	850
0.55	227	332	0	332	0.4061	930
0.60	220	339	0	339	0.3936	1086
0.65	220	339	0	339	0.3936	1231
0.70	220	339	0	339	0.3936	1304
0.75	220	339	0	339	0.3936	1462
0.80	220	339	0	339	0.3936	1452
0.85	221	338	0	338	0.3953	1451
0.90	221	338	0	338	0.3953	1372
0.95	220	339	0	339	0.3936	1611
1.00	0	0	559	559	0.0000	0

ตารางที่ 4.50 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 50

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	356	203	0	203	0.6369	243
0.05	360	199	0	199	0.6440	243
0.10	358	201	0	201	0.6404	248
0.15	352	207	0	207	0.6297	248
0.20	367	192	0	192	0.6565	232
0.25	341	218	0	218	0.6100	225
0.30	148	411	0	411	0.2648	315
0.35	146	413	0	413	0.2612	308
0.40	146	413	0	413	0.2612	394
0.45	280	279	0	279	0.5009	525
0.50	228	331	0	331	0.4079	608
0.55	226	333	0	333	0.4043	642
0.60	220	339	0	339	0.3936	751
0.65	220	339	0	339	0.3936	823
0.70	219	340	0	340	0.3918	840
0.75	220	339	0	339	0.3936	861
0.80	220	339	0	339	0.3936	802
0.85	220	339	0	339	0.3936	750
0.90	220	339	0	339	0.3936	648
0.95	218	341	0	341	0.3900	480
1.00	0	0	559	559	0.0000	0

ตารางที่ 4.51 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 60

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	345	214	0	214	0.6172	189
0.05	345	214	0	214	0.6172	187
0.10	350	209	0	209	0.6261	190
0.15	350	209	0	209	0.6261	189
0.20	350	209	0	209	0.6261	184
0.25	326	233	0	233	0.5832	179
0.30	146	413	0	413	0.2612	241
0.35	147	412	0	412	0.2630	235
0.40	145	414	0	414	0.2594	296
0.45	239	320	0	320	0.4275	387
0.50	225	334	0	334	0.4025	442
0.55	225	334	0	334	0.4025	470
0.60	220	339	0	339	0.3936	550
0.65	220	339	0	339	0.3936	602
0.70	220	339	0	339	0.3936	597
0.75	220	339	0	339	0.3936	596
0.80	220	339	0	339	0.3936	554
0.85	219	340	0	340	0.3918	504
0.90	217	342	0	342	0.3882	429
0.95	217	342	0	342	0.3882	319
1.00	0	0	559	559	0.0000	0

ตารางที่ 4.52 ผลการทดลองชุดข้อมูล Reuters-Top10 โดยอัลกอริทึม ARTC ที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 70

Limit confidence	Classifying new documents					
	Correct	Miss classification	Unclassification	Incorrect	Accuracy	Applicable rule
0.00	351	208	0	208	0.6279	138
0.05	351	208	0	208	0.6279	138
0.10	352	207	0	207	0.6297	138
0.15	352	207	0	207	0.6297	138
0.20	352	207	0	207	0.6297	138
0.25	352	207	0	207	0.6297	137
0.30	149	410	0	410	0.2665	184
0.35	149	410	0	410	0.2665	177
0.40	145	414	0	414	0.2594	225
0.45	248	311	0	311	0.4436	295
0.50	226	333	0	333	0.4043	334
0.55	224	335	0	335	0.4007	344
0.60	221	338	0	338	0.3953	394
0.65	220	339	0	339	0.3936	418
0.70	219	340	0	340	0.3918	406
0.75	220	339	0	339	0.3936	385
0.80	219	340	0	340	0.3918	353
0.85	219	340	0	340	0.3918	318
0.90	217	341	1	342	0.3882	268
0.95	217	341	1	342	0.3882	183
1.00	0	0	559	559	0.0000	0

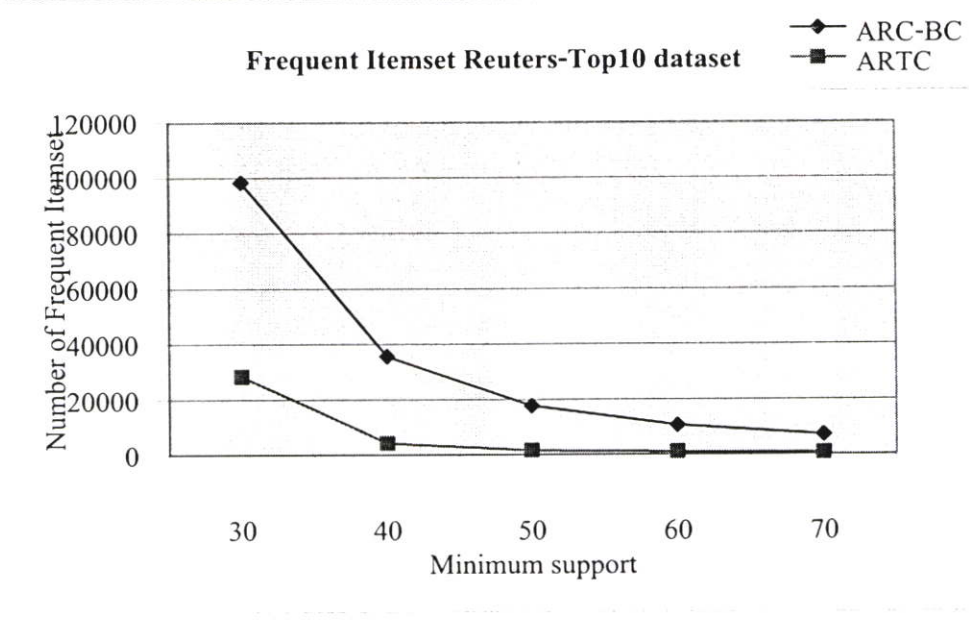
4.2.4.3 ผลการทดลองชุดข้อมูล Reuters-Top10

จากผลการทดลองของชุดข้อมูล Reuters-Top10 เลือกผลการทดลองที่ดีที่สุดของแต่ละอัลกอริทึมในแต่ละค่าสนับสนุนน้อยที่สุดที่ได้ทำการทดลอง โดยผลการทดลองที่ดีที่สุดคือผลการทดลองที่มีค่าความถูกต้องในการจำแนกประเภทเอกสาร (Accuracy rate) สูงที่สุด และกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร (Applicable rule) มีจำนวนน้อย นำผลการทดลองที่ดีที่สุดแสดงผลในตารางที่ 4.53

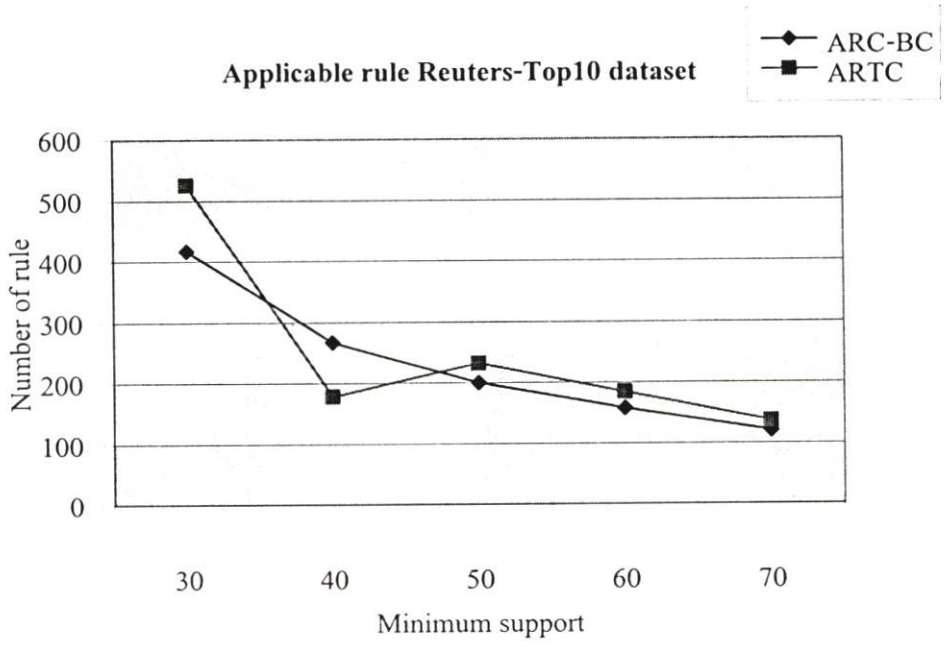
ตารางที่ 4.53 ผลการทดลองชุดข้อมูล Reuters-Top10

Minimum	ARC-BC			ARTC		
support	Frequent Itemset	Applicable rule	Accuracy	Frequent Itemset	Applicable rule	Accuracy
30	98292	417	0.5993	28490	525	0.7048
40	35675	267	0.5635	4269	178	0.6816
50	17745	200	0.5474	1607	232	0.6565
60	10675	158	0.5438	1092	184	0.6261
70	7224	121	0.5313	741	137	0.6297

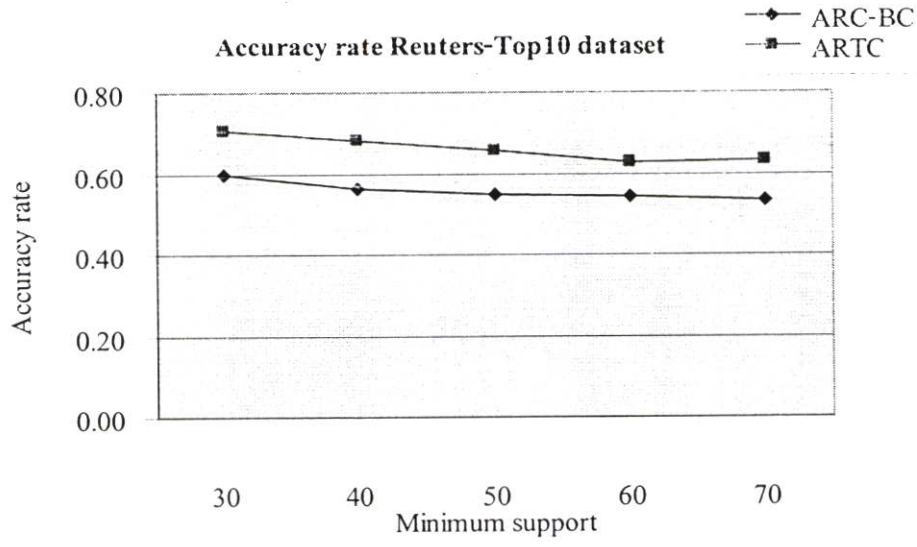
จากตารางที่ 4.53 นำข้อมูลที่ได้แสดงผลเป็นกราฟจำนวน Frequent Itemsetในรูปที่ 4.15 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารในรูปที่ 4.16 และรูปที่ 4.17 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร



รูปที่ 4.15 เปรียบเทียบจำนวน Frequent Itemset แต่ละค่าสนับสนุนน้อยที่สุดของชุดข้อมูล Reuters-Top10



รูปที่ 4.16 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร แต่ละค่าสนับสนุนน้อยที่สุด ของชุดข้อมูล Reuters-Top10



รูปที่ 4.17 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร แต่ละค่าสนับสนุนน้อยที่สุด ของชุดข้อมูล Reuters-Top10

จากการเปรียบเทียบผลการทดลองที่ได้ พบว่าในทุก ๆ ค่าสนับสนุนน้อยที่สุด อัลกอริทึม ARTC มีจำนวน Frequent Itemset น้อยกว่าอัลกอริทึม ARC-BC แสดงว่าอัลกอริทึม ARTC ค้นหาความสัมพันธ์ได้น้อยกว่าอัลกอริทึม ARC-BC เมื่อพิจารณาที่ประสิทธิภาพการจำแนกประเภทเอกสารพบว่าอัลกอริทึม ARTC มีประสิทธิภาพการจำแนกประเภทเอกสารได้ดีกว่าอัลกอริทึม ARC-BC ในจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารน้อยกว่าอัลกอริทึม ARC-BC ในทุก ๆ ค่าสนับสนุนน้อยที่สุด ยกเว้นที่ค่าสนับสนุนน้อยที่สุดเท่ากับ 40 ที่อัลกอริทึม ARC-BC ใช้กฎความสัมพันธ์ในการจำแนกประเภทเอกสารน้อยกว่า แต่อย่างไรก็ตามประสิทธิภาพการจำแนกประเภทเอกสารของอัลกอริทึม ARTC ก็ยังดีกว่าอัลกอริทึม ARC-BC

4.3 สรุปผลการทดลอง

จากการทดลองข้อมูลแต่ละชุดข้อมูล จะเลือกผลการทดลองที่ดีที่สุดในแต่ละชุดข้อมูล โดยพิจารณาที่ประสิทธิภาพการจำแนกเอกสารเป็นหลัก ทำการเปรียบเทียบทั้ง 2 อัลกอริทึม โดยเปรียบเทียบโดยเลือกประสิทธิภาพในการจำแนกประเภทเอกสารที่ดีที่สุดของอัลกอริทึม ARC-BC และเปรียบเทียบโดยเลือกประสิทธิภาพในการจำแนกประเภทเอกสารที่ดีที่สุดของอัลกอริทึม ARTC ดังต่อไปนี้

4.3.1 สรุปผลการทดลองที่ดีที่สุดของอัลกอริทึม ARC-BC

จากผลการทดลองของชุดข้อมูลทั้ง 4 ชุดข้อมูล เลือกประสิทธิภาพการจำแนกประเภทเอกสารที่ดีที่สุดของแต่ละชุดข้อมูลโดยเลือกผลการทดลองที่ดีที่สุดของอัลกอริทึม ARC-BC แสดงดังตารางที่ 4.54

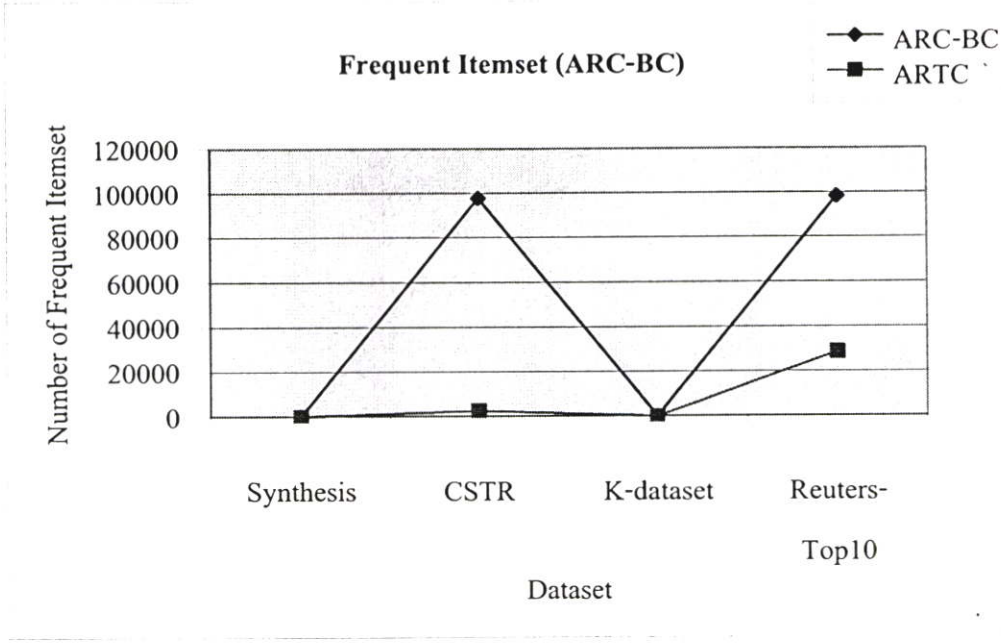
จากตารางที่ 4.54 นำข้อมูลมาสร้างกราฟแสดงผลการทดลองของข้อมูลแต่ละชุด โดยเปรียบเทียบจำนวน Frequent Itemset ที่ได้ของแต่ละชุดข้อมูลดังรูปที่ 4.18 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารดังรูปที่ 4.19 และเปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสารแสดงดังรูปที่ 4.20

ตารางที่ 4.54 เปรียบเทียบผลการทดลอง โดยเลือก Accuracy rate ที่ดีที่สุดของอัลกอริทึม ARC-BC

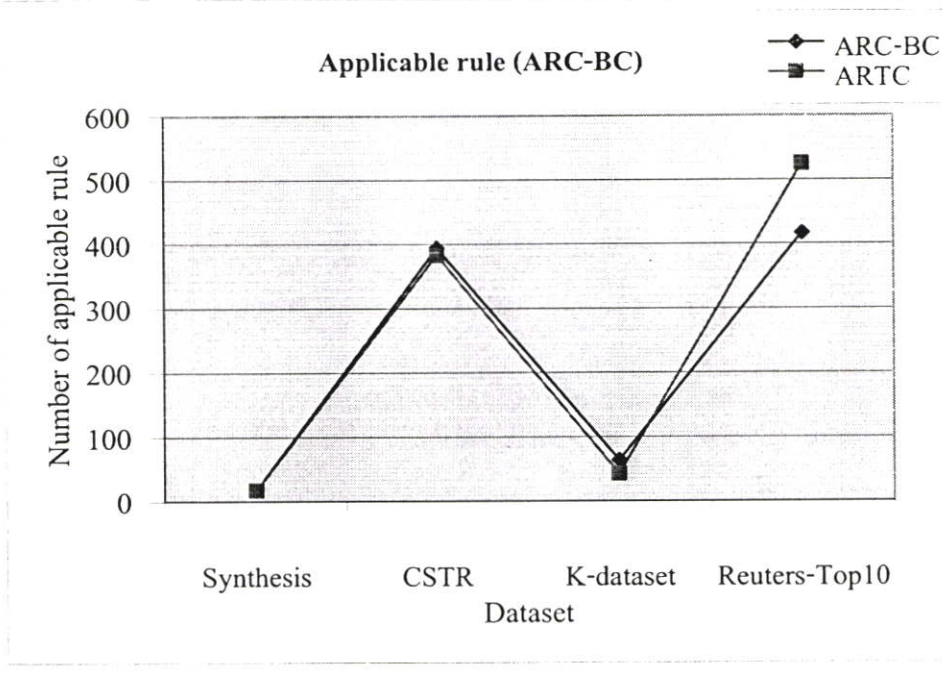
Dataset	ARC-BC			ARTC		
	Frequent Itemset	Applicable rule	Accuracy	Frequent Itemset	Applicable rule	Accuracy
Synthesis	114	18	0.9975	53	18	1.0000
CSTR	97664	394	0.7024	2546	383	0.8595
K-dataset	181	64	0.4430	72	43	0.5211
Reuters-Top10	98292	417	0.5993	28490	525	0.7048

