อัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์

A NEW ALGORITHM FOR WEB SEQUENTIAL PATTERN DISCOVERY

บัญญัติ ทิพย์หมื่น
BUNYAT THIPMOUD

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2549

ISBN 974-15-2082-2

อัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์

A NEW ALGORITHM FOR WEB SEQUENTIAL PATTERN DISCOVERY

บัญญัติ ทิพย์หมัด

BUNYAT THIPMOUD

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2549

ISBN 974-15-2682-2

A NEW ALGORITHM FOR WEB SEQUENTIAL PATTERN DISCOVERY

BUNYAT THIPMOUD

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 974-15-2682-2

หัวข้อวิทยานิพนธ์	อัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์
นักศึกษา	นายบัญญัติ ทิพย์หมัด
รหัสประจำตัว	44067082
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
พ.ศ.	2549
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ ดร.วรพจน์ กรีสุระเดช

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนออัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์ โดยใช้วิธีการ Join Operation เพื่อสร้าง Candidate Generation (C_k) มีวัตถุประสงค์เพื่อปรับปรุงการค้นหากฎความสัมพันธ์เพื่อใช้ในการทำนาย กับข้อมูลที่เป็นลักษณะเว็บเพจ ดังจะเห็นว่าหากนำวิธีการ Sequential Pattern Discovery Algorithm กับข้อมูลที่เป็นลักษณะเว็บเพจนั้นจะให้ผลลัพธ์หรือกฎความสัมพันธ์ที่ได้มีประสิทธิภาพต่ำ เพราะวิธีการดังกล่าวข้างต้นนั้นจะไม่คำนึงถึงปัจจัยในเรื่องข้อมูลที่อยู่ชิดติดกันและข้อมูลเพจสุดท้าย

ดังนั้นวิทยานิพนธ์ฉบับนี้จึงได้นำเสนอวิธีการ ค้นหากฎความสัมพันธ์ ที่สามารถค้นพบกฎที่มีความสัมพันธ์ทั้งแบบสั้นและยาว โดยเริ่มจากการค้นหากฎที่มีขนาดสั้นที่สุดก่อนแล้วนำกฎดังกล่าวนั้นไปสร้างความสัมพันธ์เพื่อที่จะค้นหากฎที่มีขนาดยาวมากยิ่งขึ้น อีกทั้งยังเป็นการแก้ปัญหาอันเกิดขึ้นจากวิธีการ Sequential Pattern Discovery Algorithm โดยความสัมพันธ์ที่จะเกิดขึ้นนั้นจะมีเงื่อนไขว่าข้อมูลจะต้องเป็นลำดับ อยู่ชิดติดกันและข้อมูลเพจสุดท้าย ซึ่งการสร้างความสัมพันธ์ดังกล่าวนี้เรียกว่า Join Operation วิธีการดังกล่าวนี้จะมีความเหมาะสมกับข้อมูลที่เป็นลักษณะเว็บล็อกไฟล์อย่างยิ่ง ซึ่งผลลัพธ์ที่ได้จะทำให้สามารถค้นพบทั้งกฎที่มีรูปแบบสั้นและยาวนั้นเอง ทำให้ความแม่นยำในการทำนายสูงขึ้นด้วย

เมื่อได้ความสัมพันธ์ที่จะนำไปสร้างเป็นกฎความสัมพันธ์แล้ว นำกฎความสัมพันธ์ที่ได้มาคัดทิ้งโดยใช้โครงสร้างทางต้นไม้ (Tree Pruning) ในการคัดทิ้งกฎความสัมพันธ์ที่ไม่จำเป็นทิ้งและหลังจากนั้นนำกฎความสัมพันธ์ที่เหลือไปใช้ในการทำนายเว็บเพจ ซึ่งผลการทดลองที่ได้จะให้อัตราความแม่นยำในการทำนาย (Precession Rate) ที่สูงขึ้นนั้นเอง

Thesis Title	A new algorithm for web sequential pattern discovery
Student	Mr.Bunyat Thipmoud
Student ID.	44067082
Degree	Master of Science
Program	Information Technology
Year	2006
Thesis Advisor	Assoc. Professor Dr.Worapoj Kreesuradej

ABSTRACT

This thesis offers a new algorithm for web sequential pattern discovery by using Join Operation Method. This method creates Candidate Generation (C_k) on the objective of improving rule of relation used in predict the World Wide Web. It is obviously seen that Sequential Pattern Discovery Algorithm method, applied with data on web page, would result in low efficiency rule of relation because this method does not concern about order and recent page factor.

Therefore, this thesis would offer method of sequential pattern discovery that could find rule with short and long rule. The short rule will be the starting point and be used to discovery relationship to search for longer sequence. This would also solve the problem of Sequential Pattern Discovery Algorithm Method by the condition that the information must be in order and sequence to each other in order to establish the relationship. This relationship establishment could be called as Join Operation. This method suits for web-log data, and resulted in the ability to discovery short and long sequence with more precise forecast.

When relationship used in rule of relation incurred, rule of relation would be selected by Tree Pruning and eliminate unimportant relation. The rest rule would be used in prediction and resulted in increasing Precession Rate of prediction.

กิตติกรรมประกาศ

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

ขอขอบคุณ ผศ.ดร หมัดอามีน หมันหลิน อาจารย์ประจำภาควิชาเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีนานาชาติ สิรินธร มหาวิทยาลัย ธรรมศาสตร์ (รังสิต) ที่ได้ให้คำแนะนำเกี่ยวกับการทำงานวิจัย

ขอขอบคุณ บริษัท เอ็มเน็ต โซลูชั่น จำกัด ที่ให้เวลาในการทำวิจัยอย่างเต็มที่จนสำเร็จลุล่วง

ขอขอบคุณ DME LAB ที่สนับสนุนอุปกรณ์การทำวิจัยจนสำเร็จ

ขอขอบคุณ สุภาภรณ์ บุตรดีวงษ์ นักศึกษาคณะเทคโนโลยีสารสนเทศ ที่ช่วยเหลือให้คำแนะนำในเรื่องการเขียนโปรแกรมสำหรับเตรียมข้อมูลในการทดลองของงานวิจัยนี้

สุดท้ายนี้ขอขอบคุณบิดา มารดา และพี่น้องร่วมอุทรที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่อง จนทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่า และประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่บิดา มารดา อันเป็นที่รักยิ่ง ซึ่งเป็นผู้ให้กำเนิดและทำให้ข้าพเจ้าได้มีวันนี้

บัญญัติ ทิพย์หมัด

สารบัญ

สารบัญ(ต่อ)

สารบัญตาราง

สารบัญตาราง(ต่อ)

สารบัญรูป

สารบัญรูป(ต่อ)

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เว็บล็อกทรานเซคชั่นนั้นประกอบไปด้วย ข้อมูลที่บันทึกการเรียกใช้งานของผู้เรียกใช้เว็บเพจ งานวิจัยนี้จะนำเสนออัลกอริธึมซึ่งมีหน้าที่หลักในการค้นหากฎเพื่อใช้ในการทำนาย ข้อมูลที่เป็นลักษณะเว็บเพจนั้นแตกต่างจากข้อมูลทั่วไปตรงที่ลำดับการเรียกใช้เว็บเพจ (User's next requests) นั้นจะมีความสำคัญมาก ดังจะเห็นว่าเมื่อนำวิธีการ Sequential Pattern กับข้อมูลที่เป็นเว็บเพจนั้นจะมีการกระโดดของข้อมูลเกิดขึ้น เมื่อนำผลที่ได้นั้นเมื่อนำไปใช้งานในการทำนายจะทำให้ผลลัพธ์ที่มีประสิทธิภาพต่ำ

จากการศึกษาทำให้ทราบถึงปัญหาเกี่ยวกับการค้นพบกฎและการลดกฎความสัมพันธ์ดังกล่าว ขนาดของการสับกฎ (Slide window) จะถูกกำหนดไว้ตายตัว ซึ่งจะส่งผลโดยตรงต่อการค้นหากฎ ดังเช่น เมื่อมีการกำหนดการสับกฎที่มีขนาดเล็กเกินไป จะทำให้กฎที่มีขนาดยาวไม่สามารถค้นพบได้ ในทางกลับกันหากกำหนดการสับกฎมากเกินไป ก็จะเกิดการสูญเสียกฎที่มีขนาดเล็กเพราะไม่สามารถค้นพบได้เช่นกัน โดยการแก้ปัญหาจะใช้วิธีการแบบ Join Operation นั้นหมายถึงว่าจะมีการสับกฎได้ทั้งกฎที่มีขนาดเล็กและใหญ่ ซึ่งผลลัพธ์ที่ได้จะทำให้สามารถค้นพบทั้งกฎที่มีรูปแบบสั้นและยาวนั้นเอง ทำให้ความแม่นยำในการทำนายสูงขึ้นด้วย

หลักการในการพิจารณากฎจะให้ Path-based Model โดยการสร้างกฎที่มีความยาวของเว็บเพจ มากขึ้นจะใช้วิธีการ Join Operation เพื่อสร้าง A New Candidate Generations (Ck) โดยวิธีการนี้จะทำให้เวลาในการประมวลผลการทำงานของอัลกอริธึมลดน้อยลงไปด้วย เพราะเหตุว่ารูปแบบการเรียกใช้ บางรูปแบบที่มีความน่าจะเป็นน้อยมากก็จะถูกลดทอนออกไปนั้นเอง

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์ฉบับนี้มีวัตถุประสงค์เพื่อนำเสนอวิธีใหม่ในการค้นหากฎความสัมพันธ์ จากแหล่งข้อมูลที่สำคัญในงานวิจัยนี้คือ เว็บล็อกไฟล์ ที่ได้จากจากเว็บเซอร์เวอร์ ซึ่งจะบันทึกพฤติกรรมการเรียกใช้งานของผู้เรียกใช้เว็บเพจ โดยข้อมูลที่เก็บไว้ดังกล่าวนั้นสามารถนำมาใช้ในการทำนายความต้องการเรียกใช้เว็บเพจได้

1.3 สมมติฐานของการศึกษา

ในส่วนของการค้นหาข้อมูลแบบลำดับ (Mining Sequential Patterns) นั้นเมื่อนำมาประยุกต์กับข้อมูลที่เป็นเว็บเพจในการทำนายความต้องการของยูสเซอร์ที่จะเรียกใช้งานเว็บเพจในลำดับต่อไป (user's next requests) จะให้ผลที่คลาดเคลื่อนเพราะเหตุว่า การใช้ Sequential pattern จะเกิดการกระโดดของไอเท็ม ซึ่งเมื่อนำมาใช้กับเว็บเพจในการทำนายเว็บเพจในลำดับต่อไปจะให้ผลลัพธ์ที่มีประสิทธิภาพต่ำ

การแก้ปัญหาข้างต้นนี้ การค้นพบกฎและการลดกฎความสัมพันธ์ดังกล่าว ขนาดของการสับกฎ (slide window) จะถูกกำหนดไว้ตายตัว ซึ่งจะส่งผลโดยตรงต่อการค้นหากฎ ดังตัวอย่างเช่น เมื่อมีการกำหนดการสับกฎที่มีขนาดเล็กเกินไป จะทำให้กฎที่มีขนาดยาวไม่สามารถค้นพบได้ โดยการแก้ปัญหาจะให้วิธีการแบบ Join Operation นั้นหมายถึงว่าจะมีการค้นหากฎได้ทั้งกฎที่มีขนาดเล็กและใหญ่ ซึ่งผลลัพธ์ที่ได้จะทำให้สามารถค้นพบทั้งกฎที่มีรูปแบบสั้นและยาวนั้นเอง

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

การค้นหากฎความสัมพันธ์ เป็นการค้นหาความสัมพันธ์ของข้อมูลจากเว็บล็อกทรานเซ็กชั่น โดยความสัมพันธ์ที่ได้สามารถนำมาทำนายแนวโน้มการเรียกใช้ของเว็บเพจ ในการคาดการว่าเว็บเพจต่อไป (Next Request) ที่ผู้เรียกใช้เว็บเพจจะเรียกใช้งานคือเว็บเพจอะไร ข้อดีของการหากฎความสัมพันธ์มีดังนี้

1. ทำงานได้ดีกับข้อมูลที่เป็นลักษณะเว็บล็อกไฟล์ และมีรูปแบบที่การเรียกใช้หรือความถี่ในการเกิดของข้อมูลซ้ำๆ กัน
2. สามารถควบคุมการจัดเก็บกฎเพื่อใช้ในการทำนายด้วยค่าสนับสนุนน้อยที่สุด (Minimum support) และค่าความเชื่อมั่นน้อยที่สุด (Minimum confidence) ได้ จะส่งผลโดยตรงต่อการเก็บกฎเพื่อใช้ในการทำนาย

การใช้วิธีการของ Join Operation จะทำให้สามารถค้นหากฎได้ทั้งแบบสั้นและยาว ส่งผลให้ความแม่นยำในการทำนายสูงขึ้นด้วย อีกทั้งยังมีการนำวิธีการคัดเลือกกฎ ที่อยู่ใกล้ชิดติดกัน และเก็บบันทึกเว็บเพจที่อยู่หลักสุดไว้ด้วย เพื่อประโยชน์ในการลดทอนปริมาณของกฎ ที่มีประโยชน์น้อยออกไปด้วยนั้นเอง

1.6 ขอบเขตการวิจัย

ในวิทยานิพนธ์ฉบับนี้เสนออัลกอริทึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์ ซึ่งข้อมูลที่ใช้ในการวิจัยจะเป็นข้อมูลจากเซอร์เวอร์ซึ่งเป็นลักษณะล็อกไฟล์ โดยข้อมูลดังกล่าว

นั้นมีคุณสมบัติบอกว่าผู้เรียกใช้เว็บเพจมีการเรียกใช้เว็บเพจอะไรบ้าง เรียกใช้เวลาใด โดยกระบวนการของอัลกอริทึมใหม่นี้ จะมีกระบวนการและขั้นตอนหลักๆ ดังต่อไปนี้

กระบวนการเตรียมข้อมูล

ค้นหารูปแบบลำดับของเว็บไซต์

การลดทอนกฎ

ภายใต้การจำลองระบบด้วยโปรแกรม MATLAB โดยใช้ข้อมูลในการทดลองคือ ข้อมูลจากการจำลอง 1 ชุดข้อมูล และข้อมูลล็อกไฟล์เซอร์เวอร์จาก www.epa.gov (the Environmental Protection Agency Web site: U.S. Environmental Protection Agency) โดยใช้ค่าความถูกต้อง (Precision rate) ในการวัดประสิทธิภาพ

1.7 ขั้นตอนของการศึกษา

ในขั้นตอนของการศึกษานี้ ได้แสดงลำดับการทำงานตั้งแต่เริ่มต้นจนถึงสิ้นสุดการทำงานวิจัย ดังรายละเอียดต่อไปนี้

1. ศึกษาทฤษฎีและงานวิจัยจากเอกสาร บทความต่าง ๆ ที่เกี่ยวข้องกับการทำงานวิจัย
2. กำหนดหัวข้อ เป้าหมาย วัตถุประสงค์ และขอบเขตการทำงานวิจัย
3. วิเคราะห์และออกแบบอัลกอริทึมใหม่
4. ค้นหาชุดข้อมูลที่จะนำมาใช้ในการทดลอง
5. เตรียมข้อมูลที่จะนำมาใช้ทดลอง
6. พัฒนาโปรแกรมพร้อมทั้งแก้ไขข้อผิดพลาด
7. ทดสอบอัลกอริทึมกับชุดข้อมูล
8. รวบรวมผลการทดลอง จากผลการทำงานของโปรแกรม
9. วิเคราะห์และสรุปผลการดำเนินงาน
10. จัดทำเอกสารประกอบงานวิจัย

บทที่ 2
ทฤษฎีพื้นฐาน และงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ และงานวิจัยที่เกี่ยวข้องในการทำวิจัย โดยเนื้อหาในบทนี้จะกล่าวถึง การค้นหากฎความสัมพันธ์ (Association rule discovery) และงานวิจัยที่เกี่ยวข้องซึ่งก็คืออัลกอริทึมการค้นหากฎแบบลำดับ Sequential pattern discovery

2.1 การค้นหากฎความสัมพันธ์ (Association rule)

การค้นหากฎความสัมพันธ์ 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 A.HTML แล้ว B.HTML” (If A.HTML Then B.HTML) ใช้สัญลักษณ์แทน “A.HTML => B.HTML” หมายถึง ถ้าเกิด A.HTML แล้วจะเกิด B.HTML ด้วย หากนำมาใช้สำหรับการค้นหารูปแบบการเรียกใช้เว็บเพจ จะหมายความว่า ถ้ามีการเรียกใช้งาน เว็บเพจ A.HTML ในลำดับต่อไปจะเป็นการเรียกใช้งานเว็บเพจ B.HTMLตามลำดับ

นิยามและความหมายของคำต่าง ๆ ที่ใช้ในการค้นหากฎความสัมพันธ์ได้แก่

1. ค่าสนับสนุน (Support value: Sup) เป็นค่าแสดงความสัมพันธ์ระหว่างจำนวนของเหตุการณ์ IF => THEN (LHS => RHS) ที่เกิดขึ้น กับจำนวนรายการที่เกิดขึ้นทั้งหมด สามารถแสดงเป็นสมการได้ดังสมการที่ 2.1

$$\text{ค่าสนับสนุน(Sup)} = \frac{\text{Count (LHS => RHS)}}{\text{Count (Table)}} \qquad (2.1)$$

2. ค่าสนับสนุนน้อยที่สุด (Minimum support) คือค่าสนับสนุนที่น้อยที่สุดที่ทำให้ความสัมพันธ์ที่ได้นั้นยังมีความน่าสนใจ
3. ค่าความเชื่อมั่น (Confidence value: Conf) เป็นค่าแสดงความเป็นจริงของกฎ สามารถคำนวณโดยการหาค่า LHS => RHS ด้วยจำนวนเหตุการณ์ LHS ที่เกิดขึ้น แสดงดังสมการที่ 2.2

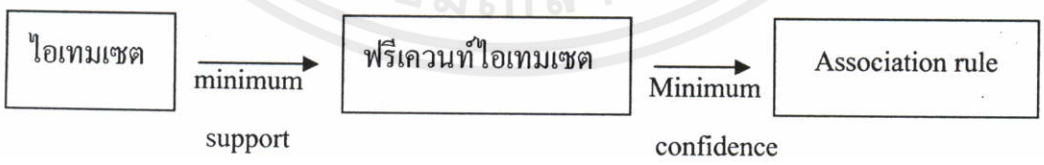
$$\text{ค่าความเชื่อมั่น (Conf)} = \frac{\text{Sup (LHS => RHS)}}{\text{Sup (LHS)}} \quad (2.2)$$