ตารางที่ 4.55 เปรียบเทียบผลการทดลอง โดยเลือก Accuracy rate ที่ดีที่สุดของอัลกอริทึม ARTC

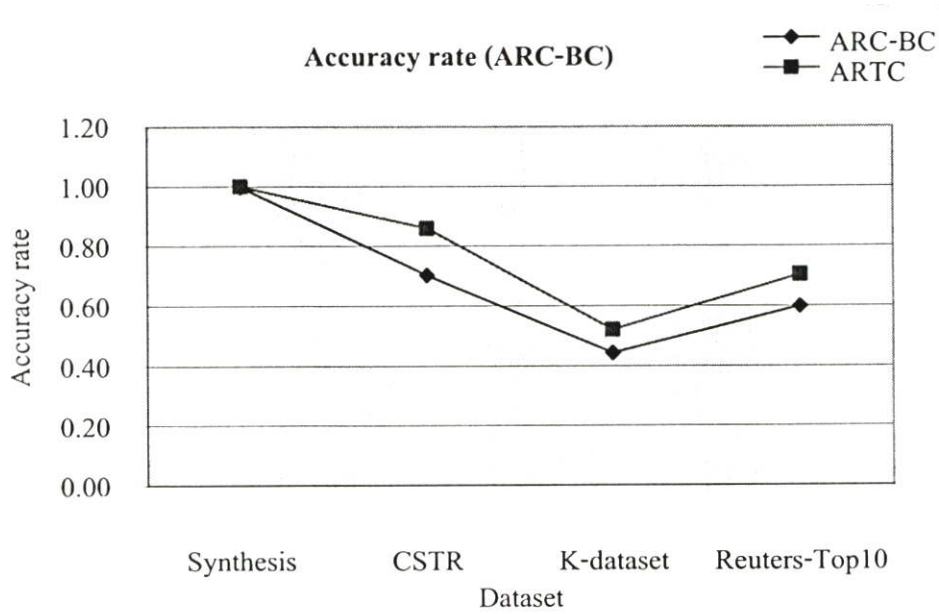
Dataset	ARC-BC			ARTC		
	Frequent Itemset	Applicable rule	Accuracy	Frequent Itemset	Applicable rule	Accuracy
Synthesis	114	18	0.9975	53	18	1.0000
CSTR	97664	394	0.7024	2546	383	0.8595
K-dataset	4149	481	0.3966	801	462	0.7869
Reuters-Top10	98292	417	0.5993	28490	525	0.7048



รูปที่ 4.18 เปรียบเทียบจำนวน Frequent Itemset โดยผลที่ดีที่สุดของอัลกอริทึม ARC-BC



รูปที่ 4.19 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร โดยผลที่ดีที่สุดของอัลกอริทึม ARC-BC



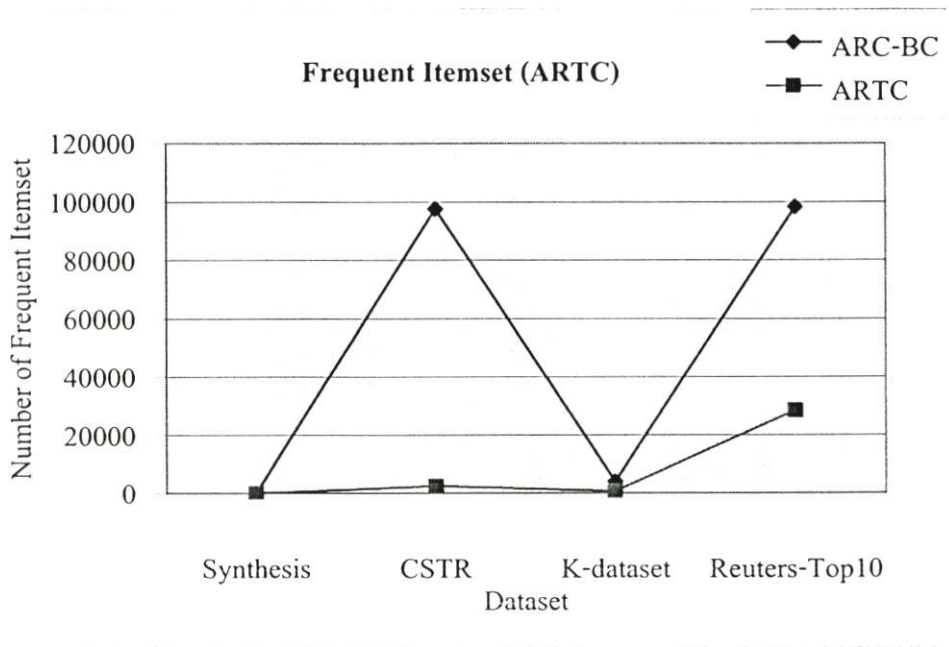
รูปที่ 4.20 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร
โดยผลที่ดีที่สุดของอัลกอริทึม ARC-BC

จากการเลือกผลการทดลองที่ดีที่สุดของอัลกอริทึม ARC-BC พบว่าประสิทธิภาพการจำแนกประเภทเอกสารของอัลกอริทึม ARC-BC น้อยกว่าอัลกอริทึม ARTC ในทุก ๆ ชุดข้อมูลที่ใช้ในการทดลอง โดยที่อัลกอริทึม ARC-BC มีจำนวน Frequent Itemset มากกว่าอัลกอริทึม ARTC ทุกชุดข้อมูล และมีจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารมากกว่าอัลกอริทึม ARTC ในชุดข้อมูลข้อความ (Synthesis) CSTR และ K-dataset สำหรับชุดข้อมูล Reuters-Top10 ที่อัลกอริทึม ARC-BC มีจำนวนกฎความสัมพันธ์น้อยกว่าอัลกอริทึม ARTC และประสิทธิภาพการจำแนกประเภทเอกสารของอัลกอริทึม ARC-BC มีน้อยกว่าอัลกอริทึม ARTC

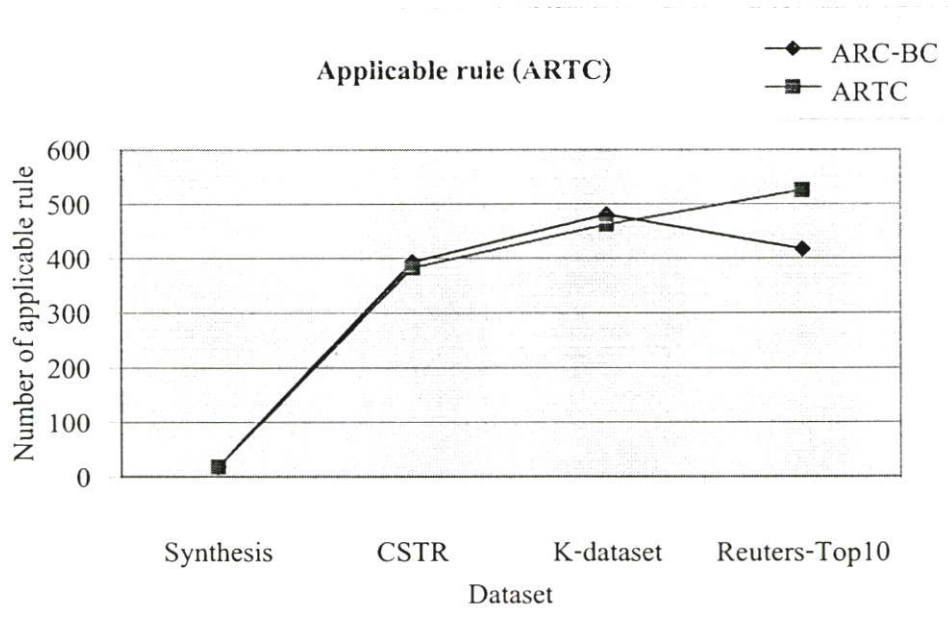
4.3.2 สรุปผลการทดลองที่ดีที่สุดของอัลกอริทึม ARTC

จากผลการทดลองของชุดข้อมูลทั้ง 4 ชุดข้อมูล เลือกประสิทธิภาพการจำแนกประเภทเอกสารที่ดีที่สุดของแต่ละชุดข้อมูลโดยเลือกผลการทดลองที่ดีที่สุดของอัลกอริทึม ARTC แสดงดังตารางที่ 4.55

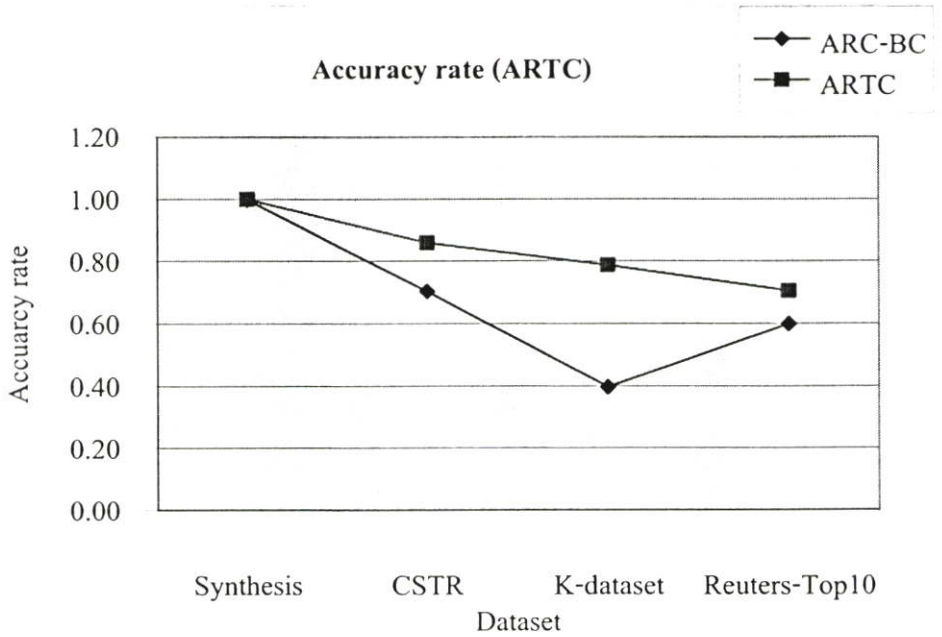
จากตารางที่ 4.55 นำข้อมูลมาสร้างกราฟแสดงผลการทดลองของข้อมูลแต่ละชุด โดยเปรียบเทียบจำนวน Frequent Itemset ที่ได้ของแต่ละชุดข้อมูลดังรูปที่ 4.21 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารดังรูปที่ 4.22 และเปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสารแสดงดังรูปที่ 4.23



รูปที่ 4.21 เปรียบเทียบจำนวน Frequent Itemset โดยผลที่ดีที่สุดของอัลกอริทึม ARTC



รูปที่ 4.22 เปรียบเทียบจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร โดยผลที่ดีที่สุดของอัลกอริทึม ARTC



รูปที่ 4.23 เปรียบเทียบประสิทธิภาพการจำแนกประเภทเอกสาร
โดยผลที่ดีที่สุดของอัลกอริทึม ARTC

จากการเลือกผลการทดลองที่ดีที่สุดของอัลกอริทึม ARTC พบว่าประสิทธิภาพการจำแนกประเภทเอกสารของอัลกอริทึม ARTC ดีกว่าอัลกอริทึม ARC-BC ในทุก ๆ ชุดข้อมูลที่ใช้ในการทดลอง โดยที่อัลกอริทึม ARTC มีจำนวน Frequent Itemset น้อยกว่าอัลกอริทึม ARC-BC ทุกชุดข้อมูล และมีจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารน้อยกว่าอัลกอริทึม ARC-BC ในชุดข้อมูลข้อความ (Synthesis) CSTR และ K-dataset สำหรับชุดข้อมูล Reuters-Top10 พบว่าอัลกอริทึม ARC-BC มีจำนวนกฎความสัมพันธ์น้อยกว่าอัลกอริทึม ARTC แต่ประสิทธิภาพการจำแนกประเภทเอกสารของอัลกอริทึม ARTC มีมากกว่าอัลกอริทึม ARC-BC

บทที่ 5
สรุปผลการวิจัย และข้อเสนอแนะ

เนื่องจากการทดลองของงานวิจัยนี้ได้ใช้ค่าพารามิเตอร์จำนวนมากในการทดลอง และได้ผลการทดลองของแต่ละค่าพารามิเตอร์มาเป็นจำนวนมาก ทั้ง 2 อัลกอริทึม เพื่อความยุติธรรมของแต่ละอัลกอริทึม ดังนั้นการเลือกผลการทดลองจึงมองออกเป็น 2 ส่วนคือ เลือกผลการทดลองเปรียบเทียบของทั้งสองอัลกอริทึมโดยอิงประสิทธิภาพในการจำแนกประเภทเอกสารที่ดีที่สุดของอัลกอริทึม ARC-BC (ตารางที่ 4.45) และเลือกผลการทดลองเปรียบเทียบของทั้งสองอัลกอริทึมโดยอิงประสิทธิภาพในการจำแนกประเภทเอกสารที่ดีที่สุดของอัลกอริทึม ARTC (ตารางที่ 4.46) ซึ่งนำมาใช้ในการสรุปงานวิจัยฉบับนี้

5.1 สรุปผลการวิจัย

เนื่องจากสมมุติฐานของงานวิจัยนี้ เริ่มต้นจากการมองข้อด้อยของอัลกอริทึม ARC-BC ซึ่งก็คือการค้นหาความสัมพันธ์ที่มากเกินความจำเป็นในการจำแนกประเภทเอกสาร การคัดกฎความสัมพันธ์ทิ้งมากเกินไปจนส่งผลให้ประสิทธิภาพในการจำแนกประเภทเอกสารด้อยไปด้วย ดังนั้นจึงได้มีการสร้างงานวิจัยนี้เพื่อปรับปรุงอัลกอริทึม ARC-BC เพื่อลดข้อด้อยทั้ง 3 ประการดังที่กล่าวมา และสามารถสรุปแต่ละประเด็นดังนี้

1. จำนวน Frequent Itemset

จากผลการทดลองที่ได้ไม่ว่าจะมองในแง่ที่เลือกผลการทดลองที่ดีที่สุดของอัลกอริทึม ARC-BC หรือผลการทดลองที่ดีที่สุดของอัลกอริทึม ARTC พบว่าจำนวน Frequent Itemset ของอัลกอริทึม ARTC น้อยกว่าอัลกอริทึม ARC-BC ทั้งนี้เพราะวิธีการในการค้นหาความสัมพันธ์ที่แตกต่างกัน โดยอัลกอริทึม ARC-BC ใช้วิธีการค้นหาความสัมพันธ์ที่ค้นหาทุก ๆ ความสัมพันธ์โดยไม่คำนึงว่าความสัมพันธ์ที่ได้สามารถนำมาสร้างเป็นกฎความสัมพันธ์ที่ใช้จำแนกประเภทเอกสารได้หรือไม่ กล่าวคือความสัมพันธ์นั้นไม่มีชื่อกลุ่มเอกสารปรากฏอยู่จึงทำให้ไม่สามารถนำมาสร้างเป็นกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารได้ ในขณะที่อัลกอริทึม ARTC ได้ออกแบบให้มีการค้นหาความสัมพันธ์ทุก ๆ ความสัมพันธ์ที่สามารถนำมาใช้ในการสร้างกฎความสัมพันธ์เพื่อจำแนกประเภทเอกสารได้เท่านั้นกล่าวคือความสัมพันธ์ที่ได้ทุก ๆ ความสัมพันธ์ต้องมีชื่อกลุ่มเอกสารปรากฏอยู่ เพื่อนำมาใช้สร้างกฎความสัมพันธ์ในการจำแนกประเภทเอกสารได้ ในส่วนนี้เองที่ทำให้อัลกอริทึม ARTC มี Frequent Itemset ที่น้อยกว่าอัลกอริทึม ARC-BC

2. จำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกเอกสาร

จากผลการทดลองที่ได้ไม่ว่าจะมองในแง่ที่เลือกผลการทดลองที่ดีที่สุดของอัลกอริทึม ARC-BC หรือผลการทดลองที่ดีที่สุดของอัลกอริทึม ARTC พบว่าในชุดข้อมูล CSTR และชุดข้อมูล K-dataset นั้นอัลกอริทึม ARTC มีจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกเอกสารน้อยกว่าอัลกอริทึม ARC-BC ในขณะที่ชุดข้อมูล Reuters-Top10 พบว่าจำนวนกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารของอัลกอริทึม ARTC มากกว่าอัลกอริทึม ARC-BC ทั้งนี้เป็นเพราะวิธีการคัดกฎความสัมพันธ์ทิ้งของแต่ละวิธีไม่เหมือนกัน อัลกอริทึม ARC-BC จะเลือกเฉพาะกฎที่สั้นที่สุด (General) เก็บไว้ ในขณะที่อัลกอริทึม ARTC จะคัดกฎความสัมพันธ์โดยใช้วิธีสร้างต้นไม้และเลือกกฎความสัมพันธ์ที่มีค่าความเชื่อมั่นของกฎผ่านค่า Limit confidence ดังนั้นหากกฎที่เป็นโหนดแม่ไม่ผ่านค่า Limit confidence โอกาสที่จะได้กฎความสัมพันธ์ที่นำไปใช้ในการจำแนกประเภทก็จะมากขึ้นตามไปด้วย จากชุดข้อมูลทั้ง 3 ชุด คาดว่าเนื่องจากชุดข้อมูล CSTR และ K-dataset จะมีกฎความสัมพันธ์ที่เป็นโหนดแม่ผ่านค่า Limit confidence แล้ว ดังนั้นจึงได้กฎความสัมพันธ์ในการจำแนกประเภทเอกสารไม่มากนัก ในขณะที่ชุดข้อมูล Reuters-Top10 มีกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทมากเพราะกฎความสัมพันธ์ที่เป็นโหนดแม่ไม่ผ่านค่า Limit confidence ดังนั้นกฎความสัมพันธ์ที่ได้ในการจำแนกประเภทเอกสารจึงมีมาก

3. ประสิทธิภาพในการจำแนกประเภทเอกสาร

จากผลการทดลองที่ได้ไม่ว่าจะมองในแง่ที่เลือกผลการทดลองที่ดีที่สุดของอัลกอริทึม ARC-BC หรือผลการทดลองที่ดีที่สุดของอัลกอริทึม ARTC พบว่าประสิทธิภาพในการจำแนกประเภทเอกสารของอัลกอริทึม ARTC ดีกว่าอัลกอริทึม ARC-BC จึงเป็นไปตามสมมุติฐานที่ตั้งไว้

5.2 ข้อเสนอแนะในการทำวิจัยต่อไป

1. เนื่องจากธรรมชาติของวิธีการค้นหากฎความสัมพันธ์ จะเป็นการค้นหากฎความสัมพันธ์ทุก ๆ ความสัมพันธ์ที่เป็นไปได้ ดังนั้นระยะเวลาในการค้นหากฎความสัมพันธ์จึงใช้เวลานานมากกว่าที่จะค้นหากฎความสัมพันธ์ได้ ทั้งนี้ขึ้นอยู่กับจำนวนข้อมูลที่ใช้ในการสอนระบบด้วย แต่เนื่องจากข้อดีของวิธีการนี้คือ ผู้ใช้สามารถควบคุมผลลัพธ์ได้ซึ่งก็คือการกำหนดค่าสนับสนุนน้อยที่สุด ในการพิจารณากำหนดค่าสนับสนุนน้อยที่สุดควรพิจารณาตามสัดส่วนของจำนวนข้อมูลที่ใช้ในการสอนระบบ หากมีข้อมูลมากก็กำหนดค่าสนับสนุนน้อยที่สุดมากตามและเช่นกันหากมีข้อมูลน้อยก็กำหนดค่าสนับสนุนน้อยที่สุดน้อยตามจำนวนข้อมูล ทั้งนี้เพราะจะได้ไม่เสียเวลามากในการค้นหากฎความสัมพันธ์ที่นานเกินไป

2. เนื่องจากงานวิจัยนี้ ได้ทำการวิจัยข้อมูลเพียง 1 มิติ ซึ่งก็คือคำสำคัญ (Keyword) ของเอกสาร หากมีการพัฒนาให้สามารถใช้งานกับข้อมูลที่มากกว่า 1 มิติ ซึ่งก็คือพิจารณาข้อมูล

เพิ่มขึ้น เช่น ชื่อเรื่อง (Title) ชื่อผู้แต่ง (Author) เป็นต้น ก็จะทำให้สามารถเพิ่มประสิทธิภาพการจำแนกประเภทเอกสารได้ดีขึ้น

บรรณานุกรม

[1] Jiawei Han and Micheline Kamber, **Data Mining: Concepts and Techniques**, CA : Morgan kaufmann, San Francisco, 2001.