เมื่อ

$$\text{Sup (LHS)} = \frac{\text{Count (LHS)}}{\text{Count (Table)}}$$

4. ค่าความเชื่อมั่นน้อยที่สุด (Minimum confidence) คือค่าความเชื่อมั่นที่น้อยที่สุดที่ทำให้กฎความสัมพันธ์ที่ได้นั้นยังมีความน่าสนใจ ซึ่งในงานวิจัยดังกล่าวนี้ หากรูปแบบการเรียกใช้งาน (Pattern) ใดมีความน่าเชื่อมั่นต่ำ เราจะมีวิธีการตัดทิ้งรูปแบบการเรียกใช้งานดังกล่าวนั้นทิ้งไป
5. ไอเทม (Item) คือข้อมูลแต่ละตัวที่ใช้ในการหากฎความสัมพันธ์ เช่น car.html, Toyota.html, loan.html, bank.html เป็นต้น
6. ไอเทมเซต (Itemset) คือ ความสัมพันธ์ของข้อมูลที่หาได้ Itemset ประกอบด้วย Item โดย k-itemsets ประกอบด้วย k-items ใน itemset นั้น ๆ เช่น 2-itemsets ยกตัวอย่าง เช่น {car.html, Toyota.html} , {loan.html, bank.html} เป็นต้น และถ้า 3-itemsets ยกตัวอย่างเช่น {index.html, halal.html, products.html} , {Thailand.html, Bangkok.html, hotel.html} เป็นต้น
7. ฟรีเควนท์ไอเทมเซต (Frequent Itemset) หรือ ลาร์จไอเทมเซต (Large Itemset) คือ ชุดของ Itemset ที่มีค่าสนับสนุนมากกว่าหรือเท่ากับค่าสนับสนุนน้อยที่สุด

การค้นหากฎความสัมพันธ์มีกระบวนการทำงาน 2 ขั้นตอนคือ หา Frequent Itemset ทั้งหมด และสร้างกฎความสัมพันธ์จาก Frequent Itemset แสดงดังรูปที่ 2.1



รูปที่ 2.1 แสดงขั้นตอนการค้นหากฎความสัมพันธ์

จากรูปที่ 2.1 แสดงขั้นตอนการค้นหากฎความสัมพันธ์ เริ่มต้นจาก Itemset ที่ผ่านค่าสนับสนุนน้อยที่สุดแล้วจะเป็น Frequent Itemset และจาก Frequent Itemset ที่ได้นี้เมื่อผ่านค่าความเชื่อมั่นน้อยที่สุดแล้ว สามารถนำ Frequent Itemset นี้มาสร้างเป็นกฎความสัมพันธ์ได้ สำหรับรายละเอียดในการหา Frequent Itemset และกฎความสัมพันธ์จะกล่าวในหัวข้อต่อไป

2.1.1 อัลกอริทึมอพริออริ (Apriori Algorithm)

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

2.1.1.1 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 เพื่อเป็นการเพิ่มประสิทธิภาพของอัลกอริทึมโดยการช่วยลดพื้นที่ ที่จะต้องค้นหา Frequent Itemset ในฐานข้อมูล กระทำโดย Itemset ใด ๆ ที่มีค่าสนับสนุนน้อยกว่าค่าสนับสนุนน้อยที่สุด (Minimum Support) ที่ตั้งไว้ ซึ่งจะกล่าวได้ว่า Itemset ดังกล่าวนั้น 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 แทนด้วยสัญลักษณ์ 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 หากเหมือนกันก็สามารถเชื่อมความสัมพันธ์กันได้ 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 การคัดสมาชิกออกด้วยคุณสมบัติของเอพริออริคือ 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 หลังจากการคัดสมาชิกออกด้วยคุณสมบัติของอพริออริแล้ว ต้องคัดสมาชิกออกอีกครั้งด้วยค่าสนับสนุนน้อยที่สุดโดยคัดสมาชิกใน C_k ที่มีความถี่น้อยกว่าค่าสนับสนุนน้อยที่สุดออก เพื่อสร้างเป็น L_k

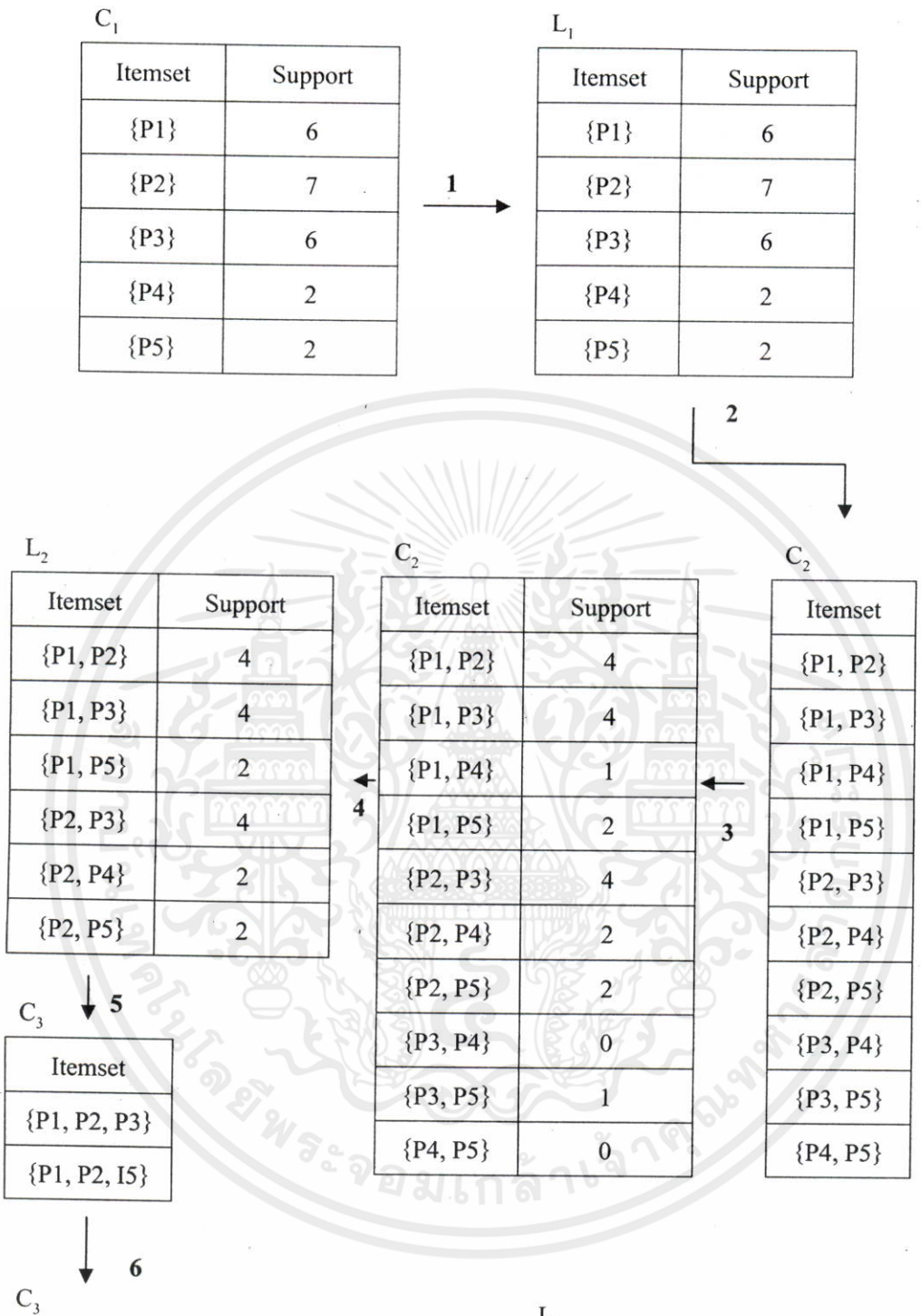
การหา Frequent Itemset จะกระทำวนซ้ำตามขั้นตอนที่ 1 และ 2 ไปเรื่อย ๆ จนกว่าจะไม่สามารถหา L_k ได้อีก จึงหยุดการหา Frequent Itemset วิธีการโดยละเอียดของการหา Frequent Itemset สามารถแสดงได้ดังตัวอย่างต่อไปนี้ [3]

กำหนดให้ฐานข้อมูล D (Database) มีจำนวนทรานแซกชัน (Transaction) 9 ทรานแซกชัน แสดงดังรูปที่ 2.2 และค่าสนับสนุนน้อยที่สุดมีค่าเท่ากับ 2 และค่าความเชื่อมั่นน้อยที่สุดมีค่า 70%

ตารางที่ 2.1 ฐานข้อมูล Database

ลำดับ	ข้อมูล
1	P1, P2, P5
2	P2, P4
3	P2, P3
4	P1, P2, P4
5	P1, P3
6	P2, P3
7	P1, P3
8	P1, P2, P3, P5
9	P1, P2, P3

จากตารางที่ 2.1 ซึ่งเป็นฐานข้อมูล Database นั้นจะมีการนำข้อมูลดังกล่าวนี้ไปใช้ในกระบวนการแรกหรือเริ่มต้นการทำงานของอัลกอริธึม โดยกระบวนการเริ่มต้นดังกล่าวนี้จะเห็นการค้นหาการเกิดขึ้นของ Itemset ที่มีอยู่สามารถนำรายการข้อมูลที่มีอยู่มาหากฎความสัมพันธ์ของข้อมูลแต่ละรายการ โดยการหาFrequent Itemset แสดงดังรูปที่ 2.2 จะเห็นได้ว่า แต่ละ Itemset นั้นจะมีอัตราการเกิดที่แตกต่างกัน ดังจะยกตัวอย่างเช่น {P1} จะมีความถี่ ในการเกิดขึ้น ด้วยกัน เมื่อทำการตรวจสอบการเกิดจากฐานข้อมูล จะเกิดหรือมีการเรียกใช้งาน ทั้งหมด 6 ครั้งด้วยกัน {P2} จะมีความถี่ในการเกิดขึ้น เมื่อทำการตรวจสอบการเกิดจากฐานข้อมูล จะมีการเรียกใช้ทั้งหมด 7 ครั้งด้วยกัน ในขณะที่การเกิดขึ้นของ {P3} เกิดขึ้น 6 ครั้ง ในขณะที่ {P4} และ {P5} มีการเกิดขึ้นที่เท่ากัน คือ 2 ครั้งนั้นเอง หากแสดงผลทั้งหมดจะได้ดังรูปที่ 2.2