[2] Fabrizio Sebastiani, "Machine learning in automated text categorization", **ACM Computing Surveys**, 2002.

[3] Osmar R. Zaiane and Maria-Luiza Antonie, "Classifying Text Documents by Association Terms with Text Categories," **Proceedings of the thirteenth Australasian conference on Database technologies**, Australian Computer Society,Inc., pp. 215-222, 2002.

[4] Osmar R. Zaiane and Maria-Luiza Antonie, "Text Document Categorization by Term Association," **Proceedings of ICDM 2002**, IEEE, pp. 19-26, 2002.

[5] The University of Rochester Computer Science. **"URCS Technical report"** [Online]. Available: http://www.cs.rochester.edu/trs/.

[6] WebACE project. [Online]. Available: ftp://ftp.cs.umn.edu/dept/users/boley/PDDPdata/.

[7] Han, E.-H., Boley, D., Gini, M., Gross, R., Hastings, K., Karypis, G., Kumar, V., Mobasher, B., & Moore, J. (1998) WebACE: A web agent for document categorization and exploration. Agents-98.

[8] Lewis D.D., "**Reuters-21578 (Top 10)**" [Online]. Avaliable: http://www.reeltwo.com/datasets.html.

[9] Lewis D.D. **"Reuters-21578 text categorization test collection distribution 1.0."** [Online]. Available: http://daviddlewis.com, 2004.

[10] Baeza-Yates, R., **Modern Information Retrieval.** New York : Addison-Wesley, Inc. 1999.

[11] Cabana, P., Hadjinian, P., Standler, R., Verhees, J. and Zanasi, A. 1997. **Discovering Data Mining from concept to implement.** New Jersey : Prentice Hall PTR.

[12] D. Boley, M. Gini, R. Gross, S. Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, J. Moore, **Partitioning-Based Clustering for Web Document Categorization,** , Decision Support Systems 27:329-341, 1999.

[13] D. Boley, M. Gini, R. Gross, E.-H. (Sam) Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, J. Moore, **Document Categorization and Query Generation on the World Wide Web Using WebACE,** AI Review 13, 1999.

[14] D. L. Boley, **Principal Direction Divisive Partitioning,** , Data Mining and Knowledge Discovery 2(4):325-344 , 1998.

[15] Tom M. Mitchell, **Machine learning**, McGraw-Hill, New York, 1997.

ภาคผนวก

ภาคผนวก ก.

Stop word list ที่ใช้ในการเตรียมข้อมูล

a
a's
able
about
above
according
accordingly
across
actually
after
afterwards
again
against
ain't
all
allow
allows
almost
alone
along
already
also
although
always
am
among
amongst
an
and
another
any
anybody
anyhow
anyone
anything
anyway
anyways
anywhere
apart
appear
appreciate
appropriate
are
aren't
around
as
aside
ask
asking
ssociated
at
available
away
awfully
b
be
became
because
become
becomes
becoming
been
before
beforehand
behind
being
believe
below
beside
besides
best
better
between
beyond
both
brief
but
by
c
c'mon
c's
came
can
can't
cannot
cant
cause
causes
certain
certainly
changes
clearly
co
com
come
comes
concerning
consequently
consider
considering
contain
containing
contains
corresponding
could
couldn't
course
currently
d
definitely
described
despite
did
didn't
different
do

does
doesn't
doing
don't
done
down
downwards
during
e
each
edu
eg
ight
either
else
elsewhere
enough
entirely
especially
et
etc
even
ever
every
everybody
everyone
everything
everywhere
ex
exactly
example
except
f
far
few
fifth
first
five
followed
following
follows
for
former
formerly
forth
four
from
further
furthermore
g
get
gets
getting
given
gives
go
goes
going
gone
got
gotten
greetings
h
had
hadn't
happens
hardly
has
hasn't
have
haven't
having
he
he's
hello
help
hence
her
here
here's
hereafter
hereby
herein
hereupon
hers
herself
hi
him
himself
his
hither
hopefully
how
howbeit
however
i
i'd
i'll
i'm
i've
ie
if
ignored
immediate
in
inasmuch
inc
indeed
indicate
indicated
indicates
inner
insofar
instead
into
inward

is
isn't
it
it'd
it'll
it's
its
itself
j
just
k
keep
keeps
kept
know
knows
known
l
last
lately
later
latter
latterly
least
less
lest
let
let's
like
liked
likely
little
look
looking
looks
ltd
m
mainly
many
may
maybe
me
mean
meanwhile
merely
might
more
moreover
most
mostly
much
must
my
myself
n
name
namely
nd
near
nearly
necessary
need
needs
neither
never
nevertheless
new
next
nine
no
nobody
non
none
noone
nor
normally
not
nothing
novel
now
nowhere
o
obviously
of
off
often
oh
ok
okay
old
on
once
one
ones
only
onto
or
other
others
otherwise
ought
our
ours
ourselves
out
outside
over
overall
own
p
particular
particularly
per
perhaps
placed
please

plus
possible
presumably
probably
provides
q
que
quite
qv
r
rather
rd
re
really
reasonably
regarding
regardless
regards
relatively
respectively
right
s
said
same
saw
say
saying
says
second
secondly
see
seeing
seem
seemed
seeming
seems
seen
self
selves
sensible
sent
serious
seriously
seven
several
shall
she
should
shouldn't
since
six
so
some
somebody
somehow
someone
something
sometime
sometimes
somewhat
somewhere
soon
sorry
specified
specify
specifying
still
sub
such
sup
sure
t
t's
take
taken
tell
tends
th
than
thank
thanks
thanx
that
that's
thats
the
their
theirs
them
themselves
then
thence
there
there's
thereafter
thereby
therefore
therein
theres
thereupon
these
they
they'd
they'll
they're
they've
think
third
this
thorough
thoroughly
those
though
three
through
throughout

thru
thus
to
together
too
took
toward
towards
tried
tries
truly
try
trying
twice
two
u
un
under
unfortunately
unless
unlikely
until
unto
up
upon
us
use
used
useful
uses
using
usually
uucp
v
value
various
very
via
viz
vs
w
want
wants
was
wasn't
way
we
we'd
we'll
we're
we've
welcome
well
went
were
weren't
what
what's
whatever
when
whence
whenever
where
where's
whereafter
whereas
whereby
wherein
whereupon
wherever
whether
which
while
whither
who
who's
whoever
whole
whom
whose
why
will
willing
wish
with
within
without
won't
wonder
would
would
wouldn't
x
y
yes
yet
you
you'd
you'll
you're
you've
your
yours
yourself
yourselves
z
zero

ภาคผนวก ข.

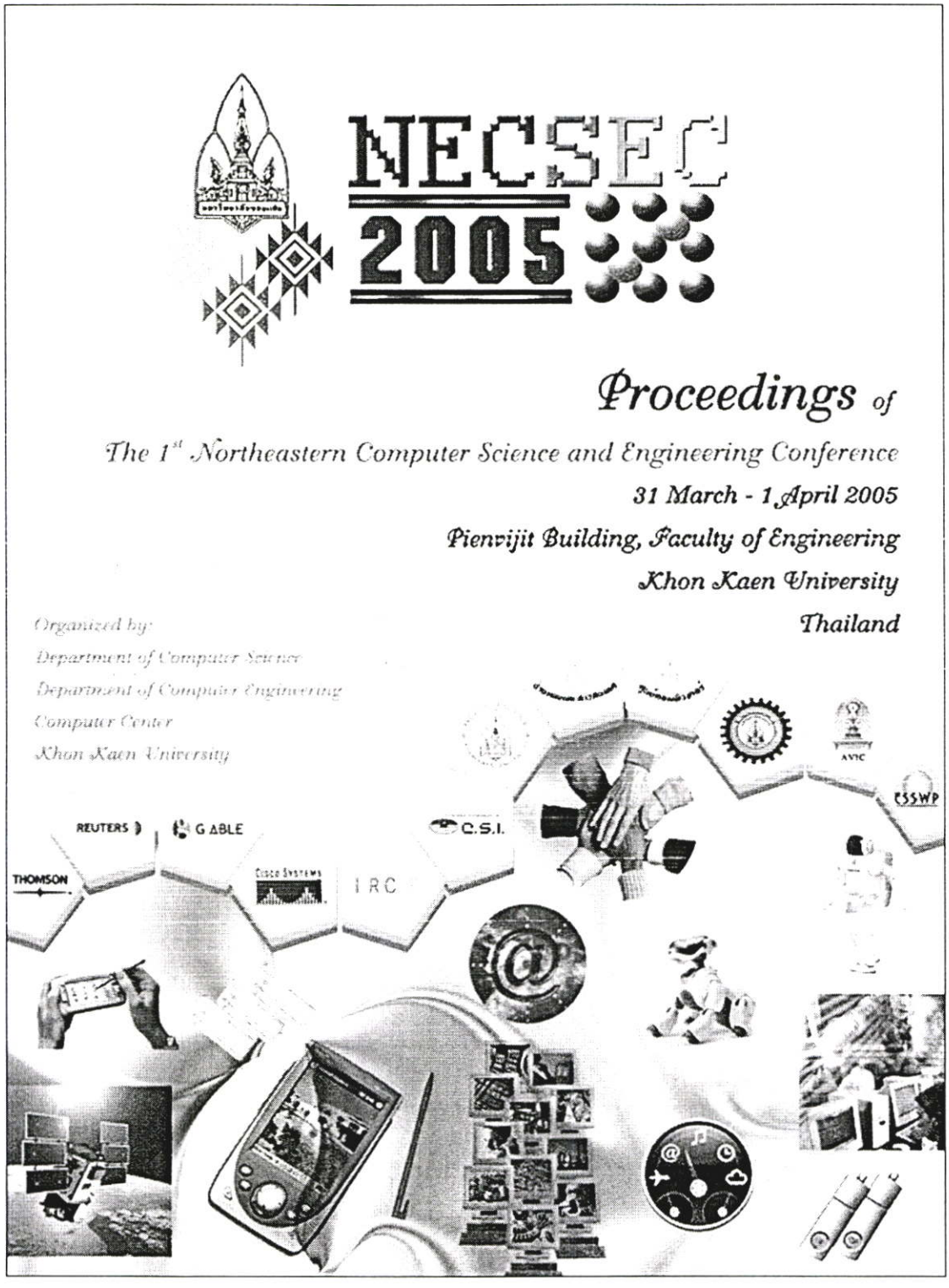
ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. Worapoj Kreesuradej and Supaporn Buddeewong, "**แบบจำลองการจัดประเภทเอกสารแบบหลายกลุ่มโดยใช้กฎความสัมพันธ์**", Proceedings of The 1st Northeastern Computer Science and Engineering Conference (NESEC 2005), Khon Kaen University, 31 March – 1 April 2005, pp. 283-288.

2. Supaporn Buddeewong and Worapoj Kreesuradej, "**Text Categorization using a New Association Rule-Based Text Classifier Algorithm**", The 9th National Computer Science and Engineering Conference (NCSEC 2005), University of the Thai Chamber of Commerce, October 27-28 2005, pp. 289-296.

3. Supaporn Buddeewong and Worapoj Kreesuradej , "**Text Cateogization using a New Text Association Rule-Based Classifier",** International Conference on Computer and Industrial Management (ICIM 2005), Assumption University, October 29-30 2005, pp. 8.1-8.4.

4. Supaporn Buddeewong and Worapoj Kreesuradej, **"A New Association Rule-Based Text Classifier Algorithm"**, The 17th IEEE International Conference on Tool with Artificial Intelligence (ICTAI05), Hong Kong, November 14-16 2005, pp.684-685


NECSEC
2005
Proceedings of
The 1st Northeastern Computer Science and Engineering Conference
31 March - 1 April 2005
Pienvijit Building, Faculty of Engineering
Khon Kaen University
Thailand
Organized by:
Department of Computer Science
Department of Computer Engineering
Computer Center
Khon Kaen University
REUTERS
G ABLE
C.S.I.
THOMSON
IRC
AVIC
ESSWP

แบบจำลองการจัดประเภทเอกสารแบบหลายกลุ่ม โดยใช้กฎความสัมพันธ์

วรพจน์ กรีสุระเดช
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง
Email: worapoj@it.kmitl.ac.th

สุภาภรณ์ บุตรดีวงษ์
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง
Email: s5066003@kmitl.ac.th

บทคัดย่อ

การจัดประเภทเอกสาร (Text categorization) เป็นกระบวนการในการแบ่งกลุ่มเอกสารให้อยู่ในกลุ่มที่เหมาะสมสำหรับเอกสารนั้น ๆ การหากฎความสัมพันธ์ (Association rule discovery) เป็นวิธีหนึ่งสำหรับการคัดแยกเอกสารอัตโนมัติ สำหรับงานวิจัยนี้ได้นำเสนอวิธีการคัดแยกเอกสารอัตโนมัติแบบหลายกลุ่มโดยใช้วิธีการหากฎความสัมพันธ์ ซึ่งกฎความสัมพันธ์ที่ได้สามารถคัดแยกเอกสารทั้งที่เป็นกลุ่มเดียว (Single class) และหลายกลุ่ม (Multi class) ได้อย่างถูกต้อง

คำสำคัญ การจัดประเภทเอกสาร, กฎความสัมพันธ์, การจัดกลุ่มเอกสาร, เทิกซ์ไมนิ่ง

1. บทนำ

การวิจัยเกี่ยวกับการจัดประเภทเอกสารเริ่มต้นเมื่อต้นปี ค.ศ. 1960 จนมาถึงปัจจุบัน ซึ่งยังมีการวิจัยกันกว้างเพื่อปรับปรุงวิธีการในการจัดประเภทเอกสารให้ดีขึ้นกว่าเดิม ทั้งนี้เนื่องจากการเพิ่มขึ้นอย่างรวดเร็วของข้อมูลอิเล็กทรอนิกส์บนเครือข่ายอินเทอร์เน็ต และในองค์กรอื่น ๆ

การจัดประเภทเอกสารเป็นการจัดกลุ่มเอกสารใหม่ที่เข้ามาในระบบไปยังกลุ่มที่เหมาะสมสำหรับเอกสารนั้น โดยทั่ว ๆ ไปมีขั้นตอนการดำเนินงานคร่าว ๆ 2 ขั้นตอน คือ ขั้นตอนการฝึกหัดระบบ (Training step) และขั้นตอนการทดสอบระบบ (Testing step) การฝึกหัดระบบเป็นกระบวนการในการสร้างแบบจำลอง (Model) ระบบจัดประเภทเอกสารจากกลุ่มเอกสารสำหรับฝึกหัดระบบ (Training set) ซึ่งเป็นการเรียนรู้ (Learning) จึงแบบแผนของข้อมูลจากกลุ่มข้อมูลที่จัดเตรียมไว้ เอกสารแต่ละเอกสารในกลุ่มของเอกสารสำหรับฝึกหัดระบบต้องประกอบด้วยชื่อกลุ่มของเอกสาร (Class label) และคุณลักษณะ (Feature) ของเอกสารนั้น ๆ เมื่อได้แบบจำลองระบบแล้วขั้นตอนต่อไปก็คือนำแบบจำลองที่ได้มาทดสอบความถูกต้องของระบบ โดยนำกลุ่มเอกสารสำหรับทดสอบระบบเข้าสู่แบบจำลองที่ได้เพื่อทดสอบการจัดประเภทเอกสารว่าสามารถจัดประเภทเอกสารได้ถูกต้องหรือไม่ สำหรับเอกสารที่นำมาทดสอบระบบนั้นประกอบด้วยคุณลักษณะของเอกสาร และกลุ่มของเอกสารด้วย เพื่อใช้ตรวจสอบว่าระบบสามารถจัดประเภทเอกสารได้ถูกต้องหรือไม่

ในเอกสารนี้ได้นำเสนอผลงานวิจัยแบบจำลองการจัดประเภทเอกสารโดยใช้กฎความสัมพันธ์ในการจำแนกเอกสารแบบหลายกลุ่ม โดยหัวข้อที่ 2 กล่าวถึงความรู้พื้นฐานของกฎความสัมพันธ์ หัวข้อที่ 3 กล่าวถึงแบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่ม ในหัวข้อที่ 4 กล่าวถึงผลการทดลอง และหัวข้อที่ 5 สรุปผลและงานที่จะทำต่อไป