รูปที่ 2.2 แสดงวิธีหาFrequent Itemset

จากรูปที่ 2.2 แสดงการหาFrequent Itemsetจากฐานข้อมูล D เริ่มต้นจากหา C_1 โดยการแสกนหา 1- Itemset จากฐานข้อมูล D และนับความถี่ของแต่ละ Itemset ในฐานข้อมูล โดยขั้นตอนการทำงานต่างๆ และการสร้างกฎความสัมพันธ์มีกล่าวไว้[3]

L_1 = {large 1-sequences};//Result of litemset phse

For (k=2; $L_{k-1} \neq \varnothing$; k++} do

Begin

C_k = New candidate generated from L_{k-1}

(see section Appriori candidate generation)

for each customer-sequence c in the database do

Increment the count of all candidates in C_k

That are contained in c

L_k = Candicates in C_k with minimum support.

End

Answer = Maximul Sequences in $U_k L_k$;

รูปที่ 2.3 Algorithm Apriori

Apriori Candidate Generation

หน้าที่ของ The apriori-generate จะใช้ตัวแปร L_{k-1} โดยขั้นตอนแรกคือการ Join L_{k-1} กับ L_{k-1}

Insert into C_k

Select $p.litemset_1$,...$p.litemset_{k-1}$, $q.litemset_{k-1}$

From L_{k-1} p, L_{k-1} q

Where $p.litemset_1 = q.litemset_1$,...

$p.litemset_{k-2} = q.litemset_{k-2}$;

2.2 Mining Sequential Patterns

เราจะกล่าวถึงปัญหาของการไมนิ่ง Sequential pattern[11] เมื่อใช้กับฐานข้อมูลที่เป็นลักษณะเว็บเพจ ก่อนอื่นจะกล่าวถึงตัวอย่างของฐานข้อมูลการเช่าภาพยนต์ของร้านวีดีโอแห่งหนึ่ง เมื่อลูกค้ามาเช่าภาพยนต์เรื่อง Henrry Potter และต่อด้วย Lord of The Ring และสุดท้ายคือเรื่อง The Last Samurai ซึ่งการเช่าภาพยนต์ดังกล่าวนั้นมิจำเป็นต้องต้องมีการเช่าตามลำดับต่อเนื่องกัน (แต่ไม่สลับการเช่า) ซึ่งลูกค้านั้นสามารถที่จะเช่าวีดีโออื่นๆ ระหว่างการเช่าวีดีโอทั้งสามเรื่องได้ ดังกล่าวนี้ก็ได้ หากเป็นกรณีดังกล่าวข้างต้นนี้ กล่าวได้ว่า Sequential Pattern นี้ก็เป็นที่ยอมรับได้ นั้นหมายความว่า ลำดับการเช่าวีดีโอ ของภาพยนต์ทั้งสามเรื่องยังคงเป็นลักษณะการเช่าแบบลำดับนั้นเอง ซึ่งจากลำดับดังกล่าวนี้หากเปรียบภาพยนต์กับการเรียกใช้งานเว็บเพจ แล้วนั้นจะเห็นว่าการใช้งานสำหรับเว็บเพจจะเห็นว่าลำดับในการเรียกใช้งานจะมีผลต่อการทำนาย นั้นคือ หากมีลำดับนั้นไม่ต่อเนื่องกัน เมื่อนำไปใช้ในการทำนายจะให้ผลที่ผิดพลาดได้นั้นเอง ซึ่งจะได้กล่าวในโอกาสต่อไป

เมื่อกำหนดให้ ฐานข้อมูลพฤติกรรมของลูกค้าโดยแต่ละกิจกรรมจะประกอบด้วย รหัสลูกค้า, เวลาการซื้อและรายชื่อสินค้าที่จัดซื้อ (ซึ่งในเวลาเดียวกันจะไม่มีลูกค้าซื้อสินค้ามากกว่าหนึ่งรายการ) โดยที่ Sequence คือลำดับของรายการ Items โดยที่เราจะแสดงให้ itemset i $(i_1,i_2,...i_m)$ เมื่อ i_j คือ item ใดๆ โดยที่ลำดับของ sequence s $\langle s_1,s_2,s_n\rangle$ โดยที่ s_j คือ itemset ใดๆ

Sequence $\langle a_1\ a_2\ a_n\rangle$ จะประกอบไปด้วย sequence อื่นๆ $\langle b_1\ b_2\ b_m\rangle$ โดยที่ $i_1 < i_2 <...<i_n$ ดังตัวอย่าง $a_1 \subseteq b_{i1}$, $a_2 \subseteq b_{i2}$ $a_n \subseteq b_{in}$ ดังตัวอย่าง Sequence $\langle$ (7) (3 8) (9) (4 5 6) (8)$\rangle$ ประกอบไปด้วย $\langle$(3) (4 5) (8)$\rangle$ ดังนั้น (3) $\subseteq$ (3 8) , (4 5) $\subseteq$ (4 5 6) และ (8) $\subseteq$ (8) อย่างไรก็ตาม $\langle$(3) (5) $\rangle$ ไม่ได้อยู่ใน $\langle$(3 5)$\rangle$ จะเห็นได้ว่าหาก Sequence ใดที่เป็น Maximal Sequence แล้วนั้น sequence ดังกล่าวจะไม่อยู่ sequence อื่นๆอีกต่อไป เราจะจัดเรียงกิจกรรม (Transaction) ตามเงื่อนไขของเวลาที่เกิด โดย T_1, T_2,.....T_n โดยที่กลุ่มของ item ใน T_i แสดงด้วย itemset (T_i) โดยที่ลำดับของลูกค้าที่ใช้บริการจะเป็นลักษณะดังนี้คือ $\langle$itemset(T_i), itemset(T_2),...itemset (T_n)$\rangle$

ปัญหาใน Mining sequential patterns คือการค้นหา Maximal sequence ซึ่งจะต้องมีค่า minimum support โดยแต่ละ Maximal sequence ใดๆ นั้นแสดง sequential pattern. โดยที่ Sequence ใดๆ ที่มีค่า minimum support ก็จะเป็น large sequence นั้นเอง

ดังตัวอย่างพิจารณาฐานข้อมูลดังรูป 2.4 โดยที่ฐานข้อมูลดังกล่าวได้มีการจัดเรียงตามรหัส (customer ID)และเวลาการเกิดกิจกรรม (transaction-time) รูปที่ 2.5 แสดงฐานข้อมูลในรูปของ customer sequence. เมื่อกำหนดให้ minimum support มีค่าเท่ากับ 25% ซึ่งจะได้ Maximal sequence ดังนี้คือ $\langle$(30) (90)$\rangle$ และ $\langle$(30) (40 70)$\rangle$ โดยที่ $\langle$(30) (90)$\rangle$ เกิดขึ้นกับ ID 1,4 โดยที่ ID 4 ได้มีการซื้อ item(40, 70) ระหว่าง item 30 และ item 90. ส่วน Sequence pattern $\langle$30(40 70)$\rangle$ เกิดจาก ID2,4 โดยที่ ID2 ซื้อ 40 ตามด้วย 60 และ 70 ตามลำดับ

Customer Id	Transaction Time	Items Bought
1	June 25 '04	30
1	June 30 '04	90
2	June 20 '04	10, 20
2	June 15 '04	30
2	June 20 '04	40, 60, 70
3	June 25 '04	30, 50, 70
4	June 25 '04	30
4	June 25 '04	40, 70
4	June 25 '04	90
5	June 12 '04	90

รูปที่ 2.4 ฐานข้อมูลถูกจัดเรียงโดยรหัสและเวลาของกิจกรรมนั้น

Customer Id	Customer Sequence
1	⟨(30) (90)⟩
2	⟨(10 20) (30) (40 60 70)⟩
3	⟨(30 50 70)⟩
4	⟨(30) (40 70) (90)⟩
5	⟨(90)⟩

รูปที่ 2.5 การจัดเรียง sequence ของฐานข้อมูล

Sequence Patterns
with support > 25%
⟨(30) (90)⟩
⟨(30 40 70)⟩

รูปที่ 2.6 แสดงคำตอบ

ตัวอย่างของ Sequence ที่ไม่มีค่า minimum support คือ sequence ⟨(10 20) (30)⟩ ที่เกิดขึ้นเฉพาะใน ID2. ส่วน Sequence ⟨(30)⟩, ⟨(40)⟩, ⟨(90)⟩, ⟨(30) (40)⟩, ⟨(30) (70)⟩ และ ⟨(40) (70)⟩ ถึงแม้ว่ามีค่า minimum support แต่ว่าไม่ได้เป็น Maximal sequence.