2. ความรู้พื้นฐาน

การหากฎความสัมพันธ์เป็นเทคนิคของดาต้าไมนิง ในการค้นหาความสัมพันธ์ระหว่างรายการในแต่ละรายการหรือกลุ่มของรายการ ที่ปรากฏขึ้นในฐานข้อมูล ความสัมพันธ์ที่ได้จะสามารถบอกลักษณะของข้อมูลหรือทำนายลักษณะของข้อมูลต่อไปได้ โดยทั่วไปความสัมพันธ์จะปรากฏอยู่ในรูปของกฎ "ถ้า ... แล้ว ..." (If ... Then ...) ในกฎหนึ่ง ๆ ประกอบด้วย 2 ส่วนคือ ส่วนด้านซ้ายของกฎ (ส่วน "ถ้า") และส่วนด้านขวาของกฎ (ส่วน "แล้ว") โดยส่วนด้านซ้ายอาจประกอบด้วยหนึ่งหรือมากกว่าหนึ่งเงื่อนไขที่เป็นจริง ที่จะทำให้ส่วนด้านขวาของกฎเป็นจริง เช่น "ถ้า T แล้ว C" ใช้สัญลักษณ์แทน "T => C" หมายถึง ถ้าเกิด T แล้วจะเกิด C ด้วย หากนำมาใช้สำหรับการจัดประเภทเอกสาร ก็จะตีความได้ว่า ถ้าเอกสารมีคุณลักษณะ T แล้วจะบอกได้ว่าเอกสารอยู่กลุ่ม C โดยที่ T เป็นคุณลักษณะของเอกสารและ C เป็นชื่อกลุ่มเอกสาร

นิยามและความหมายของคำต่าง ๆ ที่ใช้ในการหากฎความสัมพันธ์ได้แก่ ไอเทม (Item) คือข้อมูลแต่ละตัวที่ใช้ในการหากฎความสัมพันธ์ ไอเทมเซต (Itemset) คือกลุ่มของข้อมูลที่ประกอบด้วยไอเทมมากกว่า 1 ไอเทม โดยทั่วไปจะใช้แทนเป็น k-itemsets หมายถึงไอเทมเซตที่มี k ไอเทม เมื่อหากฎความสัมพันธ์ได้แล้วย่อมต้องมีการประเมินค่าของกฎที่ได้ โดยใช้ค่าสนับสนุน (Support value) และค่าความน่าเชื่อถือ (Confidence value) ค่าสนับสนุนคือเปอร์เซ็นต์ของจำนวนครั้งที่กฎเกิดขึ้น ค่าความน่าเชื่อถือ คือเปอร์เซ็นต์ของความเป็นไปได้ของความสัมพันธ์ของ T และ C ฟรีเควนท์ไอเทมเซต (Frequent Itemset) คือชุดของไอเทมเซตที่มีค่าสนับสนุนมากกว่าค่าสนับสนุนน้อยที่สุด (Minimum support) โดยที่ค่าสนับสนุนน้อยที่สุด คือค่าที่สนับสนุนที่น้อยที่สุดที่ทำให้กฎนั้นยังมีความน่าสนใจ และค่าความน่าเชื่อถือน้อยที่สุด (Minimum confidence) คือค่าความน่าเชื่อถือที่น้อยที่สุดที่ทำให้กฎนั้นยังมีความน่าสนใจ

อัลกอลิทึมที่นิยมใช้ในการหากฎความสัมพันธ์ของข้อมูลคือ อัลกอลิทึม Apriori ซึ่งเป็นอัลกอลิทึมดั้งเดิมสำหรับหาฟรีเควนท์ไอเทมเซต ถึงแม้ว่าจะมีอัลกอลิทึมอื่นที่มีประสิทธิภาพดีกว่าแต่ก็มีพื้นฐานมาจากอัลกอลิทึมนี้เป็นส่วนใหญ่ ดังนั้นจึงใช้อัลกอลิทึมนี้ในการอธิบายการหากฎความสัมพันธ์ ซึ่งประกอบด้วย 2 ขั้นตอนคือ การหาฟรีเควนท์ไอเทมเซตทั้งหมด และการสร้างกฎความสัมพันธ์จากฟรีเควนท์ไอเทมเซตโดยแต่ละขั้นตอนมีวิธีการต่าง ๆ ดังนี้

- การหาฟรีเควนท์ไอเทมเซต : การทำงานของขั้นตอนนี้คือจะมีการทำงานที่ทำซ้ำไปเรื่อย ๆ กล่าวคือฟรีเควนท์ไอเทมเซต k จะถูกใช้ในการหา (k+1)-ฟรีเควนท์ไอเทมเซต , L_1 เป็นสัญลักษณ์ใช้แทน 1-ฟรีเควนท์ไอเทมเซต และ L_1 จะถูกใช้ในการหา 2-ฟรีเควนท์ไอเทมเซต หรือ L_2 , L_2 ก็จะถูกใช้เพื่อหา 3-ฟรีเควนท์ไอเทมเซต หรือ L_3 เช่นนี้ไปเรื่อย ๆ เพื่อเป็นการเพิ่มประสิทธิภาพของอัลกอลิทึม คือ เพื่อช่วยลดพื้นที่ที่จะต้องค้นหาฟรีเควนท์ไอเทมเซตในฐานข้อมูล จะกระทำโดย ไอเทมเซตใด ๆ ที่มีค่าสนับสนุนน้อยกว่าค่าสนับสนุนน้อยที่สุดที่ตั้งไว้ ถือว่าไอเทมเซตนั้น ๆ ไม่เป็นฟรีเควนท์ไอเทมเซต
- การสร้างกฎความสัมพันธ์จากฟรีเควนท์ไอเทมเซต : ฟรีเควนท์ไอเทมเซตที่ได้จะถูกนำมาสร้างกฎความสัมพันธ์โดยนำ ฟรีเควนท์ไอเทมเซต L หาสับเซตของฟรีเควนท์ไอเทมเซต L และทุกสับเซต s ของ L ยกเว้นเซตว่าง จะได้กฎดังนี้ "s => (L-s)"

3. แบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่ม

แบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่ม เป็นวิธีใหม่สำหรับการจัดประเภทเอกสารที่จะนำเสนอในหัวข้อนี้ โดยมีวัตถุประสงค์เพื่อลดจำนวนกฎให้มีจำนวนน้อยลง และกฎที่ได้ไม่ขัดแย้งกัน ซึ่งผลที่ได้ทำให้สามารถจัดแยกเอกสารทั้งที่เป็นกลุ่มเดียวและหลายกลุ่มได้อย่างถูกต้อง

3.1 การหากฎความสัมพันธ์

โดยทั่วไปของการหากฎความสัมพันธ์จะมีเซตของข้อมูลหลัก ๆ อยู่ 2 เซตคือแคนดิเดตท์ไอเทมเซต (Candidate itemset) และฟรีเควนท์ไอเทมเซต แต่สำหรับแบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่มในงานวิจัยนี้มีเซตข้อมูลมากกว่าที่กล่าวมา คือ T, OL และ ML

แบบจำลองการจำแนกกลุ่มเอกสารแบบหลายกลุ่มประกอบด้วยกลุ่มไอเทมเซตต่าง ๆ ดังนี้ เซต T เกิดจากการเชื่อมความสัมพันธ์ (Join) ระหว่าง L_0 และ L_1 โดย L_0 เป็นไอเทมเซตที่เป็นชื่อประเภทเอกสารที่ผ่านค่าสนับสนุนที่น้อยที่สุดแล้ว และ L_1 เป็นไอเทมเซตที่เป็นคุณลักษณะของเอกสารที่ผ่านค่าสนับสนุนที่น้อยที่สุดแล้ว จากเซต T ที่เกิดขึ้นจะถูกใช้ในการหาฟรีเควนท์ไอเทมเซตทั้ง 2 แบบ (L_2 , OL_2) ของแบบจำลองนี้ และ ML โดย L_2 คือไอเทมเซตทั้งหมดจากเซต T ที่คุณลักษณะของเอกสารไม่ซ้ำกัน (Non overlap) OL_2 คือไอเทมเซตทั้งหมดจากเซต T ที่มีคุณลักษณะของเอกสารซ้ำกัน (Overlap) และ ML คือไอเทมเซตทั้งหมดใน L2 ที่มีชื่อกลุ่มเอกสารเหมือนกับไอเทมเซตใน OL_2 ตัวอย่างการเกิด L_2, OL_2 และ ML แสดงดังรูปที่ 2

โดยทั่วไปการหาฟรีเควนท์ไอเทมเซตจะมีเพียงฟรีเควนท์ไอเทมเซตเดียวคือ L_k แต่สำหรับแบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่มมีฟรีเควนท์ไอเทมเซต 2 ฟรีเควนท์ไอเทมเซต คือ L_k และ OL_k ซึ่งวิธีการสร้างฟรีเควนท์ไอเทมเซตของทั้ง 2 ฟรีเควนท์ไอเทมเซตแตกต่างกัน อธิบายได้ดังนี้

- การสร้าง OL_k ใช้วิธีการเชื่อมความสัมพันธ์ระหว่างไอเทมเซต โดยใช้วิธีการเดียวกับอัลกอริทึม apriori กล่าวคือ OL_k จะถูกใช้ในการหา $OL_{(k+1)}$ โดยที่ไอเทมเซตใน OL_k ที่เชื่อมความสัมพันธ์กันต้องมี (k-1) ไอเทมที่เหมือนกันและตำแหน่งไอเทมเซตเดียวกันจึงสามารถเชื่อมความสัมพันธ์ระหว่างไอเทมเซตได้ ดังแสดงวิธีการเชื่อมความสัมพันธ์ดังรูปที่ 3 จากรูปเป็นการเชื่อมความสัมพันธ์ระหว่าง OL_3 และ OL_3 เพื่อหา OL_4
- การสร้าง L_k ใช้วิธีการเชื่อมความสัมพันธ์ระหว่างไอเทมเซตด้วยวิธีใหม่ อธิบายได้ดังนี้คือ จำนวนไอเทมของไอเทมเซตไม่จำเป็นต้องจำนวนเท่ากัน และการพิจารณาว่าสามารถเชื่อมความสัมพันธ์กันได้หรือไม่นั้น พิจารณาเพียงแค่ไอเทมแรกเท่านั้น หากไอเทมแรกของไอเทมเซตเหมือนกันก็สามารถเชื่อมความสัมพันธ์กันได้ แต่ถ้าไอเทมแรกของไอเทมเซตไม่เหมือนกันก็ไม่สามารถเชื่อมความสัมพันธ์กันได้ โดย L_k จะเกิดจาก $OL_{(k-1)}$ เชื่อมความสัมพันธ์กับ ML ดังแสดงในรูปที่ 4 จากรูปเป็นการเชื่อมความสัมพันธ์ระหว่าง OL_4 และ ML เพื่อหา L_5

จากอัลกอริทึมการจัดประเภทเอกสารแบบหลายกลุ่มจะได้กฎความสัมพันธ์ 2 กลุ่มคือ Single_Rules และ Multi_Rules ซึ่ง Single_Rules นำมาใช้จำแนกเอกสารที่เป็นกลุ่มเดียว และ Multi_Rules นำมาใช้จำแนกเอกสารที่เป็นหลายกลุ่ม แต่อย่างไรก็ตามกฎใน Single_Rules ยังไม่สามารถนำมาใช้ในการจำแนกเอกสารที่ครอบคลุมทั้งหมดได้ ต้องนำกฎที่ได้ใน Single_Rules มาแปลงให้อยู่ในรูปแบบที่อยู่ในระดับ

3. แบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่ม

แบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่ม เป็นวิธีใหม่สำหรับการจัดประเภทเอกสารที่จะนำเสนอในหัวข้อนี้ โดยมีวัตถุประสงค์เพื่อลดจำนวนกฎให้มีจำนวนน้อยลง และกฎที่ได้ไม่ขัดแย้งกัน ซึ่งผลที่ได้ทำให้สามารถคัดแยกเอกสารทั้งที่เป็นกลุ่มเดียวและหลายกลุ่มได้อย่างถูกต้อง

3.1 การหากฎความสัมพันธ์

โดยทั่วไปของการหากฎความสัมพันธ์จะมีเซตของข้อมูลหลัก ๆ อยู่ 2 เซตคือแคนดิเดตท์ไอเทมเซต (Candidate itemset) และฟรีเควนท์ไอเทมเซต แต่สำหรับแบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่มในงานวิจัยนี้มีเซตข้อมูลมากกว่าที่กล่าวมา คือ T, OL และ ML

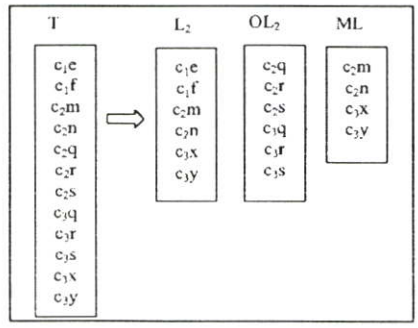
แบบจำลองการจำแนกกลุ่มเอกสารแบบหลายกลุ่มประกอบด้วยกลุ่มไอเทมเซตต่าง ๆ ดังนี้ เซต T เกิดจากการเชื่อมความสัมพันธ์ (Join) ระหว่าง L_0 และ L_1 โดย L_0 เป็นไอเทมเซตที่เป็นชื่อประเภทเอกสารที่ผ่านค่าสนับสนุนที่น้อยที่สุดแล้ว และ L_1 เป็นไอเทมเซตที่เป็นคุณลักษณะของเอกสารที่ผ่านค่าสนับสนุนที่น้อยที่สุดแล้ว จากเซต T ที่เกิดขึ้นจะถูกใช้ในการหาฟรีเควนท์ไอเทมเซตทั้ง 2 แบบ (L_2 , OL_2) ของแบบจำลองนี้ และ ML โดย L_2 คือไอเทมเซตทั้งหมดจากเซต T ที่มีคุณลักษณะของเอกสารไม่ซ้ำกัน (Non overlap) OL_2 คือไอเทมเซตทั้งหมดจากเซต T ที่มีคุณลักษณะของเอกสารซ้ำกัน (Overlap) และ ML คือไอเทมเซตทั้งหมดใน L2 ที่มีชื่อกลุ่มเอกสารเหมือนกับไอเทมเซตใน OL_2 ตัวอย่างการเกิด L_2, OL_2 และ ML แสดงดังรูปที่ 2

โดยทั่วไปการหาฟรีเควนท์ไอเทมเซตจะมีเพียงฟรีเควนท์ไอเทมเซตเดียวคือ L_k แต่สำหรับแบบจำลองการจำแนกประเภทเอกสารแบบหลายกลุ่มมีฟรีเควนท์ไอเทมเซต 2 ฟรีเควนท์ไอเทมเซต คือ L_k และ OL_k ซึ่งวิธีการสร้างฟรีเควนท์ไอเทมเซตของทั้ง 2 ฟรีเควนท์ไอเทมเซตแตกต่างกัน อธิบายได้ดังนี้

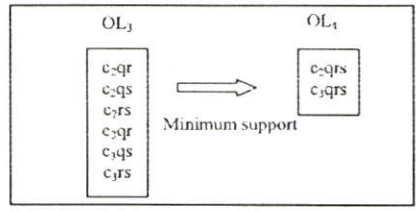
- การสร้าง OL_k ใช้วิธีการเชื่อมความสัมพันธ์ระหว่างไอเทมเซต โดยใช้วิธีการเดียวกับอัลกอริทึม apriori กล่าวคือ OL_k จะถูกใช้ในการหา $OL_{(k+1)}$ โดยที่ไอเทมเซตใน OL_k ที่เชื่อมความสัมพันธ์กันต้องมี (k-1) ไอเทมที่เหมือนกันและตำแหน่งไอเทมเซตเดียวกันจึงสามารถเชื่อมความสัมพันธ์ระหว่างไอเทมเซตได้ ดังแสดงวิธีการเชื่อมความสัมพันธ์ดังรูปที่ 3 จากรูปเป็นการเชื่อมความสัมพันธ์ระหว่าง OL_3 และ OL_3 เพื่อหา OL_4
- การสร้าง L_k ใช้วิธีการเชื่อมความสัมพันธ์ระหว่างไอเทมเซตด้วยวิธีใหม่ อธิบายได้ดังนี้คือ จำนวนไอเทมของไอเทมเซตไม่จำเป็นต้องจำนวนเท่ากัน และการพิจารณาว่าสามารถเชื่อมความสัมพันธ์กันได้หรือไม่นั้นพิจารณาเพียงแค่ไอเทมแรกเท่านั้น หากไอเทมแรกของไอเทมเซตเหมือนกันก็สามารถเชื่อมความสัมพันธ์กันได้ แต่ถ้าไอเทมแรกของไอเทมเซตไม่เหมือนกันก็ไม่สามารถเชื่อมความสัมพันธ์กันได้ โดย L_k จะเกิดจาก $OL_{(k-1)}$ เชื่อมความสัมพันธ์กับ ML ดังแสดงในรูปที่ 4 จากรูปเป็นการเชื่อมความสัมพันธ์ระหว่าง OL_4 และ ML เพื่อหา L_5

จากอัลกอริทึมการจัดประเภทเอกสารแบบหลายกลุ่มจะได้กฎความสัมพันธ์ 2 กลุ่มคือ Single_Rules และ Multi_Rules ซึ่ง Single_Rules นำมาใช้จำแนกเอกสารที่เป็นกลุ่มเดียว และ Multi_Rules นำมาใช้จำแนกเอกสารที่เป็นหลายกลุ่ม แต่อย่างไรก็ตามกฎใน Single_Rules ยังไม่สามารถนำมาใช้ในการจำแนกเอกสารที่ครอบคลุมทั้งหมดได้ ต้องนำกฎที่ได้ใน Single_Rules มาแปลงให้อยู่ในรูปแบบที่อยู่ในระดับ

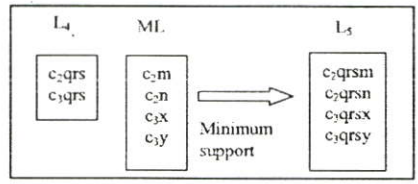
ที่สูงกว่าที่เรียกว่าเจอเนอรอลไลซ์เซชัน (Generalization) โดยกฎที่ได้แสดงดังรูปที่ 6



รูปที่ 2. ตัวอย่างไอเทมเซต T, L_2, OL_2 และ ML



รูปที่ 3. การหาฟรีเควนท์ไอเทมเซต OL_k



รูปที่ 4. การหาฟรีเควนท์ไอเทมเซต L_k

(1) $C_0 \leftarrow$ {Category and their support}
(2) $L_0 \leftarrow \{c \in C_0 \mid support > min\_support\}$
(3) $C_1 \leftarrow$ {Feature 1-itemset and their support}
(4) $L_1 \leftarrow \{c \in C_1 \mid support > min\_support\}$
(5) $C_2 \leftarrow$ Gen_2_Itemsets (L_0 , L_1)
(6) $T \leftarrow \{c \in C_2 \mid support > min\_support\}$
(7) $OL_2 \leftarrow$ Find_Overlap_Itemset (T)
(8) $L_2 \leftarrow T\text{-}OL_2$
(9) $DB_2 \leftarrow$ Cut_Useless_DB_Record (DB_1, OL_2)
(10) ML $\leftarrow$ Select_Itemset(OL_2,L_2)