L1

1-Sequences	Support
⟨1⟩	4
⟨2⟩	2
⟨3⟩	4
⟨4⟩	4
⟨5⟩	4

L2

2-Sequences	Support
⟨1 2⟩	2
⟨1 3⟩	4
⟨1 4⟩	3
⟨1 5⟩	3
⟨2 3⟩	2
⟨2 4⟩	2
⟨3 4⟩	3
⟨3 5⟩	2
⟨4 5⟩	2

L3

3-Sequences	Support
⟨1 2 3⟩	2
⟨1 2 4⟩	2
⟨1 3 4⟩	3
⟨1 3 5⟩	2
⟨2 3 4⟩	2

L4

4-Sequences	Support
⟨1 2 3 4⟩	2

รูปที่ 2.7 แสดง Large Sequences

2.3 การสร้าง Prediction Model

กล่าวถึงการสร้าง Prediction Model [6] ในหลายๆแบบด้วยกัน ซึ่ง Rule-Representation Methods มันเป็นเครื่องมือสำคัญในงานวิจัย โดยจะกล่าวได้ดังต่อไปนี้ การแยกเว็บเพจจากล็อกไฟล์ โดยใช้ Association rule ในการแยกแยะข้อมูลต่างๆ จากล็อกไฟล์

Web Logs และ User Sessions ในงานวิจัยนี้จะใช้ข้อมูลจาก Web server logs ซึ่งสิ่งสำคัญในลำดับต้นๆ คือการทำความเข้าใจเกี่ยวกับข้อมูลต่างๆ ที่ได้จากล็อกไฟล์ ซึ่งข้อมูลเหล่านี้จะนำมาใช้ประกอบในการสร้าง Prediction Model ต่างๆ ดังภาพที่ 2.8 นั้นจะแสดงล็อกไฟล์จากเว็บเซฟเวอร์ โดยปกติแล้วล็อกไฟล์นั้นประกอบด้วยเรคคอร์ดหลายๆ เรคคอร์ด ซึ่งจะมากจะน้อยขึ้นอยู่กับปริมาณการเยี่ยมชมของผู้เรียกใช้งาน ซึ่งการเยี่ยมชมเว็บไซต์ต่างๆ ของผู้เรียกใช้งานนั้น

จะถูกบันทึกเก็บไว้ในไฟล์ดังกล่าวนี้ โดยแต่ละผู้เรียกใช้งาน ที่ถูกบันทึกจะแยกเป็นแต่ละเรคอร์ด โดยจะประกอบไปด้วยส่วนสำคัญๆ ดังนี้ คือ

- User's host name หรือ IP address
- Time stamp ซึ่งจะบันทึกเวลาในการเรียกใช้งานของผู้เรียกใช้งาน
- HTTP method (GET, Post,etc) วิธีการเรียกใช้งานเว็บเพจของผู้เรียกใช้งาน
- URL ของ เว็บเพจที่ถูกเรียกใช้
- Status code ของการตอบสนองจาก HTTP
- Number of byte จำนวนหรือขนาดของไฟล์ที่ถูกเรียกใช้งาน ขนาดของไฟล์ดังกล่าวนี้จะเป็นตัวบอกว่าไฟล์ที่ผู้เรียกใช้งานเรียกใช้งานนั้นมีขนาดเท่าไหร่

```
in24.inetnebr.com - - [01/Aug/1995:00:00:01 -0400] "GET /shuttle/missions/sts-68/news/sts-68-mcc-05.txt HTTP/1.0" 200 1839
uplherc.upl.com - - [01/Aug/1995:00:00:07 -0400] "GET / HTTP/1.0" 304 0
uplherc.upl.com - - [01/Aug/1995:00:00:08 -0400] "GET /images/ksclogo-medium.gif HTTP/1.0" 304
```

รูปที่ 2.8 ข้อมูลที่ได้จาก Log File

เมื่อเราได้ล็อกไฟล์มาแล้วประการต่อมาคือการทำการตัด (Clean Processing)[5] บางส่วนที่ไม่จำเป็นออกไป ซึ่งสิ่งต่างๆเหล่านี้ดังเช่น ภาพต่างๆ, ไฟล์วีดีโอคลิป อื่นๆ ซึ่งส่วนประกอบที่ไม่จำเป็นเหล่านี้ ควรจะตัดออกไปเพื่อที่จะให้ข้อมูลที่เหลือนั้นเป็นข้อมูลที่จำเป็นสำหรับการสร้าง Prediction Model เท่านั้น

ขั้นตอนต่อไปคือการแยกเหตุการณ์ของแต่ละผู้เรียกใช้งาน (User Session) จากเว็บล็อกไฟล์ ที่ผ่านการกรั่นกรองเรียบร้อยแล้ว ดังตัวอย่าง สมมุติว่าเว็บล็อกไฟล์ประกอบด้วย การเรียกใช้งานเว็บเพจต่างๆดังต่อไปนี้

Time	User ID	Requested Document
00:00:01	U1	A
00:00:02	U2	B
00:00:03	U2	C
00:00:04	U3	D
00:00:05	U1	E
.....	.....	.....

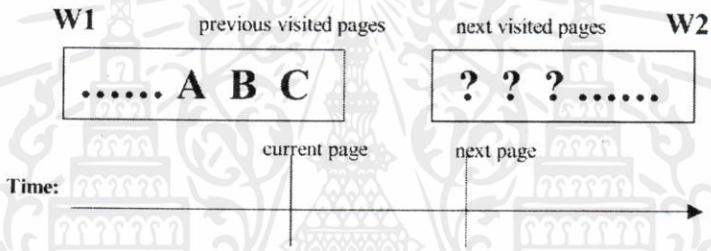
User ID	Session Sequence
U1	A, E, ...
U2	B, C,.....
U3	D,.....
.....	.....

รูปที่ 2.9 แสดงการเรียกใช้งานเว็บเพจของยูสเซอร์แต่ละท่าน

2.4 Moving Window Pairs และตารางล็อกไฟล์

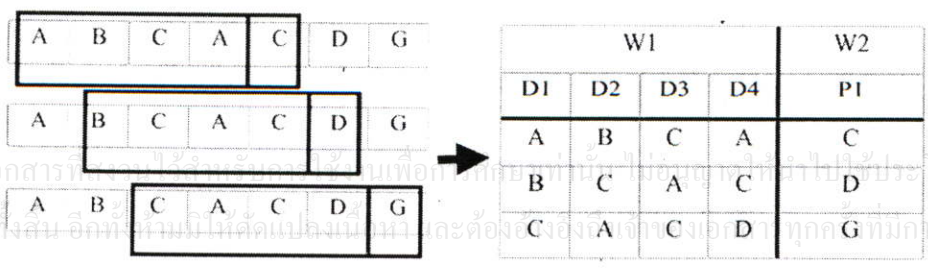
ในการค้นหารูปแบบการเรียกใช้งานของยูสเซอร์ (Access Pattern) ที่เป็น Association rules นั้น เราจะได้มาจากการทำประมวลผลจากข้อมูลเว็บล็อกไฟล์ โดยพิจารณาลำดับและเวลา เราได้ให้คำนิยามคำว่า "Moving window pair"[7] โดยที่มันจะประกอบด้วยสอง Adjacent window โดยที่หน้าต่างแรกนั้นเราเรียกว่า Antecedent window ซึ่งจะครอบคลุมเว็บเพจที่เคยเยี่ยมชมมาแล้ว โดยเรียงจากอดีตจนถึงปัจจุบัน โดยการเปรียบเทียบเวลาเป็นเงื่อนไขหลัก และในหน้าต่างที่สองเราเรียกว่า Consequent window ซึ่งจะครอบคลุมเว็บเพจที่เยี่ยมชมในอนาคต

ในการเรียกใช้งานหน้าต่างทั้งสองแบบนั้นในโอกาสต่อไปจะเรียก Antecedent window = W1 และ Consequent window = W2 ดังจะเห็นได้ว่าเว็บเพจในส่วน W1 นั้นจะประกอบไปด้วยเว็บเพจของ W2 ด้วย จะเห็นได้ว่า เว็บเพจใน W1 จะมีผลในการทำนายด้วยเช่นกัน



รูปที่ 2.10 แสดง Moving window pair

จากการประยุกต์ใช้งาน Moving pair window กับข้อมูลที่ได้ผ่านการกรั่นกรองแล้ว สิ่งที่เราจะได้รับคือตารางความสัมพันธ์ที่เป็นกฎความสัมพันธ์ ซึ่งมันจะถูกนำมาทำการไมนิ่ง โดยจำนวนของคอลัมพ์นั้นจะเป็นตัวบอกขนาดของ Moving pair window เราจะเรียกตารางดังกล่าวนี้ว่า Log Table โดยที่มันจะแสดงเหตุการณ์ทั้งหมดของเว็บล็อกนั้นเอง ดังรูป 10 แสดงตัวอย่างของเหตุการณ์ (A, B, C, A, C, D, G) เมื่อเรากำหนดขนาดของ W1 = 4 และ W2 = 1 ดังรูป 2.11 เมื่อมีการเลื่อน Moving Pair window แต่ละครั้งเราจะได้รูปแบบการเรียกใช้งานดังรูปด้านล่าง



รูปที่ 2.11 แสดงการเลื่อนของ Moving Pair Window

ขนาดของ W2 นั้นขึ้นอยู่กับว่าความสามารถของการทำนายว่าจะเป็นเท่าไร ดังตัวอย่าง หากให้ W2 = 2 นั้นหมายถึงว่าสามารถที่จะทำนายได้สองเพจ ในงานวิจัยนี้จะใช้ W2 = 1 เพื่อให้ง่ายในการปรับเปลี่ยนค่าพารามิเตอร์ต่างๆนั้นเอง