(11) For (k=3; $OL_k \neq \phi$; k++)
(12) $\{C_k \leftarrow$ Gen_k_Itemsets (OL_{k-1} , ML)
(13) $L_k \leftarrow \{c \in C_k \mid support > min\_support\}$
(14) $OL_k \leftarrow$ Gen_Overlap_Itemsets (OL_{k1})
(15) $DB_k \leftarrow$ Cut_Useless_DB_Record (DB_{k1} , OL_k)}
(16) Single_SetOfRule $\leftarrow \{L_2, L_3, \ldots, L_k\}$
(17) Multi_SetOfRule $\leftarrow \{OL_2, OL_3, \ldots, OL_k\}$
(18) Single_Rules = ϕ
(19) Multi_Rules = ϕ
(20) For each item in Single_SetOfRule do
(21) {Single_Rules $\leftarrow t_1 \wedge t_2 \wedge \ldots t_{k-1} \Rightarrow C_i \mid$ t is feature of document and C is label of class, i $\in$ 1,2,…,p }
(22) end
(23) For each item in Multi_SetOfRule do
(24) {Multi_Rules $\leftarrow t_1 \wedge t_2 \wedge \ldots t_{k-1} \Rightarrow C_i, C_j, \ldots C_p \mid$ t is feature of document and C is label of a class, i $\in$ 1,2,…,p }
(25) end

รูปที่ 5. อัลกอริทึมการจัดประเภทเอกสารแบบหลายกลุ่ม

$a \wedge * \Rightarrow C_1$
$o \wedge p \wedge m \wedge * \Rightarrow C_2$
$o \wedge p \wedge v \wedge * \Rightarrow C_3$

รูปที่ 6. ตัวอย่างกฎความสัมพันธ์ที่เจอเนอรอลไลซ์เซชัน

จากรูปที่ 6 อธิบายความหมายของกฎที่ได้ดังนี้ $o \wedge p \wedge v \wedge * \Rightarrow C_3$ หมายถึงเอกสารที่จะบอกได้ว่าอยู่ประเภท C_3 ต้องมีคุณลักษณะ o,p,v และคุณลักษณะอะไรก็ได้ กล่าวคือมีเพียงแค่ 3 คุณลักษณะนี้ก็สามารถระบุได้ว่าเอกสารนี้อยู่ประเภท C_3

3.2 การจำแนกเอกสาร

การพิจารณาการจำแนกเอกสารที่เป็นกลุ่มเดียวใช้กฎจาก Single_Rules ที่ผ่านขบวนการเจอเนอรอลไลซ์เซชันแล้ว โดยพิจารณาคุณลักษณะเอกสารที่ต้องการจำแนกกลุ่มว่ามีคุณลักษณะตรงกับส่วนด้านซ้ายทุกคุณลักษณะของกฎหรือไม่ ถ้าหากใช่ให้จำแนกกลุ่ม

ตารางที่ 2. แสดงตัวอย่างกฎที่ใช้สำหรับจำแนกเอกสารกลุ่มเดียว

จำนวน	กฎที่ใช้ในการจำแนกเอกสารแบบกลุ่มเดียว
1	$a \wedge * \Rightarrow C_1$
2	$o \wedge m \wedge * \Rightarrow C_2$
3	$o \wedge p \wedge u \wedge * \Rightarrow C_3$

ตารางที่ 3. แสดงตัวอย่างกฎสำหรับจำแนกเอกสารแบบหลายกลุ่ม

จำนวน	กฎที่ใช้จำแนกเอกสารแบบหลายกลุ่ม
1	$o \Rightarrow C_2, C_3$
2	$p \Rightarrow C_2, C_3$
3	$o \wedge p \Rightarrow C_2, C_3$

ตารางที่ 4. ข้อมูลที่ใช้ทดสอบ

ชุดข้อมูล	สมาชิกแต่ละกลุ่ม			
	กลุ่มที่ 1	กลุ่มที่ 2	กลุ่มที่ 3	กลุ่มที่ 2 และ 3
1	350	300	310	40
2	317	330	300	53
3	318	325	320	37

จากการทดลองผลการจำแนกเอกสารมีความถูกต้อง 100 % แสดงในตารางที่ 5

5. สรุปผลและงานที่จะทำต่อไป

งานวิจัยนี้เป็นการเสนอแนวทางในการหากฎความสัมพันธ์เพื่อจัดประเภทเอกสารแบบหลายกลุ่ม โดยฟรีเควนท์ไอเทมเซตที่ใช้ในการหากฎความสัมพันธ์มี 2 ประเภท คือฟรีเควนท์ไอเทมสำหรับสร้างกฎความสัมพันธ์เพื่อจัดประเภทเอกสารแบบกลุ่มเดียว และฟรีเควนท์ไอเทมสำหรับสร้างกฎความสัมพันธ์เพื่อจัดประเภทเอกสารแบบหลายกลุ่ม จากกฎความสัมพันธ์ที่ได้สามารถนำมาจำแนกประเภทเอกสารได้อย่างถูกต้อง

ในการดำเนินงานต่อไปคือการนำแบบจำลองที่ได้ไปทดสอบกับข้อมูลมาตรฐานสำหรับการจัดประเภทเอกสาร นั่นคือข้อมูลข่าวรอยเตอร์-21578[1]

7. เอกสารอ้างอิง

[1] David D. Lewis, Reuters-21578 text categorization test collection distribution 1.0., http://www.daviddlewis.com/resources/testcollections/reuters21

[2] Fabrizio Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, Vol.34 No.1, Australian Computer Society,Inc., pp. 1-47, 2002.

[3] Jiawei Han, and Micheline Kamber, Data Mining: Concepts and Techniques, CA : Morgan kaufmann, San Francisco, 2001.

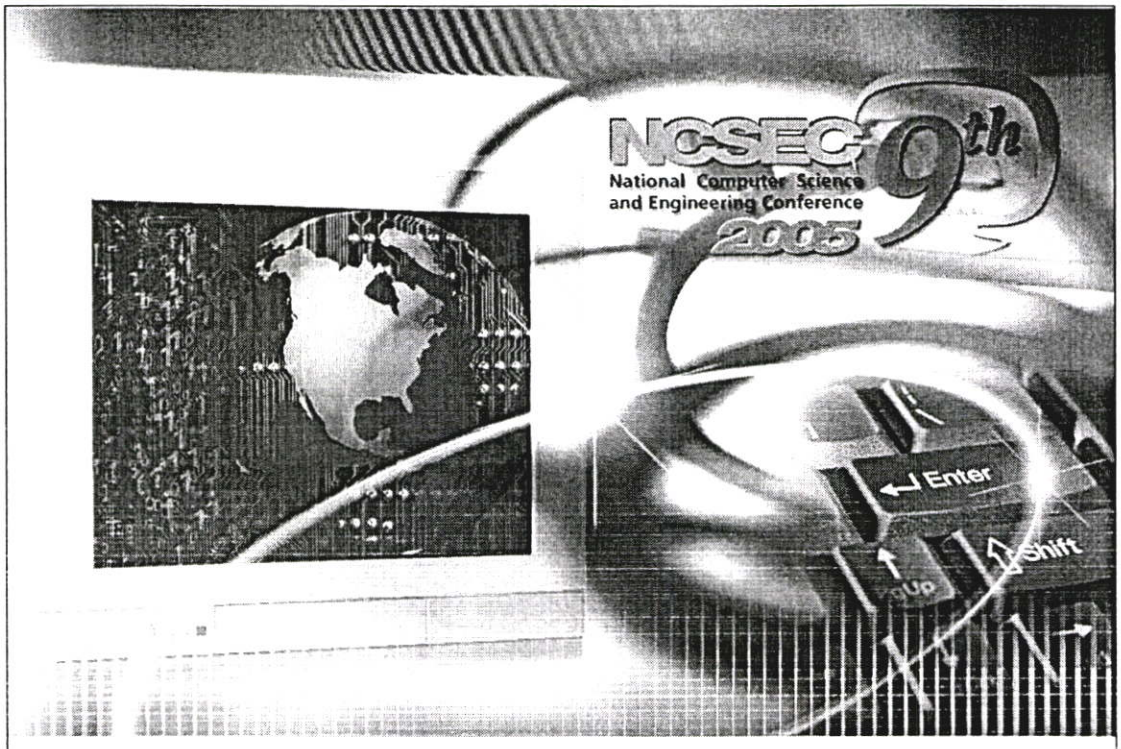
[4] Osmar R. Zaiane, and Maria-Luiza Antonie, "Classifying Text Documents by Association Terms with Text Categories," Proceedings of the thirteenth Australasian conference on Database technologies – Volume 5, Australian Computer Society,Inc., pp. 215-222. 2002.

[5] Osmar R. Zaiane, and Maria-Luiza Antonie, "Text Document Categorization by Term Association," Proceedings of ICDM 2002, IEEE, pp. 19-26, 2002.

[6] Tom M. Mitchell, Machine learning, McGraw-Hill, New York, 1997.

ตารางที่ 5. แสดงผลการจำแนกเอกสาร

ชุดข้อมูล	สมาชิกแต่ละกลุ่ม				ผลการจำแนกประเภทเอกสาร				ความถูกต้อง
	กลุ่มที่ 1	กลุ่มที่ 2	กลุ่มที่ 3	กลุ่มที่ 2 และ 3	กลุ่มที่ 1	กลุ่มที่ 2	กลุ่มที่ 3	กลุ่มที่ 2 และ 3	
1	350	300	310	40	350	300	310	40	100%
2	317	330	300	53	317	330	300	53	100%
3	318	325	320	37	318	325	320	37	100%



The 9th National Computer Science and Engineering Conference

October 27-28, 2005

University of Thai Chamber of Commerce, Bangkok Thailand

Organized by:

Department of Computer Engineering, School of Engineering,
University of Thai Chamber of Commerce

In Cooperation with:

Electrical Engineering; Electronics, Computer, Telecommunications and Information Technology Association of Thailand (ECTI)

IEEE Communications Society, Thailand Chapter

Sponsored by:

University of Thai Chamber of Commerce

National Electronics and Computer Technology Center (NECTEC)

Sun Microsystems (Thailand)

CS Loxinfo Public Company Limited

Pearson Education Indochina Limited

The OGA Group

Text Categorization using A New Association Rule-Based Text Classifier Algorithm

อัลกอริทึมใหม่สำหรับการจำแนกประเภทเอกสาร โดยใช้กฎความสัมพันธ์

สุภาภรณ์ บุตรดีวงษ์[1] และ วรพจน์ กรีสุระเดช[2]

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

Email: s5066003@kmitl.ac.th [1],worapoj@it.kmitl.ac.th [2]

บทคัดย่อ

งานวิจัยนี้นำเสนออัลกอริทึมใหม่สำหรับใช้จำแนกเอกสาร โดยใช้กฎความสัมพันธ์ มีวัตถุประสงค์เพื่อปรับปรุงการจำแนกเอกสารของอัลกอริทึม Association Rule-based Classifier By Categories (ARC-BC) และอัลกอริทึม Association Rule-based Classifier with All Categories (ARC-AC) โดยอัลกอริทึมใหม่ที่นำเสนอจะมีฟรีเควนท์ไอเทมเซต (Frequent itemsets) 2 ชนิดคือ (1) ฟรีเควนท์ไอเทมเซตที่เก็บคุณลักษณะ (Feature) ของเอกสาร ที่ไม่ซ้อนทับ (Overlap) กับกลุ่มใด ๆ ใช้สัญลักษณ์แทนด้วย L_k (2)ฟรีเควนท์ไอเทมเซตที่เก็บคุณลักษณะของเอกสารที่ซ้อนทับกันหลายกลุ่ม ใช้สัญลักษณ์แทนด้วย OL_k ซึ่งงานวิจัยนี้ยังได้นำเสนอวิธีการเชื่อมความสัมพันธ์ (Join) แบบใหม่ระหว่างไอเทมเซต (Itemsets) ของ OL_k ผลการทดลองที่ได้แสดงค่าความถูกต้องในการจำแนกประเภทเอกสารได้ดีกว่าอัลกอริทึม ARC-BC และ ARC-AC

Abstract

This paper proposes a new Association Rule-Based Text Classifier algorithm to improve the prediction accuracy of Association Rule-based Classifier By Categories (ARC-BC) algorithm and Association Rule-based with All Categories (ARC-AC) algorithm. Unlike the pervious algorithms, the proposed association rule generation algorithm constructs two types of frequent itemsets. The first frequent itemstes, i.e. L_k contain all term that have no an overlap with other categories. The second frequent itemsets, i.e. OL_k contain all features that have an overlap with other categories. In addition, this paper also proposed a new join operation for the second frequent itemstes. The experimental results are shown a good performance of the proposed classifier.

Key Words: Text categorization; Classification; Association rule discovery; Text mining.

1. บทนำ

การวิจัยเกี่ยวกับการจำแนกประเภทเอกสารเริ่มต้นเมื่อต้นปี ค.ศ. 1960 จนมาถึงปัจจุบัน ซึ่งยังมีการวิจัยค้นคว้าเพื่อปรับปรุงวิธีการในการจัดประเภทเอกสารให้ดีขึ้นกว่าเดิม ทั้งนี้เนื่องจากการเพิ่มขึ้นอย่างรวดเร็วของข้อมูลอิเล็กทรอนิกส์บนเครือข่ายอินเทอร์เน็ต และในองค์กรอื่น ๆ วิธีการในการจำแนกประเภทเอกสารมีด้วยกันหลายวิธี เช่น โครงข่ายประสาทเทียม (Neural networks) แบบจำลองความน่าจะเป็น (Probabilistic model) และการค้นหากฎความสัมพันธ์ (Association rule discovery) เป็นต้น

การค้นหากฎความสัมพันธ์เป็นวิธีหนึ่งที่ใช้ในการจำแนกประเภทเอกสาร วิธีนี้สามารถทำงานได้กับฐานข้อมูลขนาดใหญ่ ไม่มีการคำนวณที่ซ้ำซ้อนมาก และผลลัพธ์ที่ได้อยู่ในรูปของกฎความสัมพันธ์จึงง่ายต่อการทำความเข้าใจ ซึ่งวิธีการนี้ได้มีการพัฒนาขึ้นได้แก่อัลกอริทึม Association Rule-based Classifier by Categories (ARC-BC) [1] และอัลกอริทึม Association Rule-based Classifier with All Categories (ARC-AC) [2] โดยทั้งสองวิธีนี้ได้พัฒนาขึ้นมาจนมีความสามารถในการจำแนกประเภทเอกสารได้ดี แต่อย่างไรก็ตาม อัลกอริทึมทั้งสองที่กล่าวมาไม่สามารถทำนายกลุ่มเอกสารในกรณีที่เอกสารนั้นมีคุณลักษณะของเอกสารซ้อนทับกันได้ดี ดังนั้นงานวิจัยนี้ จึงนำเสนออัลกอริทึมใหม่สำหรับการจำแนกประเภทเอกสาร โดยใช้กฎความสัมพันธ์ เพื่อแก้ไขปรับปรุงการทำนายกลุ่มเอกสารของอัลกอริทึม ARC-BC และ ARC-AC ให้ดีขึ้น

เนื้อหาของบทความงานวิจัยนี้ประกอบไปด้วย 5 หัวข้อ ได้แก่ หัวข้อที่ 1 บทนำ หัวข้อที่ 2 การจำแนกเอกสารด้วยกฎความสัมพันธ์ หัวข้อที่ 3 อัลกอริทึมใหม่สำหรับการจำแนกประเภทเอกสาร หัวข้อที่ 4 ผลการทดลอง และหัวข้อที่ 5 เป็นการสรุปผลและงานที่จะทำต่อไป

2. ความรู้พื้นฐาน

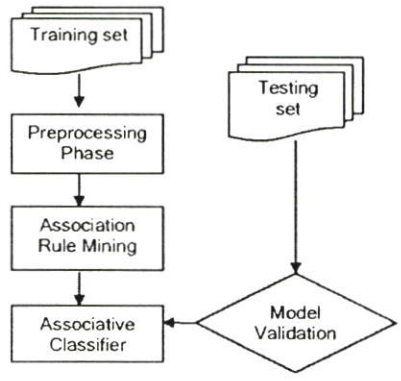
การหากฎความสัมพันธ์ เป็นเทคนิคของดาต้าไมนิ่งในการค้นหาความสัมพันธ์ระหว่างรายการในแต่ละรายการหรือกลุ่มของรายการ ที่ปรากฏขึ้นในฐานข้อมูล ความสัมพันธ์ที่ได้จะสามารถบอกลักษณะของข้อมูลหรือทำนายลักษณะของข้อมูลต่อไปได้ โดยทั่วไปความสัมพันธ์จะปรากฏอยู่ในรูปของกฎ "ถ้า ... แล้ว ..." (If ... Then ...) ในกฎหนึ่ง ๆ ประกอบด้วย 2 ส่วนคือ ส่วนด้านซ้ายของกฎ (ส่วน "ถ้า") และส่วนด้านขวาของกฎ (ส่วน "แล้ว") โดยส่วนด้านซ้ายอาจประกอบด้วยหนึ่งหรือมากกว่าหนึ่งเงื่อนไขที่เป็นจริง ที่จะทำให้ส่วนด้านขวาของกฎเป็นจริง เช่น "ถ้า T แล้ว C" ใช้สัญลักษณ์แทน "T => C" หมายถึง ถ้าเกิด T แล้วจะเกิด C ด้วย หากนำมาใช้สำหรับการจำแนกประเภทเอกสารจะคือความได้ว่า ถ้าเอกสารมีคุณลักษณะ T แล้วจะบอกได้ว่าเอกสารอยู่กลุ่ม C โดยที่ T เป็นคุณลักษณะของเอกสารและ C เป็นชื่อกลุ่มเอกสาร

การจำแนกประเภทเอกสาร เป็นการจัดกลุ่มเอกสารใหม่ที่เข้ามาในระบบไปยังกลุ่มที่เหมาะสมสำหรับเอกสารนั้น ๆ ซึ่งวิธีการจำแนกประเภทเอกสารโดยทั่วไปมีขั้นตอนการดำเนินงาน 2 ขั้นตอน คือ ขั้นตอนการฝึกหัดระบบ (Training step) และขั้นตอนการทดสอบระบบ (Testing step) การฝึกหัดระบบเป็นกระบวนการสร้างแบบจำลอง (Model) ระบบจำแนกประเภทเอกสาร ซึ่งเป็นการเรียนรู้ (Learning) ถึงแบบแผนของข้อมูลจากกลุ่มข้อมูลสำหรับฝึกหัดระบบ เมื่อได้แบบจำลองระบบแล้วขั้นตอนต่อไปคือนำแบบจำลองที่ได้มาทดสอบหาความถูกต้องของระบบก่อนที่จะนำไปใช้งาน

การจำแนกประเภทเอกสารโดยใช้กฎความสัมพันธ์สามารถแสดงได้ดังรูปที่ 1 โดยแบ่งขั้นตอนการทำงานออกเป็น 2 ขั้นตอนคือการทดสอบระบบ และขั้นตอนการทดสอบระบบ ในส่วนการทดสอบระบบ เริ่มจากการนำชุดข้อมูลสำหรับทดสอบระบบ (Training set) ซึ่งเป็นกลุ่มเอกสารที่แต่ละเอกสารประกอบด้วยชื่อกลุ่มและคุณลักษณะเอกสาร (Feature) นำเอกสารนี้ไปทำการเตรียมข้อมูล (Preprocessing Phase) โดยการนำคุณลักษณะของเอกสารไปตัดคำที่ไม่เหมาะที่จะนำมาเป็นตัวแทนเอกสาร (Stemming word and stop word) จะได้ลักษณะของแต่ละเอกสารคือ

$$D_i = \{c_1, c_2, \ldots, c_m, t_1, t_2, \ldots, t_m\}$$

โดยที่ D_i เป็นเอกสารตัวที่ i ประกอบไปด้วยชื่อกลุ่มเอกสาร c_1 จนถึง c ที่ m ตัว และประกอบด้วยคุณลักษณะ t_1 จนถึง t ที่ m ตัว หลังจากนั้นนำไปค้นหากฎความสัมพันธ์ (Association Rule Mining) โดยกลุ่มของกฎความสัมพันธ์ที่ใช้นำการจำแนกประเภทเอกสารต้องมีลักษณะคือกฎทางด้านซ้าย (Antecedence) ต้องเป็น



รูปที่ 1. แสดงวิธีการจำแนกประเภทเอกสารโดยใช้กฎความสัมพันธ์ [1]

คุณลักษณะของเอกสาร และทางด้านขวาของกฎ (Consequence) เป็นชื่อกลุ่มเอกสาร หลังจากนั้นจะเป็นการทดสอบ ตัวจำแนกเอกสาร (Associative Classifier) ที่ได้ โดยนำเอกสารสำหรับทดสอบระบบ (Testing set) มาทดสอบตัวจำแนกเอกสารว่าทำงานได้ถูกต้องหรือไม่

2.1 อัลกอริทึม ARC-BC และ ARC-AC

อัลกอริทึม ARC-BC และ ARC-AC [1,2] เป็นวิธีการสร้างตัวจำแนกประเภทเอกสารโดยใช้กฎความสัมพันธ์ ซึ่งวิธีการสร้างกฎความสัมพันธ์ของทั้งสองวิธีนี้ใช้แนวคิดต่างกัน โดยที่อัลกอริทึม ARC-AC เป็นการค้นหาความสัมพันธ์ของข้อมูลโดยมองชุดข้อมูลรวมกันเป็นชุดเดียว แล้วแยกข้อมูลเป็นชื่อกลุ่มเอกสาร และกลุ่มของคุณลักษณะเอกสาร แล้วค้นหากฎความสัมพันธ์ของข้อมูลโดยใช้แนวทางเดียวกับอัลกอริทึม Apriori ซึ่งผลลัพธ์ที่ได้ในบางลักษณะของข้อมูลจะทำให้ได้ผลลัพธ์ไม่ดี กรณีที่กลุ่มข้อมูลมีขนาดเล็ก หรือกรณีที่ข้อมูลมีความสัมพันธ์กันหลายกลุ่ม ดังนั้นจึงได้คิดอัลกอริทึม ARC-BC เพื่อแก้ไขปัญหาเหล่านี้ซึ่งวิธีการค้นหากฎความสัมพันธ์ต่างกับ ARC-AC โดยที่ ARC-BC มองชุดข้อมูลแยกเป็นแต่ละเอกสาร แล้วนำมาค้นหาความสัมพันธ์ของข้อมูลโดยใช้แนวทางเดียวกับอัลกอริทึม Apriori ซึ่งจะได้ความสัมพันธ์มากกว่าอัลกอริทึม ARC-AC ซึ่งกฎความสัมพันธ์ที่ได้สามารถแก้ไขปัญหาของ ARC-AC ได้

แต่อย่างไรก็ตาม กฎความสัมพันธ์ที่ได้จากอัลกอริทึม ARC-BC สามารถนำมาจำแนกประเภทเอกสารได้ หากแต่จะมีปัญหาที่ไม่สามารถจำแนกได้ถูกต้องเมื่อเอกสารนั้นมีคุณลักษณะซ้ำกับเอกสารกลุ่มอื่น ตัวจำแนกประเภทเอกสารจากวิธีนี้จะไม่สามารถจำแนกได้ถูกต้อง ดังนั้นงานวิจัยนี้จึงได้คิดค้นวิธีแก้ไขวิธีการค้นหากฎความสัมพันธ์แบบใหม่เพื่อให้สามารถทำนายกลุ่มเอกสารได้ทั้งหมดไม่ว่าจะเป็นเอกสารที่ไม่มีคุณลักษณะซ้ำกับกลุ่มใด หรือมีคุณลักษณะซ้ำกับกลุ่มอื่น ๆ ก็ตาม โดยอัลกอริทึมใหม่สำหรับการจำแนกประเภทเอกสารโดยใช้กฎความสัมพันธ์ของงานวิจัยนี้จะนำเสนอในหัวข้อต่อไป

3. อัลกอริทึมใหม่สำหรับการจำแนกประเภทเอกสาร โดยใช้กฎความสัมพันธ์

ตัวจำแนกประเภทเอกสารโดยใช้กฎความสัมพันธ์ (Association Rule-Based Text Classifier: ARTC) เป็นวิธีใหม่สำหรับการจำแนกประเภทเอกสารที่จะนำเสนอในหัวข้อนี้ โดยมีวัตถุประสงค์เพื่อพัฒนาผลการทำนายกลุ่มเอกสารให้ถูกต้องมากขึ้น

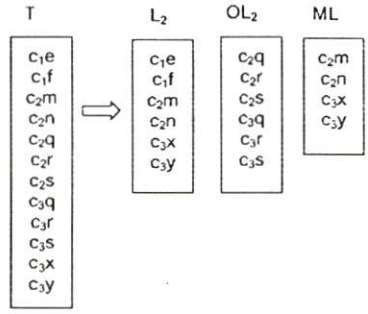
ในหัวข้อนี้แบ่งออกเป็น 3 ส่วนด้วยกันคือ วิธีการสร้างกฎความสัมพันธ์ (Association rule generation) การคัดกฎความสัมพันธ์ทิ้ง (Pruning set of association rules) และวิธีการจำแนกประเภทเอกสาร (Prediction of classes associated with new documents)

3.1 การสร้างกฎความสัมพันธ์

วิธีการโดยส่วนใหญ่ในการค้นหากฎความสัมพันธ์จะใช้อัลกอริทึม Apriori เป็นพื้นฐานในการค้นหากฎความสัมพันธ์ ซึ่งจะมีเซตของข้อมูลหลัก ๆ อยู่ 2 เซตคือ แคนดิเดตไอเทมเซต (Candidate Itemset) และฟรีเควนท์ไอเทมเซต แต่สำหรับ อัลกอริทึมของงานวิจัยนี้มีเซตข้อมูลมากกว่าที่กล่าวมา คือ T, OL และ ML โดยที่ T เกิดจากการเชื่อมความสัมพันธ์ (Join) ระหว่าง L_0 และ L_1 โดย L_0

เป็นไอเทมเซตที่เป็นชื่อประเภทเอกสารที่ผ่านค่าสนับสนุนที่น้อยที่สุดแล้ว และ L_1 เป็นไอเทมเซตที่เป็นคุณลักษณะของเอกสารที่ผ่านค่าสนับสนุนที่น้อยที่สุด (Minimum support) แล้ว จากเซต T ที่เกิดขึ้นจะถูกใช้ในการหาฟรีเควนท์ไอเทมเซตทั้ง 2 แบบ (L_2 , OL_2) ของงานวิจัยนี้ และ ML โดย L_2 คือไอเทมเซตทั้งหมดจากเซต T ที่คุณลักษณะของเอกสารไม่ซ้ำกัน (Non overlap) OL_2 คือไอเทมเซตทั้งหมดจากเซต T ที่มีคุณลักษณะของเอกสารซ้ำกัน (Overlap) และ ML คือไอเทมเซตทั้งหมดใน L_2 ที่มีชื่อกลุ่มเอกสารเหมือนกับไอเทมเซตใน OL_2 ตัวอย่างการเกิด L_2, OL_2 และ ML แสดงดังรูปที่ 2

จากรูปที่ 2 ให้ C = {c_1,c_2,c_3} โดยที่ C เป็นเซตของชื่อกลุ่ม และ T = {e,f,g,..,z} เป็นคุณลักษณะของเอกสาร L_2 ประกอบด้วยไอเทมเซตใน T ที่ไม่มีคุณลักษณะในไอเทม

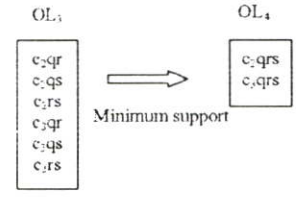


รูปที่ 2. แสดงตัวอย่างการเกิด L_2, OL_2 และ ML

เซตซ้ำกับไอเทมเซตอื่น เช่น c_1e พบว่าคุณลักษณะ e ไม่ซ้ำกับคุณลักษณะอื่นในไอเทมเซตในเซต T ดังนั้น c_1e จึงเป็นสมาชิก L_2 ที่ OL_2 ประกอบไปด้วยไอเทมเซตที่มีคุณลักษณะซ้ำกัน เช่น c_2q และ c_3q เป็นต้น และ ML ประกอบด้วยสมาชิกของ L_2 ที่มีชื่อกลุ่มเอกสารเหมือนกับ OL_2 จากรูป ที่ OL_2 มีชื่อกลุ่ม 2 ชื่อคือ c_2 และ c_3 ดังนั้นไอเทมเซตใน L2 ที่มี c_2 และ c_3 จึงนำเป็นสมาชิกใน ML

โดยทั่วไปการหาฟรีเควนท์ไอเทมเซตจะมีเพียงฟรีเควนท์ไอเทมเซตเดียวคือ L_k แต่สำหรับอัลกอริทึมของงานวิจัยนี้มีฟรีเควนท์ไอเทมเซต 2 ฟรีเควนท์ไอเทมเซต คือ L_k และ OL_k ซึ่งวิธีการสร้างฟรีเควนท์ไอเทมเซตของทั้ง 2 ฟรีเควนท์ไอเทมเซต อธิบายได้ดังนี้

- OL_k เกิดจากการเชื่อมความสัมพันธ์ระหว่างไอเทมเซต โดยใช้วิธีการเชื่อมความสัมพันธ์วิธีเดียวกับอัลกอริทึม Apriori กล่าวคือ OL_k จะถูกใช้ในการหา $OL_{(k+1)}$ โดยที่ไอเทมเซตใน OL_k ที่เชื่อมความสัมพันธ์กันต้องมี (k-1) ไอเทมเซตที่เหมือนกัน และตำแหน่งไอเทมเซตเดียวกันจึงสามารถเชื่อมความสัมพันธ์ระหว่างไอเทมเซตได้ ดังแสดงวิธีการเชื่อมความสัมพันธ์ดังรูปที่ 3 จากรูปที่ 3 เป็นการเชื่อมความสัมพันธ์ระหว่าง OL_3 และ OL_3 จะได้ OL_4 ในที่นี้ k คือ 3 ดังนั้น

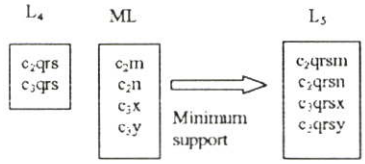


รูปที่ 3. แสดงการเชื่อมความสัมพันธ์แบบ Apriori

ต้องพิจารณา (3-1) ไอเทมใน OL_3 ว่าเหมือนกันหรือไม่หากเหมือนกันก็สามารถเชื่อมความสัมพันธ์กันได้ จากรูป $c_2$qr และ $c_2$qs สามารถเชื่อมความสัมพันธ์กันได้ เพราะ 2 ไอเทมแรกเหมือนกัน จะได้ $c_2$qrs เป็นสมาชิกใน OL_4 และ $c_2$qs และ $c_2$rs ไม่สามารถเชื่อมความสัมพันธ์กันได้เพราะ 2 ไอเทมแรกไม่เหมือนกัน

- L_k เกิดจากการเชื่อมความสัมพันธ์ระหว่างไอเทมเซตด้วยวิธีใหม่จากงานวิจัยนี้ ซึ่งอธิบายได้ดังนี้คือ จำนวนไอเทมของไอเทมเซตไม่

จำเป็นต้องมีจำนวนเท่ากัน และการพิจารณาว่าสามารถเชื่อมความสัมพันธ์กันได้หรือไม่นั้น พิจารณาเพียงแค่ไอเทมแรกเท่านั้น หากไอเทมแรกของไอเทมเซตไม่เหมือนกันก็ไม่สามารถเชื่อมความสัมพันธ์กันได้ โดย L_k จะเกิดจาก $OL_{(k-1)}$ เชื่อมความสัมพันธ์กับกลุ่มไอเทมเซต ML ดังแสดงในรูปที่ 4



รูปที่ 4. แสดงการเชื่อมความสัมพันธ์แบบใหม่ของงานวิจัย

จากรูปที่ 4 เป็นการเชื่อมความสัมพันธ์ระหว่าง L_4 และ ML จะได้ L_5 จากวิธีการเชื่อมความสัมพันธ์ของงานวิจัยนี้ ให้พิจารณาแค่ไอเทมแรกของ L_4 และ ML หากเหมือนกันก็เชื่อมความสัมพันธ์กันได้เป็นไอเทมเซตใน L_5 จากตัวอย่าง c_2qrs และ c_2m สามารถเชื่อมความสัมพันธ์กันได้เพราะไอเทมแรกของทั้งสองไอเทมเซตเหมือนกัน จะได้เป็นไอเทมเซตใหม่ คือ c_2qrsm และเช่นกันกรณี c_2qrs และ c_3x ไม่สามารถเชื่อมความสัมพันธ์กันได้เพราะไอเทมตัวแรกไม่เหมือนกัน

วิธีการค้นหากฎความสัมพันธ์ด้วยวิธีของงานวิจัยนี้แสดงดังรูปที่ 5 โดยที่ขั้นตอนที่ 1-10 เป็นการหาเซตของข้อมูลที่ใช้ในการหาฟรีเควนท์ไอเทมเซต ขั้นตอนที่ 11-15 เป็นการหาฟรีเควนท์ไอเทมเซตทั้ง L_k และ OL_k ในขั้นตอนที่ 16-20 เป็นการสร้างกฎความสัมพันธ์จากฟรีเควนท์ไอเทมเซตที่ได้ โดยสนใจกฎความสัมพันธ์ที่ทางด้านซ้ายของกฎบอกคุณลักษณะของเอกสารและทางด้านขวาของกฎบอกชื่อกลุ่มเอกสาร

(1) C_0 ← {Category and their support}
(2) L_0 ← {c ∈ C_0 | support > min_support}
(3) C_1 ← {Feature 1-itemset and their support}
(4) L_1 ←{c ∈ C_1 | support > min_support}
(5) C_2 ← Gen_2_Itemsets (L_0 , L_1)
(6) T ← {c ∈ C_2 | support > min_support}
(7) OL_2← Find_Overlap_Itemset (T)
(8) L_2 ←T-OL_2
(9) DB_2← Cut_Useless_DB_Record (DB_1, OL_2)
(10) ML← Select_Itemset(OL_2,L_2)
(11) For (k=3; $OL_k \neq \phi$; k++)
(12) {C_k ← Gen_k_Itemsets (OL_{k-1} , ML)
(13) L_k ← {c ∈ C_k | support > min_support}
(14) OL_k← Gen_Overlap_Itemsets (OL_{k-1})
(15) DB_k ←Cut_DB_Record ($DB_{(k-1)}$,OL_k) }
(16) SetOfRule ←{L_2, L_3, .. L_k,OL_2,OL_3,..,OL_k}
(17)Rules = ϕ
(18)For each item in SetOfRule do
(19) {Rules← $t_1 \wedge t_2 \wedge ... t_{k-1} \Rightarrow C_i$ | t is feature of document and C is label of class, i∈ 1,2,..,p }
(20)end

รูปที่ 5. อัลกอริทึมการค้นหากฎความสัมพันธ์

3.2 การคัดกฎความสัมพันธ์ทิ้ง

จากการค้นหากฎความสัมพันธ์ที่ได้ พบว่ามีกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสารเป็นจำนวนมาก หากนำไปใช้ในการจำแนกประเภทเอกสารเลยจะทำให้ประสิทธิภาพในการทำงานของระบบน้อยลง ดังนั้นจึงคิดวิธีการคัดกฎความสัมพันธ์ที่ได้ให้เหลือน้อยลง แต่ยังคงมีความสามารถในการทำนายกลุ่มเอกสารได้ถูกต้องเช่นเดิม ในส่วนนี้จึงจะกล่าวถึงวิธีการคัดกฎความสัมพันธ์ทิ้ง

การคัดกฎความสัมพันธ์ทิ้งเป็นการคัดกฎที่เฉพาะเกินไป (Specific rule) ทิ้ง และเก็บเฉพาะกฎที่ใช้กับกรณีทั่วไปได้ (General rule) ซึ่งมีวิธีการดังนี้

วิธีการคัดกฎความสัมพันธ์ทิ้ง : ให้กฎความสัมพันธ์ 2 กฎดังนี้ T_1 => C เป็นกฎที่ 1 และ T_2 => C เป็นกฎที่ 2 จะกล่าวได้ว่ากฎที่ 1 เป็นกฎที่ใช้กับกรณีทั่วไปได้มากกว่ากฎที่ 2 ก็ต่อเมื่อ $T_1 \subseteq T2$ ให้ตัดกฎที่ 2 ทิ้ง เก็บกฎที่ 1 ไว้ใช้ในการจำแนกประเภทเอกสาร ตัวอย่างเช่น

firewall ∧ web => Category2 (1)
web => Category2 (2)