2.5 วิธีการ Rule Representation

จะกล่าวถึงการแตกกฎตามรูปแบบของ LHS➔RHS จากตารางล็อกตามที่ได้กล่าวไว้ข้างต้น กล่าวได้ว่า LHS คือเว็บเพจที่ได้มีการเรียกใช้ก่อนหน้าหรือที่ถูกเยี่ยมชมผ่านมาแล้ว และในขณะกัน RHS คือเพจที่จะถูกเรียกใช้งานต่อไป ตามกฎความสัมพันธ์ ดังตัวอย่างเช่น {A, B, C} ➔ D หมายความว่า ได้มีการคลิกเรียกใช้เว็บเพจหรือเยี่ยมชมเว็บเพจ A, เพจ B, เพจ C ตามลำดับ และต่อไปคาดว่าจะมีการเรียกใช้เว็บเพจ D นั้นเอง ในส่วนวิธีการแตก LHS เราจะนำเสนอ 5 วิธีการด้วยกัน [6]

- วิธีการแรกเรียกว่า the subset rules

ซึ่งกฎเหล่านี้นั้นจะเหมือนกับกฎความสัมพันธ์แบบเดิมซึ่งจะเป็นวิธีที่ไม่ได้นำลำดับและตัวที่อยู่ติดกันมาใช้ประกอบการวิเคราะห์ ดังนั้นเมื่อเรานำวิธีการกฎความสัมพันธ์ดังเช่น Aprioi method [10] มาใช้ในการประมวลผลกับตารางล็อกไฟล์ เราจะได้ subset rules ดังตารางที่ 2.2

ตารางที่ 2.2 แสดงตัวอย่างการใช้ วิธี The Subset Rule representation

W1	W2	Extracted Rules
A, B, C	D	{A, B, C} → D, {A, B} → D, {B, C} → D, {A, C} → D, {A} → D, {B} → D, {C} → D

- วิธีทีสอง เรียกว่า The sub sequence rules.

ซึ่งจะเป็นการนำลำดับของข้อมูล (Order) เข้ามาพิจารณาด้วย โดยจะเรียงตามลำดับการเกิดของเว็บเพจ ดังตารางที่ 2.3 ตามตัวอย่างเมื่อใช้วิธีการ sub sequence rules จะได้ A, B เมื่อ A เป็นเว็บเพจที่เกิดก่อน B กล่าวได้ว่าวิธีการนี้จะคล้ายคลึงกับอัลกอริธึมใน Sequential mining [4]

ตารางที่ 2.3 แสดงตัวอย่างการใช้วิธี The sub sequence Rule representation

W1	W2	Extracted Rules
A, B, C	D	(A, B, C) → D, (A, B) → D, (B, C) → D, (A, C) → D, (A) → D, (B) → D, (C) → D

- วิธีที่สาม เรียกว่า The latest-subsequence rules.

จะมีการนำลำดับของข้อมูล(Order) และข้อมูล ณ.ปัจจุบัน (Recency) เข้ามาพิจารณาด้วย ดังตัวอย่างเมื่อตารางล็อกได้ถูกแยกแยะด้วยวิธีการ the latest-subsequence rule จะได้ข้อมูลต่างๆ ดังตัวอย่าง จะสังเกตุเห็นว่าส่วนของ W1 นั้นเพจสุดท้ายจะเป็นข้อมูลชุดปัจจุบัน นั้นคือ C เสมอ

ตารางที่ 2.4 แสดงตัวอย่างการใช้วิธี The latest subsequence rule representation

W1	W2	Extracted Rules
A, B, C	D	(A, B, C) → D, (B, C) → D, (A, C) → D, (C) → D

- วิธีการที่ สี่ เรียกว่า The sub string rules.

จะมีการนำลำดับของข้อมูล (Order) และข้อมูลที่มีลักษณะอยู่ติดกัน (Adjacency) เข้ามาพิจารณาด้วย ดังตัวอย่างจะเห็นข้อมูลหรือกฎที่ผ่านการแยกแยะแล้วนั้นจะต้องมีการเรียงติดกันเสมอ ดังจะเห็นว่ากฎ A, C ➔ D นั้นจะไม่สามารถค้นพบด้วยวิธีดังกล่าวนี้ เพราะว่าเป็นข้อมูลที่ไม่ติดกัน

ตารางที่ 2.5 แสดงตัวอย่างการใช้งานของ the Substring rule representation

W1	W2	Extracted Rules
A, B, C	D	<A, B, C> → D, <A, B> → D, <B, C> → D, <A> → D, <B> → D, <C> → D

- วิธีการที่ ห้า เรียกว่า The latest-substring rules.

จะมีการนำลำดับของข้อมูล (Order), ข้อมูลที่มีลักษณะอยู่ติดกัน (Adjacency) และข้อมูล ณ. ปัจจุบัน (recency) เข้ามาพิจารณาด้วย

ตารางที่ 2.6 แสดงตัวอย่างการใช้งานของ the latest substring rule representation

W1	W2	Extracted Rules
A, B, C	D	<A, B, C> → D, <B, C> → D, <C> → D

โดยให้แต่ละกฎนั้นจะอยู่ในรูปแบบ LHS➔RHS โดยจะมีการนิยามค่า Support และ confidence factor ขึ้นมาดังต่อไปนี้

$$\sup = \frac{count(LHS, RHS)}{count(Table)} \qquad \sup(LHS) = \frac{count(LHS)}{count(Table)} \qquad conf = \frac{\sup(LHS, RHS)}{\sup(LHS)}$$

จากสมการข้างต้น ซึ่งจากการนับค่าจากจำนวนการเกิดของตารางจะได้ Count (Table) ซึ่งนับจากตารางล็อก มาใช้ในการคำนวณ

สิ่งสำคัญของการนำวิธีการแบบไมนิ่งกฎความสัมพันธ์ ดังกล่าวข้างต้นไปใช้งานนั้นคือ การกรองกฎบางกฎทิ้ง โดยค่า Minimum Support และ Minimum Confidence หากกฎใดมีค่าเหล่านี้ต่ำกว่าค่าที่กำหนดไว้ ซึ่งการกระทำเหล่านี้เรียกว่าการพรุนนิ่งเป็นวิธีการที่ได้กล่าวไว้ในอัลกอริธึมไมนิ่งความสัมพันธ์

2.6 การเตรียมการของข้อมูล

ประกอบด้วยขั้นตอนหลักดังต่อไปนี้

- กระบวนการแยกแยะ (Cleaning process)ข้อมูลที่ไม่ได้ใช้งาน (Embedded Object) หรือไม่จำเป็นทิ้ง
- กระบวนการระบุการใช้งานของยูสเซอร์แต่ละคน (User Identification)
- กระบวนแยกเหตุการณ์ของแต่ละยูสเซอร์ (User Session)
- กระบวนการระบุการใช้งานทรานเซ็คชั่น (Transaction Identification)

ตารางที่ 2.7 ตารางแสดงข้อมูล จากล็อกไฟล์เซอร์เวอร์

IP	Time Stamp	Method	Page	Agent	Code	Size
101	[29:23:53:25]	"GET	/a.html	HTTP/1.0"	200	1497
yyy	[29:23:53:26]	"GET	/docs/	HTTP/1.0"	200	56431
202	[29:23:53:36]	"GET	/b.html	HTTP/1.0"	200	1325
202	[29:23:53:53]	"GET	/c.html	HTTP/1.0"	200	1014
101	[29:23:54:15]	"GET	/b.html	HTTP/1.0"	200	4889
303	[29:23:54:16]	"GET	/c.html	HTTP/1.0"	200	2624
303	[29:23:54:18]	"GET	/d.html	HTTP/1.0"	200	935
101	[29:23:54:19]	"GET	/c.html	HTTP/1.0"	200	2788
404	[29:23:54:19]	"GET	/w.html	HTTP/1.0"	200	124

ตารางที่ 2.7 (ต่อ)

IP	Time Stamp	Method	Page	Agent	Code	Size
404	[29:23:54:19]	"GET	/x.html	HTTP/1.0"	200	124
202	[29:23:54:19]	"GET	/d.html	HTTP/1.0"	200	156
101	[29:23:54:19]	"GET	/d.html	HTTP/1.0"	200	2788
303	[29:23:54:19]	"GET	/e.html	HTTP/1.0"	302	-
404	[29:23:54:20]	"GET	/y.html	HTTP/1.0"	200	231
101	[29:23:54:25]	"GET	/c.html	HTTP/1.0"	200	991
101	[29:23:54:37]	"GET	/m.html	HTTP/1.0"	302	-
404	[29:23:54:37]	"GET	/z.html	HTTP/1.0"	200	4217
505	[29:23:54:40]	"GET	/m.html	HTTP/1.0"	200	1250
101	[29:23:55:01]	"GET	/n.html	HTTP/1.0"	200	51661
505	[29:23:55:21]	"GET	/n.html	HTTP/1.0"	200	4602
xxx	[29:23:55:23]	"GET	/docs/	HTTP/1.0"	200	56431

จากตารางแสดงข้อมูล ซึ่งประกอบด้วยจำนวนเรคคอร์ดต่างๆ โดยเซอร์เวอร์จะทำการบันทึกการเรียกใช้ข้อมูลโดยมีเวลาเป็นตัวบอกว่า เวลาดังกล่าวนั้นได้มีการจัดทำกิจกรรมอะไรบ้าง