จากตัวอย่างทั้ง 2 กฎ พบว่ากฎที่ 2 ใช้กับกรณีทั่วไปได้มากกว่ากฎที่ 1 ดังนั้นในกรณีนี้จึงตัดกฎที่ 1 ทิ้ง และเก็บกฎที่ 2 ไว้ใช้ในการจำแนกประเภทเอกสาร จะได้เซตของกฎความสัมพันธ์ใหม่ที่ใช้ในการจำแนกเอกสารได้

3.3 วิธีการจำแนกประเภทเอกสาร

การจำแนกประเภทเอกสาร เนื่องจากได้มีการคัดกฎความสัมพันธ์ทิ้งไป ดังนั้นจึงมีวิธีการจำแนกเอกสารใหม่เพื่อให้สามารถจำแนกเอกสารใหม่ให้ถูกต้อง โดยใช้กฎความสัมพันธ์ที่ได้หลังจากคัดกฎความสัมพันธ์ทิ้งแล้ว โดยให้ใช้กลุ่มของกฎความสัมพันธ์ที่ได้หลังจากการคัดทิ้งแล้วมาใช้ในการจำแนกเอกสาร นำเอกสารใหม่ที่ต้องการจำแนกประเภทไปเทียบกับกลุ่มกฎความสัมพันธ์ที่ใช้ในการจำแนกประเภทเอกสาร จะได้กลุ่มกฎความสัมพันธ์ที่ใช้ในการจำแนกเอกสารนี้ ให้พิจารณาการจำแนกเอกสารดังนี้

- กฎทางด้านซ้ายที่ได้ทั้งหมดอยู่กลุ่มเดียวกัน ให้จำแนกว่าเอกสารอยู่กลุ่มนั้น
- กฎทางด้านซ้ายที่ได้บอกว่าอยู่กลุ่มใดมากกว่า 1 กลุ่ม ให้พิจารณาดังนี้
  - พิจารณาว่ากฎของกลุ่มใดมีมากกว่าให้ทำนายว่าเอกสารอยู่กลุ่มนั้น เช่น ถ้ากฎที่บอกว่าเอกสารอยู่ Category2 มี 2 กฎ และกฎที่บอกว่าเอกสารอยู่ Category3 มี 3 กฎ ให้ทำนายว่าเอกสารนี้อยู่ Category3
  - หากจำนวนกฎของแต่ละกลุ่มเท่ากัน ให้พิจารณาที่ผลรวมของค่าความเชื่อมั่น (Confidence value) ของกลุ่มนั้น ๆ หากกลุ่มไหนมีผลรวมของค่าความเชื่อมั่นมากกว่า ให้ทำนายว่าเอกสารอยู่กลุ่มนั้น เช่น ถ้ากฎที่บอกว่าเอกสารอยู่ Category2 มี 2 กฎ และกฎที่บอกว่าเอกสารอยู่ Category3 มี 2 กฎ ให้นำค่าความเชื่อมั่นของแต่ละกฎในกลุ่มนั้นมารวมกันแล้วหารด้วยจำนวนกฎในกลุ่มนั้น จะได้เป็นผลรวมของค่าความเชื่อมั่นของกลุ่มนั้น หากผลรวมของค่าความเชื่อมั่นของ Categroy2 มากกว่า Category3 ให้ทำนายว่าแยกสารนี้อยู่ Category2
  - หากจำนวนกฎของแต่ละกลุ่ม และผลรวมค่าความเชื่อมั่นเท่ากันให้พิจารณาที่ค่าผลรวมสนับสนุนของแต่ละกลุ่ม โดยนำค่าสนับสนุนของแต่ละกฎมารวมกันแล้วหารด้วยจำนวนกฎในกลุ่มนั้น ๆ หากกลุ่มไหนมีผลรวมค่าสนับสนุนมากกว่าให้ทำนายว่าเอกสารอยู่กลุ่มนั้น

วิธีการจำแนกประเภทเอกสารอธิบายดังรูปที่ 6

```
(1)for each new documents
(2) R <- φ
(3)for each Rule
(4) if feature_new_doc ⊆ Rule
(5)     R = rule   % rule ∈Rule
(6) end
(7) group R by category: R1,R2,...,Rn
(8) for each R
(9)     sum the confidences of rules and divide
          by the number of rules in R
(10)    sum the support of rules and divide by the
          number of rules in R
(11)end
(12) if 1 group has highest number of rule
(13)    put the new document in the class that
          has highest number of rule in group.
(14)    if more 1 group has highest number of
          rule
(15)      put the new document in the class that
            has highest confidence sum.
(16)      if more 1 group has highest confidence
            sum
(17)        put the new document in the class
              that has highest support sum.
(18) end
(29) next new document
```

รูปที่ 6. อัลกอริทึมการจำแนกเอกสาร

จากรูปที่ 6 ขั้นตอนที่ 4-6 เป็นการหากฎความสัมพันธ์ที่สามารถใช้จำแนกเอกสารนี้ได้ ขั้นตอนที่ 7 เป็นการแบ่งกลุ่มของกฎความสัมพันธ์ที่ได้ ให้กฎที่อยู่กลุ่มเดียวกันอยู่ด้วยกัน ขั้นตอนที่ 8-11 หาค่าความเชื่อมั่นและค่าสนับสนุนของแต่ละกลุ่มกฎความสัมพันธ์ และขั้นตอนที่

11-18 เป็นการทำนายว่าเอกสารใหม่ควรอยู่กลุ่มใดโดยใช้กฎความสัมพันธ์ที่ได้เป็นตัวตัดสิน

4. ผลการทดลอง

ในส่วนนี้จะกล่าวถึงผลการทดลองของงานวิจัยนี้ ข้อมูลที่ใช้ทดลอง เป็นเอกสาร Computer Science Technical Report (CSTR) เป็นข้อมูลที่ใช้ในการจำแนกประเภทเอกสาร (Categorization) และการจัดกลุ่มเอกสาร (Clustering) [3, 4, 5, 6] โดยข้อมูล CSTR ประกอบด้วยเอกสารทั้งหมด 472 เอกสาร มี 4 กลุ่มเอกสาร รายละเอียดของแต่ละกลุ่ม แสดงดังตารางที่ 1 และตัวอย่างเอกสารแสดงดังรูปที่ 7

ตารางที่ 1. ลักษณะข้อมูล

ชื่อกลุ่ม	จำนวนเอกสาร
AI	107
Systems	132
Robotics and Vision	88
Theory	145

ในการทดลองเบื้องต้น นำคำสำคัญ (Keyword) ของเอกสารแต่ละเอกสารมาใช้แทนแต่ละเอกสาร โดยแบ่งข้อมูลในการเรียนรู้ระบบ (Training) และทดสอบระบบ (Testing) ดังตารางที่ 2

นำข้อมูลที่ได้เข้าสู่วิธีการสร้างกฎความสัมพันธ์โดยวิธีของ ARC-BC, ARC-AC และวิธีของงานวิจัยนี้ (ARTC) ได้ผลการทดลองดังตารางที่ 3 ซึ่งแสดงผลการจำแนกเอกสารของ ARC-BC, ARC-AC และ ARTC

จากตารางที่ 3 แสดงจำนวนเอกสารที่ใช้ในการทดสอบของแต่ละกลุ่ม ผลการทดลองมีผลการทดลองของ ARC-BC, ARC-AC และ ARTC แสดงจำนวนเอกสารที่อัลกอริทึมนั้น ๆ จำแนกได้ถูกต้อง และที่ความถูกต้อง (Accuracy rate) แสดงอัตราความถูกต้องในการจำแนกเอกสารของทั้งสองอัลกอริทึม โดยแสดงตามกลุ่ม และแสดงอัตราความถูกต้องโดยรวม พบว่าตามวิธีของงานวิจัยนี้ ให้ผลการจำแนกเอกสารที่ดีกว่า ARC-BC และ ARC-AC โดย ARC-BC และ ARC-AC ให้ความถูกต้องในการจำแนกเอกสารร้อยละ 80.33 และ ARTC ให้ความถูกต้องในการจำแนกเอกสารร้อยละ 95.08 ซึ่ง ARTC จำแนกเอกสารได้ถูกต้องมากกว่า

He, S., Gildea, D.J., "Semantic Labeling by Maximum Entropy Model", TR847, Computer Science Dept., U. Rochester, September 2004.
04.tr847rev.Semantic_labeling_by_maximum_entropy_model.pdf

Keywords: language understanding; maximum entropy; semantic roles.

In this paper, we present the results for semantic labeling, extending the work of [Gildea and Jurafsky, 2002], [Fleischman et al., 2003], [Pradhan et al., 2004], and others. The main labeling approach is based on Maximum Entroopy. We show the performance of the baseline system as well as those by applying coreference resolution, stemming and feature combinations to the feature files.

รูปที่ 7. ตัวอย่างเอกสาร

ตารางที่ 2. ข้อมูลที่ใช้ในการเรียนรู้และทดสอบระบบ

ชื่อกลุ่ม	จำนวนเอกสาร	
	เรียนรู้ระบบ	ทดสอบระบบ
AI	84	23
Systems	99	33
Robotics and Vision	61	27
Theory	106	39

5. สรุปผล และงานที่จะทำต่อไป

งานวิจัยนี้เป็นการเสนอวิธีการสร้างตัวจำแนกประเภทเอกสารโดยใช้กฎความสัมพันธ์ ซึ่งมีฟรีเควนท์ไอเทมเซต 2 ชนิดที่ใช้ในการสร้างกฎความสัมพันธ์คือ L_k เป็นฟรีเควนท์ไอเทมเซตที่เก็บคุณลักษณะของเอกสารที่ไม่ซ้อนทับกับกลุ่มใด ๆ และ OL_k เป็นฟรีเควนท์ไอเทมเซตที่เก็บคุณลักษณะของเอกสารที่ซ้อนทับกันหลายกลุ่ม โดยกฎความสัมพันธ์ที่ได้สามารถจำแนกเอกสารได้ถูกต้องมากกว่าอัลกอริทึม ARC-BC และ ARC-AC

แนวทางในการดำเนินงานต่อไปคือ การทดสอบกับข้อมูลชุดอื่น ๆ ต่อไป เพื่อทดสอบประสิทธิภาพอัลกอริทึมของงานวิจัยนี้

6. เอกสารอ้างอิง

[1] O. Zaiane, and M. Antonie, "Text Document Categorization by Term Associaton", Proceedings of ICDM 2002, IEEE, 2002, pp.19-26.

[2] O. Zaiane, and M. Antonie, "Classifying Text Documents by Association Terms with Text Categories", Proceedings of thirteenth Australasian conference on Database technologies, Australian Compter Society Inc., 2002, pp-215-222.

[3] T. Li, S. Zhu, and M. Ogihara, "Efficient Multi-way Text Categorization via Generalized Discriminate Analysis", Proceedings of the Twelfth International Conference on Information and Knowledge Management, 2003, pp. 317-324.

[4] T. Li, S. Zhu, and M. Ogihara, "Document clustering via asaptive subspace iteration", Proceedings of the 12th ACM International Conference on Multimedia, ACM, 2004, pp. 264-367.

[5] T. Li, S. Zhu, and M. Ogihara, "Entropy-Based criterion in Categorical Clustering", Proceedings of the 21st International Conference on Machine Learning, 2004, pp. 536-543.

[6] B. Tang, M. Shepheral, E. Milios, and M. Heywood, "Comparing and Combining Division Reduction Techniques for Efficient Text Clustering", International Workshop on Feature Selection for Data Mining: Interfacing Machine Learning and Statistics,2005.

[7] F. Sebastiani, Machine learning in automated text categorization, ACM Computing Surveys, 2002.

[8] T.M. Mitchell, Machine learning. McGraw Hill, New York, 1997.

[9] J. Han, and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco, 2001.

ตารางที่ 3. ผลการจำแนกเอกสาร

ชื่อกลุ่ม	เอกสารทดสอบระบบ	ผลการทดลอง			ความถูกต้อง (Accuracy rate)		
		ARC-BC	ARC-AC	ARTC	ARC-BC	ARC-AC	ARTC
AI	23	15	15	22	65.22 %	65.22 %	95.65 %
Systems	33	29	29	31	87.88 %	87.88 %	93.94 %
Robotics and Vision	27	19	19	25	70.37 %	70.37 %	92.59 %
Theory	39	35	35	38	89.74 %	89.74 %	97.43 %
รวม	122	98	98	116	80.33 %	80.33 %	95.08 %

IJCIM
INTERNATIONAL JOURNAL OF THE COMPUTER, THE INTERNET AND MANAGEMENT

Editor-in-Chief: Srisakdi Charmonman
www.charm.au.edu www.ijcim.th.org
Volume 13 Number SP2 October 2005 ISSN 0858-7027

Special Issue of IJCIM eIndustry 2005

ICIM 2005

Proceedings of the International Conference on Computer and Industrial Management

Srisakdi Charmonman IT Center
Assumption University, Bangna Campus, Thailand.
October 29 - 30, 2005

Assumption University Press

Text Categorization using a New Text Association Rule-Based Classifier

Supaporn Buddeewong and Worapoj Kreesuradej

Faculty of Information Technology
King mongkut's Institute of Technology Ladkrabang.
Bangkok, 10520 Thailand.

Abstract. This paper proposes a new association rule-based Classifier algorithm to improve the prediction accuracy of Association Rule-based Classifier By Categories (ARC-BC) algorithm. Unlike the previous algorithms, the proposed association rule generation algorithm constructs two types of frequent itemsets. The first frequent itemsets, i.e. L_k, contain all term that have no an overlap with other categories. The second frequent itemsets, i.e. OL_k, contain all features that have an overlap with other categories. In addition, this paper also proposes a new join operation for the second frequent itemsets. The experimental results are shown a good performance of the proposed classifier.

Keywords. Text Categorization; Text mining; Classification; Association rule.

1. Introduction

Text categorization task is defined as assigning category labels to new documents based on their contents. Text categorization research has a long history, starting in early 1960s. There are several text categorization approaches have been proposed such as neural networks, genetic algorithms and probabilistic models, support vector machine.

Recently, Association Rule-based Classifier By Categories (ARC-BC) algorithm [4, 5] that is based on association rule mining approach have been proposed. These classifiers have been proven to powerful. In addition, these classifiers produce clear and understandable results. However, an algorithm can not work well for the single-class document that has some terms of document mutually associated with other class. As a result, an algorithms may incorrectly classify those single-class documents. Therefore, this paper proposes a new association rule-based classifier algorithm to improve the prediction accuracy of ARC-BC algorithm.

This paper is organized as following. The second section gives the overview of a text categorization with association rule. In the third section, we introduce our new text categorization approach. Experimental results are described in Section 4. We summarize our research and future work in the fifth Section.

2. Text categorization with association rule

The construct process of an associative classifier is shown in figure 1.

Here, the training set is a document collection. A document D_i of the collection is assigned to a set of categories $C = \{c_1, c_2, c_3, \ldots, c_m\}$. At preprocessing phase, The set of term $T = \{t_1, t_2, t_3, \ldots, t_n\}$ of document D_i is retained after term pruning and stemming. Then, a document D_i is model as the following:

$$D_i = \{c_1, c_2, .., c_m, t_1, t_2, \ldots, t_n\}.$$

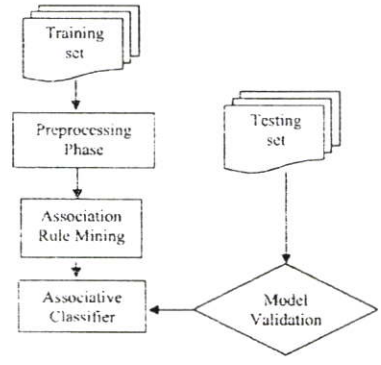


Figure 1. Association classification method [5]

The next step is the generation of association rule. At the associative rule mining, a set of rules that associate the terms of a document and its categories is extracted from the training data by using an apriori-based algorithm. However, the association rules are constrained in that the antecedence has to be a conjunction of term from T, while the consequence of the rule has member of C.

After generating the set of rule, an important step is building and validating an associative text classifier. Recently, Association Rule-based Classifier By Categories (ARC-BC)

algorithm is proposed to build an associative text classifier. Method of ARC-BC considers each set of documents belonging to one category as a separate text collection to generate association rules. If a document belongs to more than one category, this document will be present in each set associated with the categories that the document falls into. However, both algorithms may fail to classify a single-class document that has some terms of document mutually associated with other class. Therefore, in this paper, we propose a new text rule-based classifier to deal with such problem. The new text association rule-based classifier is presented in the next section.

3. Categorization using a new association rule-based classifier

A new algorithm for a association rule-based classifier is introduced. Our algorithm is proposed to deal with misclassifying problem of a single-class document that has some terms of document mutually associated with other class. Here, a new algorithm for association rule generation and a new categorization algorithm based on the new set of rules is proposed.

3.1. Association rule generation

Like typical apriori-based algorithms, our associative rules are generated from frequent itemsets. Therefore, frequent itemsets are constructed in order to get the set of associative rules. However, unlike the previous algorithms, our method constructs two types of frequent itemsets. The first frequent itemsets, i.e. L_k, contain all term that have no an overlap with other categories. The second frequent itemsets, i.e. OL_k, contain all features that have an overlap with other categories.

Basically, any frequent itemsets has to include a category label starting from 2-itmesets to k-itemsets. For the first frequent itemsets, L_k, is constructed using apriori join. As an example, L_k is constructed by using apriori join between L_{k-1} and L_{k-1}. In this paper, a new join operation for the second frequent itemsets, OL_k, is proposed.

For the new join operation, any two items in an itemset can be joined if they have the same category. The examples of both of apriori join shown in figure 2 and the new join shown in figure 3. For figure 2, figure 3 and figure 4. We use the model of document as

$$D_j = \{c_1, c_2, c_3, a, b, c, \ldots, z\}.$$

Document D_j consists of category label, c_1, c_2, c_3. Term of document are represent by character, a-z.

In addition, the algorithm, called ARTC, for generating association rules are shown in figure 4.

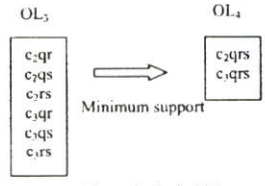


Figure 2. Apriori join

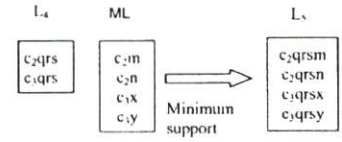


Figure 3. ARTC join

In our algorithm, T set is generated from L_0 and L_1 using apriori join. L_0 is a set of categories that satisfies by support threshold and L_1 is a set of terms that satisfy by support threshold. Then, L_2 is generated form T set. Next, frequent itemsets starting from L_3 to L_k are generated by OL_{k-1} join ML. ML includes all itemsets that have the same category in OL_k .Figure 3 shows an example of T, L_2, OL_2 and ML.

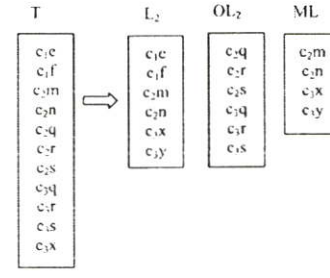


Figure 3. Sample of T, L_2, OL_2 and ML

(1) C_0 ← {Category and their support}
(2) L_0 ← {c ∈ C_0 | support > min_support}
(3) C_1 ← {Feature 1-itemset and their support}
(4) L_1 ← {c ∈ C_1 | support > min_support}
(5) C_2 ← Gen_2_Itemsets (L_0 , L_1)
(6) T ← {c ∈ C_2 | support > min_support}
(7) OL_2 ← Find_Overlap_Itemset (T)
(8) L_2 ← T-OL_2
(9) DB_2 ← Cut_Useless_DB_Record (DB_1, OL_2)
(10) ML ← Select_Itemset(OL_2,L_2)
(11) For (k=3; $OL_k \neq \phi$; k++)
(12) {C_k ← Gen_k_Itemsets (OL_{k-1} , ML)
(13) L_k ← {c ∈ C_k | support > min_support}
(14) OL_k ← Gen_Overlap_Itemsets (OL_{k-1})
(15) DB_k ← Cut_DB_Record ($DB_{(k-1)}$,OL_k) }
(16) SetOfRule ← {L_2, L_3, … L_k,OL_2,OL_3,….,OL_k }
(17)Rules = ϕ
(18)For each item in SetOfRule do
(19) {Rules ← $t_1 \wedge t_2 \wedge \ldots t_{k-1} \Rightarrow C_i$ | t is feature of document and C is label of class, i∈ 1,2,….,p }
(20)end

Figure 4. Algorithm for generating association rule

3.2. Prediction of classes association with new document

After obtaining a set of associative rules, an associative classifier makes use of the set of associative rules in the prediction of classes for new documents. The algorithm for classifying a new document is shown in figure 5. Basically, the algorithm take feature terms form each new document to compare with each rule in set of rule. If all feature terms are matched with the rule, that new document is classify by that rule.

```
(1)for each new documents
(2) R <- ϕ
(3)for each Rule
(4)       if feature_new_doc ⊆ Rule
(5)       R = rule    % rule ∈ Rule
(6) end
(7) group R by category: R1,R2,...,Rn
(8) for each R
(9)       sum the confidences of rules and divide
          by the number of rules in R
(10)      sum the support of rules and divide by the
          number of rules in R
(11)end
(12) if 1 group has highest number of rule
(13)      put the new document in the class that
          has highest number of rule in group.
(14)      if more 1 group has highest number of  rule
(15)        put the new document in the class that
            has highest confidence sum.
(16)      if more 1 group has highest confidence  sum
(17)        put the new document in the class
            that has highest support sum.
(18) end
(29) next new document
```

Figure 5. Algorithm for classifying a new document.

4. Experimental results

In this section, the experimental results of document classification using text rule-base classifier are reported.

For our experiment, we use data generated dataset and Computer Science Technical Report collection dataset (CSTR) dataset for test our algorithm. To evaluate the proposed algorithm, we use classification accuracy rate of classification.

Data generated dataset is the data text that we generated for basic testing our algorithm. The dataset divided into 3 Category: Category1, Category2 and Category3. Venn diagram of feature term is shown in figure 6. We generated 3 dataset, each dataset contained 1000 documents. For each dataset, the training data consists of 800 documents and testing data consists of 200 documents. Some of data generated documents are shown in table 1.

The table 2 showed the accuracy rate of classification by ARC-BC and ARTC algorithm. The accuracy rate from ARTC algorithm is more than that from ARC-BC algorithm.

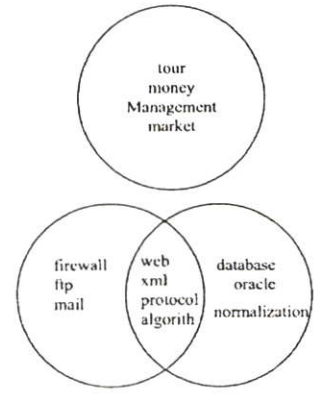


Figure 6. Venn diagram of features term

Table 1. Some data generated documents and category.

No.	Feature of document	Category
1	firewall, ftp, web, xml	2
2	database, oracle, web, protocol	3
3	web, xml, protocol, mail, ftp	2
4	normalization, oracle, database ,xml	3
5	database, oracle, normalization	3
6	tour, management, market, hotel	1
7	management, hotel, tour, money	1
8	hotel, tour, money	1
9	oracle, web, database	3
10	ftp, firewall, mail	2

Table 2. Data generated experimental result

Dataset	Accuracy rate (%)	
	ARC-BC	ARTC
1	48.5	100
2	42	100
3	44	100

CSTR data set is the abstracts of URCS technical reports published in the department of computer science at the University of Rochester between 1987 and 2005. It has been use in [1, 3, 4, 5] for text categorization and clustering. The dataset contained 472 abstracts, which were divided into four research areas: AI, Robotics and Vision, Systems and Theory. The set of keywords of a documents is used on the set of term, $T = \{t_1, t_2, \ldots, t_n\}$, of document

CSTR document is shown in figure 7. It's an abstract from 472 documents. Each document consists of title keywords and abstract.

Shaw, J., "Predictive Coding with Temporal Invariance", TR859, Computer Science Dept., U. Rochester, March 2005.
05.tr859.Predictive_coding_with_temporal_invariance.pdf

Keywords: temporal invariance; predictive coding; unsupervised learning.

Predictive coding and temporal invariance are two major unsupervised learning principles which have been used to explain the behavior of parts of the brain (most notably the striate cortex). Although both have been around for a number of years, no formal relationship between them has been established. We prove that temporal invariance is a form of predictive coding. To do this, we begin with the goal of predictive coding, make a set of assumptions about the class of problem we are dealing with, and derive temporal invariance from the predictive coding goal and our added assumptions.

Figure 7. CSTR document

CSTR experimental results are shown in table 3.

Table 3.CSTR Experimental results

Category	Accuracy rate (%)	
	ARC-BC	ARTC
AI	65.22	95.65
Systems	87.68	93.94
Robotics and Vision	70.37	92.59
Theory	89.74	97.43
All	80.33	95.08

The table 3 shows the accuracy rate of classification by categories that are given by ARC-BC algorithm and ARTC algorithm. In addition, the last line from table 3 also shows the total accuracy rate of the CSTR dataset. The accuracy rate from ARTC algorithm is more than that from ARC-BC algorithm.

5. Conclusion and future work

In this paper, we introduce the new algorithm for text categorization, Association Rule-Based Text Classifier algorithm (ARTC). According to the experimental results, ARTC algorithm shows good performance in classifying the data. In the future, we will further conduct experimentations on k-datasets and Reuters-top10 collection datasets to evaluate the performance of the algorithm.

6. References

[1] B. Tang. M. Shepheral , E. Milios, & M. Heywood. (2005). Comparing and Combining Division Reduction Techniques for Efficient Text Clustering. *International Workshop on Feature Selection for Data Mining: Interfacing Machine Learning and Statistics*.

[2] F. Sebastiani. (2002). ACM Computer Surveys. *Machine learning in automated text categorization*. 3, 1-47.

[3] J. Han, & M. Kamber. (2001). *Data Mining: Concepts and Techniques*. San Francisco: Morgan Kaufmann.

[4] O. Zaiane, & M.-L. Antonie. (2002). Text Document Categorization by Term Association. *Proceeding of The 2002 IEEE International Conference on Data Mining* (pp.19-26). Japan: Maebashi.

[5] O. Zaiane, & M.-L. Antonie. (2002). Classifying Text Documents by Association Terms with Text Categories. *In Proc. of the Thirteenth Australasian Database Conference* (pp.215-222). Australia: Melbourne.

[6] T. Li, S. Zhu, & M. Ogihara. (2003). Efficient Multi-way Text Categorization via Generalized Discriminate Analysis. *Proceeding of the Twelfth International Conference on Information and Knowledge Management* (pp.317-324)

[7] T. Li, S. Zhu, & M. Ogihara. (2004). Document clustering via adaptive subspace iteration. *Proceedings of the 27th annual international conference on Research and development in information retrieval 2004* (pp.218-225).

[8] T. Li, S. Zhu, &M. Ogihara. (2004). Entropy-Based criterion in Categorical Clustering. *Proceedings of the twenty-first international conference on Machine learning 2004* (pp.536-543).

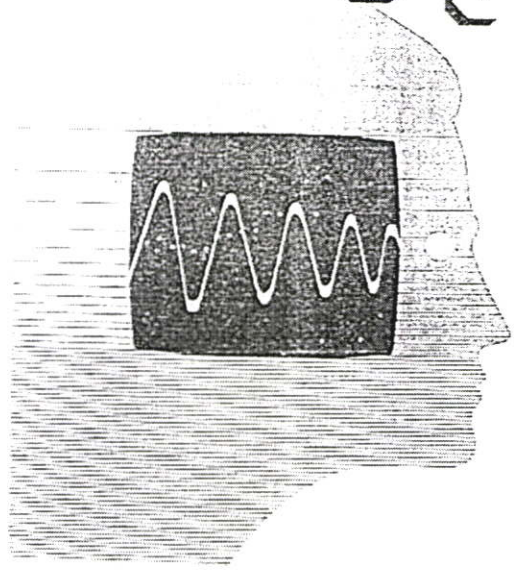
[9] T.M. Mitchell. (1997). *Machine learning*.. New York: McGraw Hill.

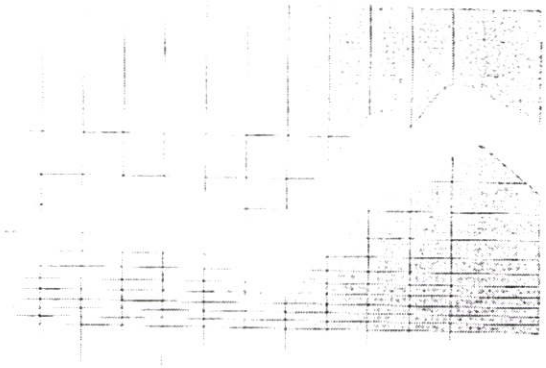
Proceedings

Seventeenth International Conference on

TOOLS WITH ARTIFICIAL INTELLIGENCE

14–16 November 2005
Hong Kong, China





Sponsored by
IEEE Computer Society
Co-Sponsored by
Information Technology Research Institute,
Wright State University
The Hong Kong University of Science and Technology

Edited by
Andrew Lim

IEEE COMPUTER SOCIETY

IEEE

A New Association Rule-Based Text Classifier Algorithm

Supaporn Buddeewong[1] and Worapoj Kreesuradej[2]
Faculty of Information Technology
King Mongkut's Institute of Techology Ladkrabang,
Bangkok, 10520 Thailand
E-mail:s5066003@kmitl.ac th[1], worapoj@it.kmitl.ac.th[2]

Abstract

This paper proposes a new association rule-based text classifier algorithm to improve the prediction accuracy of Association Rule-based Classifier By Categories (ARC-BC) algorithm. Unlike the previous algorithms, the proposed association rule generation algorithm constructs two types of frequent itemsets. The first frequent itemsets, i.e. L_k contain all term that have no an overlap with other categories. The second frequent itemsets, i.e. OL_k contain all features that have an overlap with other categories. In addition, this paper also proposes a new join operation for the second frequent itemsets. The experimental results are shown a good performance of the proposed classifier

1. Introduction

Text categorization task is defined as assigning category labels to new documents based on their contents. Text categorization research has a long history, starting in early 1960s. There are several text categorization approaches have been proposed such as neural networks, genetic algorithms and probabilistic models, support vector machine.

Recently, Association Rule-based Classifier By Categories (ARC-BC) [3, 4] algorithm is proposed to build an associative text classifier. ARC-BC algorithm considers each set of documents belonging to one category as a separate text collection to generate association rules. If a document belongs to more than one category, this document will be present in each set associated with the categories that the document falls into. However, the algorithms may fail to classify a single-class document that has some terms of document mutually associated with other class. Therefore, in this paper, we propose a new association rule-based text classifier to deal with such problem.

This paper is organized as following. The second, we introduce our new text categorization approach. Experimental results are described in Section 3. We summarize our research and future work in the forth Section.

2. A New Association Rule-Based Text Classifier Algorithm

Here, a new algorithm for association rule generation and a new categorization algorithm based on the new set of rules is proposed.

2.1. Association Rule Generation

Like typical apriori-based algorithms, our associative rules are generated from frequent itemsets. Therefore, frequent itemsets are constructed in order to get the set of associative rules. However, unlike the previous algorithms, our method constructs two types of frequent itemsets. The first frequent itemsets, i.e. L_k, contain all term that have no an overlap with other categories. The second frequent itemsets, i.e. OL_k, contain all features that have an overlap with other categories.

Basically, any frequent itemsets has to include a category label starting from 2-itmesets to k-itemsets. For the first frequent itemsets, L_k, is constructed using apriori join. As an example, L_k is constructed by using apriori join between L_{k-1} and L_{k-1}. In this paper, a new join operation for the second frequent itemsets, OL_k, is proposed.

For the new join operation, any two items in an itemset can be joined if they have the same category. The examples of both of apriori join and the new join (ARTC join) are shown in figure 1 and figure2. In addition, the algorithm, called ARTC, for generating association rules are shown in figure 3.

1082-3409/05 $20.00 © 2005 IEEE

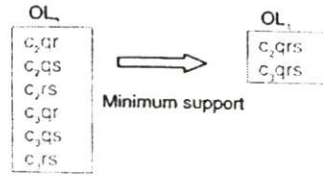


Figure 1. Apriori join

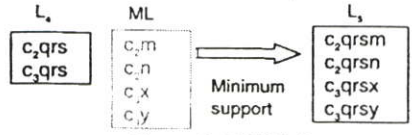


Figure 2. ARTC join

(1) C_0 ← {Category and their support}
(2) L_0 ← {$c \in C_0$ | support > min_support}
(3) C_1 ← {Feature 1-itemset and their support}
(4) L_1 ← {$c \in C_1$ | support > min_support}
(5) C_2 ← Gen_2_Itemsets (L_0,L_1)
(6) T ← {$c \in C_2$ | support > min_support}
(7) OL_2 ← Find_Overlap_Itemset (T)
(8) L_2 ← T - OL_2
(9) DB_2 ← Cut_Useless_DB_Record (DB_1, OL_2)
(10) ML ← Select_Itemset(OL_2,L_2)
(11) For (k=3; $OL_{k-1} \neq \phi$; k++)
(12) {C_k ← Gen_k_Itemsets (OL_{k-1}, ML)
(13) L_k ← {$c \in C_k$ | support > min_support}
(14) OL_k ← Gen_Overlap_Itemsets (OL_{k-1})}
(15) SetOfRule ← { $\in L_0, L_1, \ldots, L_k, OL_2, OL_3, \ldots, OL_k$ }
(16) Rules = ϕ
(17) For each item in SetOfRule do
(18) {Rules ← $t_1 \wedge t_2 \wedge t_3 \ldots \wedge t_{k-1} \Rightarrow C$;t is feature of document and C is label of class}

Figure 3. ARTC algorithm

In our algorithm, T set is generated from L_0 and L_1 using apriori join. L_0 is a set of categories that satisfies by support threshold and L_1 is a set of terms that satisfy by support threshold. Then, L_2 is generated form T set. Next, frequent itemsets starting from L_3 to L_k are generated by OL_{k-1} join ML. ML includes all itemsets that have the same category in OL_k .Figure 4 shows an example of T, L_2, OL_2 and ML.

3. Experimental Results

For our experiment, we use Computer Science Technical Report collection dataset (CSTR) to evaluate our algorithm. CSTR data is the abstracts of URCS technical reports published in the department of computer science at the University of Rochester between 1987 and 2005. The dataset contained 472 abstracts, which were divided into four research areas: AI, Robotics and Vision, Systems and Theory.

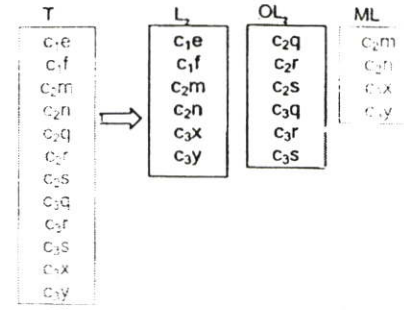


Figure 4. Sample of T, L_2, OL_2 and ML

Here, we use classification accuracy rate for evaluation. Training set and testing set of each category and experimental results are shown in table 1.

Table 1.Experimental results

Category	Accuracy rate	
	ARC-BC	ARTC
AI	65.22 %	95.65 %
Systems	87.88 %	93.94 %
Robotics and Vision	70.37 %	92.59 %
Theory	89.74 %	97.43 %
All	80.33 %	95.08 %

4. Conclusion and Future work

In this paper, we introduce the new algorithm for text categorization, Association Rule-Based Text Classifier algorithm (ARTC). According to the experimental results, ARTC algorithm shows good performance in classifying the data. In the future, we will further conduct experimentations on the other datasets to evaluate the performance of the algorithm.

5. References

[1] F. Sebastiani, Machine learining in automated text categorization, ACM Computing Surveys, 2002.
[2] J. Han, and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco, 2001.
[3] O. Zaiane, and M. Antonie, "Text Document Categorization by Term Associaton", Proceedings of ICDM 2002, IEEE, 2002, pp.19-26.
[4] O. Zaiane, and M. Antonie, "Classifying Text Documents by Association Terms with Text Categories", Proceedings of thirteenth Australasian conference on Database technologies, Australian Compter Society Inc., 2002, pp-215-222.
[5] T.M. Mitchell, Machine learning. McGraw Hill, New York, 1997.

ประวัติผู้เขียน

ชื่อ	นางสาวสุภาภรณ์ บุตรดีวงษ์
วัน เดือน ปีเกิด	25 มีนาคม 2519
ที่อยู่	เลขที่ 15 หมู่ 1 ต.โคกปีบ อ.ศรีมโหสถ จ.ปราจีนบุรี 25190
ประวัติการศึกษา	2542 วิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยหอการค้าไทย
ประวัติการทำงาน	2452-ปัจจุบัน รับราชการ ตำแหน่งนักวิชาการคอมพิวเตอร์ สังกัดสำนักหอสมุด มหาวิทยาลัยเกษตรศาสตร์