2.6.1 กระบวนการแยกแยะข้อมูลที่ไม่ได้ใช้งาน (Embedded Object) หรือไม่จำเป็นทิ้ง

โดยในกระบวนดังกล่าวนี้วัตถุประสงค์ก็เพื่อ ทำการกรั่นกรอง เอาข้อมูลที่ไม่เหมาะสม หรือข้อมูลที่ไม่ได้ใช้ประโยชน์ทิ้งออกไป ซึ่งจะเป็นการช่วยลดการสิ้นเปลืองการใช้งานของหน่วยความจำนั้นเอง โดยปกติแล้วจะลบเรคคอร์ดที่ทำการบันทึกการเรียกใช้ไฟล์ จำพวกรูปภาพ ไฟล์ที่ไม่ได้มีการโหลดเว็บเพจ หรือวีดีโอต่างๆ ทิ้งออกไป จากตารางข้อมูล ล็อกไฟล์เซอร์เวอร์จะตัดทิ้งเรคคอร์ดดังต่อไปนี้

ตารางที่ 2.8 ตารางแสดงข้อมูลที่ผ่านกระบวนการ Clean Process

yyy	[29:23:53:26]	"GET	/docs/	HTTP/1.0"	200	56431
xxx	[29:23:55:23]	"GET	/docs/	HTTP/1.0"	200	56431

โดยข้อมูลสุดท้ายที่เหลือ เพื่อดำเนินการในกระบวนการต่อไป

ตารางที่ 2.9 แสดงข้อมูลที่ผ่านกระบวนการ Clean Process

Order	Request	Order	Request
1	101,[29:23:53:25],/a.html,	10	202,[29:23:54:19],/d.html,
2	202,[29:23:53:36],/b.html,	11	101,[29:23:54:19],/d.html,
3	202,[29:23:53:53],/c.html,	12	303,[29:23:54:19],/e.html,
4	101,[29:23:54:15],/b.html,	13	404,[29:23:54:20],/y.html,
5	303,[29:23:54:16],/c.html,	14	101,[29:23:54:25],/c.html,
6	303,[29:23:54:18],/d.html,	15	101,[29:23:54:37],/m.html,
7	101,[29:23:54:19],/c.html,	16	404,[29:23:54:37],/z.html,
8	404,[29:23:54:19],/w.html,	17	505,[29:23:54:40],/m.html,
9	404,[29:23:54:19],/x.html,	18	101,[29:23:55:01],/n.html,
		19	505,[29:23:55:21],/n.html,

2.6.2 กระบวนการระบุการใช้งานของแต่ละคน (User Identification)

ความจำเป็นในการบวนการระบุการใช้งานของผู้เรียกใช้งานเว็บไซต์ โดยขั้นตอนหรือกระบวนการดังกล่าวนี้ จะให้ผลลัพธ์ที่ทำให้ทราบว่าผู้เรียกใช้หรือยูสเซอร์แต่ละคนมีพฤติกรรมการเรียกใช้งานเว็บเพจอย่างไรบ้าง ความจำเป็นของกระบวนการดังกล่าวนี้ ดังที่ได้กล่าวข้างต้นแล้วนั้น ยังมีการตรวจสอบว่ารหัสของผู้เรียกใช้แต่ละคนมีรหัสอะไรบ้าง แตกต่างกันอย่างไร โดยที่ในการใช้งานจริงๆ นั้นผู้เรียกใช้เว็บเพจ อาจจะเล่นอินเตอร์เน็ตผ่าน เบอร์ไอพีกล่างซึ่งจะทำการแยกแยะได้ยาก นั้นเอง โดยกระบวนการจะมีการจัดเรียงยูสเซอร์ว่าแต่ละท่าน มีการใช้งานเว็บเพจใดก่อนหรือหลังนั้นเอง

การตรวจสอบสามารถกระทำได้โดยการนำเวลาหรือ Time Stamp ที่ได้มีการบันทึกไว้ในล็อกไฟล์มาทำการวิเคราะห์ โดยเวลาที่บันทึกในการเรียกใช้ดังกล่าวนี้ จะมีความแตกต่างกัน ทั้งนโดยทั่วไปแล้วจะมีการตั้งระยะเวลาช่องว่างการเรียกใช้งาน ดังตัวอย่างเช่น ในงานวิจัยนี้ได้มีการตั้งช่องว่างเวลา Gap Time [8] ไว้ที่ 2 ชั่วโมงหรือ 120 นาที ถึงแม้ว่า ยูสเซอร์คนเดียวกัน หากการเรียกใช้เว็บเพจแรก กับเว็บเพจต่อมาห่างกันเกิน 120 นาที เราจะถือว่าเป็นการเริ่มการใช้งานใหม่ของยูสเซอร์คนดังกล่าว นอกจากเวลาแล้วยังไม่การนำเอาวันที่ มาเป็นปัจจัยในการแบ่งแยกการใช้งานใหม่อีกด้วย

แสดงขั้นตอนในการระบุการใช้งานของแต่ละยูสเซอร์

ตารางที่ 2.10 แสดงผลการทำในส่วน User Identification

Order	User ID	Request
1	101	[29:23:53:25],/a.html,
2	101	[29:23:54:15],/b.html,
3	101	[29:23:54:19],/c.html,
4	101	[29:23:54:19],/d.html,
5	101	[29:23:54:25],/c.html,
6	101	[29:23:54:37],/m.html,
7	101	[29:23:55:01],/n.html,
8	202	[29:23:53:36],/b.html,
9	202	[29:23:53:53],/c.html,
10	202	[29:23:54:19],/d.html,
11	303	[29:23:54:16],/c.html,
12	303	[29:23:54:18],/d.html,
13	303	[29:23:54:19],/e.html,
14	404	[29:23:54:19],/w.html,
15	404	[29:23:54:19],/x.html,
16	404	[29:23:54:20],/y.html,
17	404	[29:23:54:37],/z.html,
18	505	[29:23:54:40],/m.html,
19	505	[29:23:55:21],/n.html,

2.6.3 กระบวนแยกเหตุการณ์ของแต่ละยูสเซอร์ (User Session)

ผลลัพธ์ของกระบวนการนี้จะเป็นการจัดเรียงลำดับว่า การเรียกใช้งานเว็บเพจของผู้เรียกใช้แต่ละท่าน มีการเรียกใช้เว็บเพจอะไรบ้าง ตั้งแต่เพจแรกจนถึงเพจสุดท้าย

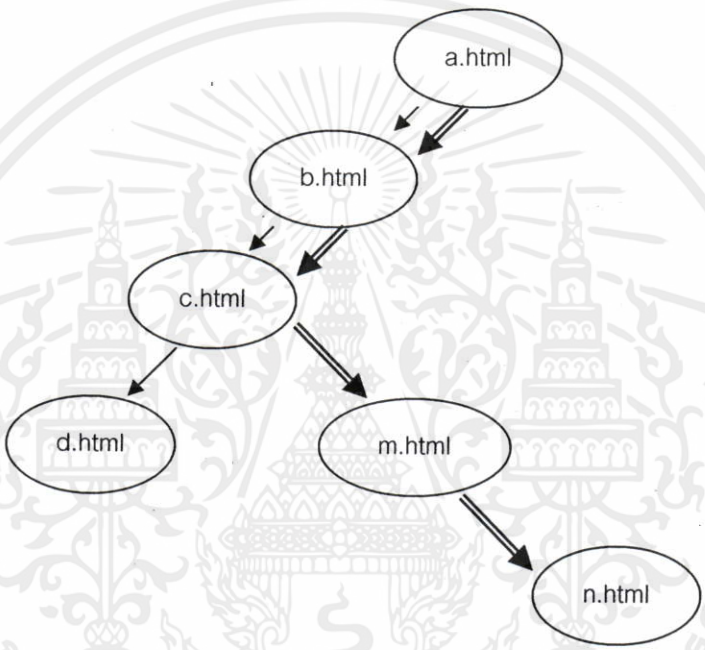
ตารางที่ 2.11 ตารางแสดงผลของ User Session

/a.html,/b.html,/c.html,/d.html,/c.html,/m.html,/n.html,
/b.html,/c.html,/d.html,
/c.html,/d.html,/e.html,
/w.html,/x.html,/y.html,/z.html,
/m.html,/n.html,

2.6.4 กระบวนการระบุการใช้งานทรานเซ็คชั่น (Transaction Identification)

กระบวนการระบุการใช้งานทรานเซ็คชั่นนี้จะใช้วิธีการ Maximal Forward Reference [1] จะสังเกตุได้ได้ว่า ในเรคคอร์ด ที่ 1 นั้นเดิมที่ข้อมูลการเรียกใช้จะเป็นดังนี้

/a.html,/b.html,/c.html,/d.html,/c.html,/m.html,/n.html,



รูปที่ 2.12 แสดงทางเดินของยูสเซอร์ เมื่อมีการประมวลผล MF Algorithm

เมื่อมีการประมวลผล Maximal Forward Reference จะได้ทรานเซ็คชั่นเพิ่มขึ้น คือ

- /a.html,/b.html,/c.html,/d.html,
- /a.html,/b.html,/c.html,/m.html,/n.html,

สามารถอธิบายการทำงานได้ดังต่อไปนี้

จะเห็นได้ว่าเส้นทางที่หนึ่ง การเรียกใช้งานของผู้เรียกเข้าชมเว็บไซต์ จะเริ่มต้นที่ a.html โดยเส้นทางดังกล่าวนี้จะเรียกใช้เว็บเพจ เพจสุดท้ายหรือเพจที่ลึกที่สุดคือ d.html

และในเส้นทางที่ สอง จะ มีการสิ้นสุดที่การเรียกใช้เว็บเพจ n.html

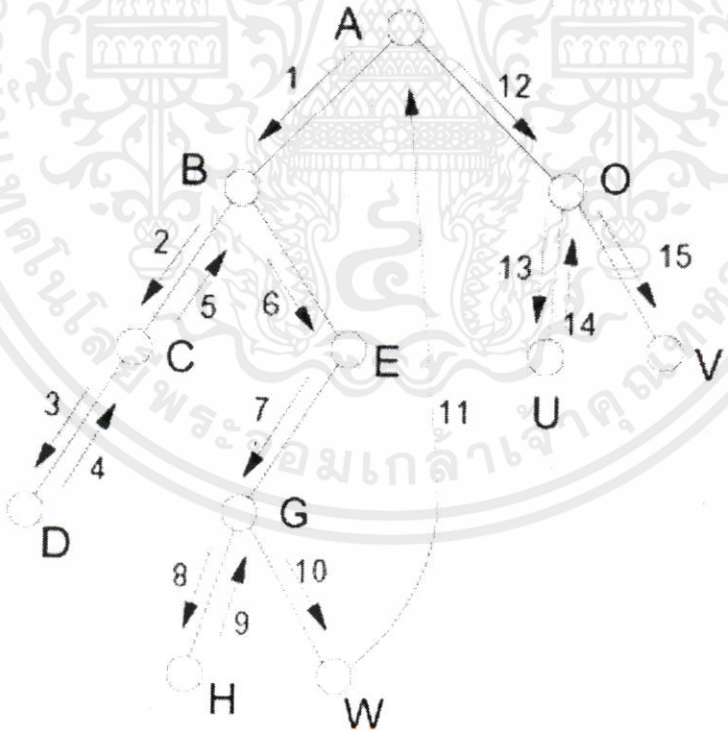
โดยผลลัพธ์ที่ได้จะเป็นข้อมูลที่จะใช้ในการประมวลผลของอัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์

ตารางที่ 2.12 ตารางแสดงผลการระบุการใช้งานทรานเซ็คชั่น

/a.html,/b.html,/c.html,/d.html,
/a.html,/b.html,/c.html,/m.html,/n.html,
/b.html,/c.html,/d.html,
/c.html,/d.html,/e.html,
/w.html,/x.html,/y.html,/z.html,
/m.html,/n.html,

2.6.5 ลักษณะของ MF algorithm

ลักษณะของเว็บเพจที่ต่อถึงกันหรือลิงค์ถึงกันนั้น การท่องเว็บของ User บ่อยครั้งที่กดปุ่ม Back เพื่อกลับไปยังหน้าเพจที่เคยเยี่ยมชมมาแล้วนั้น แทนที่จะทำการเปิดหน้าเว็บเพจหน้าใหม่ ดังเช่น เมื่อ user เข้าเยี่ยมชมหน้าเว็บเพจ โดยเริ่มต้นที่หน้าเพจ {A, B, C, D, E, F} ตามลำดับ และ User ได้กดปุ่ม Backward กลับมายังหน้าเพจ E ซึ่งจะทำให้เกิด เหตุการณ์ที่เรียกว่า Maximal Forward Reference [1]



รูปที่ 2.13 เป็นตัวอย่างของ Traversal Pattern

อัลกอริธึมในการค้นหา Maximal Forward Reference (MF) โดยเห็นได้ดังรูปที่ 2.13 โดยประกอบด้วยเส้นทางเดิน {A, B, C, D, C, B, E, G, H, G, W, A, O, U, O, V} หลังจากเราใช้ Algorithm MF จะได้กลุ่มเส้นทางของ Maximal Forward Reference ดังนี้ {ABCD, ABEGH,

ABEGW, AOU, AOV} จากเส้นทางข้างต้นนั้นจะเป็นที่มาของ Large Reference Sequence โดยกล่าวได้ว่า หากเส้นทางไหนที่มีการเกิดบ่อยครั้ง ที่มีจำนวนพอเพียงเท่ากับค่า Minimal Support

A large k-reference ก็คือ Large Reference Sequence ซึ่งประกอบด้วย k element โดยที่จะแสดงสัญลักษณ์ว่า L_k โดยประกอบด้วย Candidate set (C_k) โดยที่ค่า Large Reference Sequence สามารถนำมาหาค่าของ Maximal Reference Sequences ซึ่งหามาได้โดยง่าย กล่าวได้ว่า Maximal Reference Sequences ก็คือ Large Reference Sequence ซึ่งมิได้ประกอบอยู่ใน Maximal Reference Sequence ดังตัวอย่าง สมมุติให้ {AB, BE, AD, CG, GH, BG} แสดงกลุ่มของ Large 2 Reference (L2) และ {ABE,CGH} คือกลุ่มของ Large 3 Reference(L3) และจะได้ค่า Maximal Reference Sequence ดังนี้ AD, BG, ABE, CGH

โดยปกติ Log database ประกอบด้วยลิงค์ต่าง ซึ่งจะมีเป็นคู่ คือต้นทาง (Source) และปลายทาง (Destination) โดยที่เมื่อ User เข้ามาครั้งแรกนั้น เรามิอาจจะทราบที่มาได้ว่ามาจากไหน ดังนั้นจึงให้ค่าเริ่มต้น หรือ Source เป็นค่า Null โดยเมื่อเราทำการแปลง log database แล้วจะได้ Traversal sequence {(s1, d1), (s2, d2),.....(sn, dn)} และเมื่อผ่านขั้นตอนของ Algorithm MF ก็จะถูกเก็บยัง DF Database

Algorithm Maximal Forward Reference (MF)

Step 1: Set i=1 and string Y to null for initialization, where string Y is used to store the current forward reference path. Also, set the flag F=1 to indicate a forward traversal

Step 2: Let A=si and B=di

If A is equal to null then

begin

Write out the current string Y (if not null) to the database DF;

Set string Y=B

Go to Step 5

Step 3: If B is equal to some reference (say the j-th reference) in string Y then

/* this is a cross-referencing back to a previous reference */

begin

If F is equal to 1 then write out string Y to database;

Discard all the references after the j-th one in string Y;

F = 0

Go to Step 5

รูปที่ 2.14 แสดงอัลกอริธึม Maximum Forward Reference

End

Step 4: Otherwise, append B to the end of string Y.

/* we are continuing a forward traversal */

If F is equal to 0, set F=1

Step 5: Set i=i+1. If the sequence is not completed scanned then go to Step 2.

รูปที่ 2.14 (ต่อ)

ตารางที่ 2.15 ตารางตัวอย่างผลลัพธ์เมื่อใช้ Algorithm MF

	ini	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
si	null	A	B	C	D	C	B	E	G	H	G	W	A	O	U	O
di	A	B	C	D	C	B	E	G	H	G	W	A	O	U	O	V

move	string Y	output DF
1	AB	-
2	ABC	-
3	ABCD	-
4	ABC	ABCD
5	AB	-
6	ABE	-
7	ABEG	-
8	ABEGH	-
9	ABEG	ABEGH
10	ABEGW	-
11	A	ABEGW
12	AO	-
13	AOU	-
14	AO	AOU
15	AOV	AOV (END)

ดังจะเห็นว่าค่าเริ่มต้นของ Source นั้นกำหนดให้เป็นค่า null เพราะเราไม่ทราบถึงแหล่งที่มาของ user จากรูปที่ 2.13 นั้นจะมีการเขียนข้อมูลลงใน database โดยหากดูในตารางรูปที่ 2.14 จะมีการเขียน ลงใน DF ในขั้นตอนที่ 4-th, 9-th, 11-th และ 15-th

2.7 การสร้างต้นไม้เพื่อการพรุนนิ่ง

วิธีการสำคัญอีกวิธีหนึ่งเมื่อเราได้กฎมาแล้วคือการนำกฎเหล่านี้มาสร้างต้นไม้ [6] ก็คือ Latest substring index tree (LSIT) ซึ่งเป็นวิธีการสร้าง prediction model ที่มีประสิทธิภาพและมีการใช้หน่วยความจำน้อยที่สุด โดยเรามีหลักการสร้างต้นไม้ ดังต่อไปนี้

- แต่ละ กฎ เปรียบได้ดัง โหนดหนึ่งโหนด
- โหนดตัวบนแสดงเป็นโหนดแม่ (parent node) ซึ่งโหนดตัวล่างจะเป็นโหนดลูก (children node)
- รูทของต้นไม้จะเป็น default rule

ดังตัวอย่างต่อไปนี้จะประกอบด้วย 6 โหนด ซึ่งสร้างมาจากการทำนายที่มีผลการทำนาย 5 กฎด้วยกัน

เมื่อสร้าง LSIT เรียบร้อยแล้วจะเห็นได้ว่าจะมีการเลื่อนหรือเปลี่ยนแปลงโหนดบางโหนด ด้วยวิธีการดังกล่าวต่อไปนี้ ขั้นตอนการพรุนนิ่ง (pruning process) [6]จะใช้หลักการ post-order traverse โดย

- หากโหนดลูกมีค่า Confidence น้อยกว่า confidence แม่ โหนดดังกล่าวก็จะถูกตัดทิ้ง
- หากโหนดลูกมีการทำนาย เหมือนกับโหนดแม่ โหนดดังกล่าวก็จะถูกตัดทิ้งเช่นกัน

จะมีการ โปรโมต บางโหนดภายใต้โหนดแม่ดังกล่าวนั้น ดังรูปที่ 2.15

โหนด <C> ➔ N จะถูกพรุน เพราะมีค่า confidence น้อยกว่า โหนดแม่

โหนด <A> ➔ M จะถูกพรุน เพราะว่ามีการทำนายเว็บเพจต่อไป เหมือนกับโหนดแม่

หลังจาก 2 โหนดดังกล่าวนี้ถูกพรุน

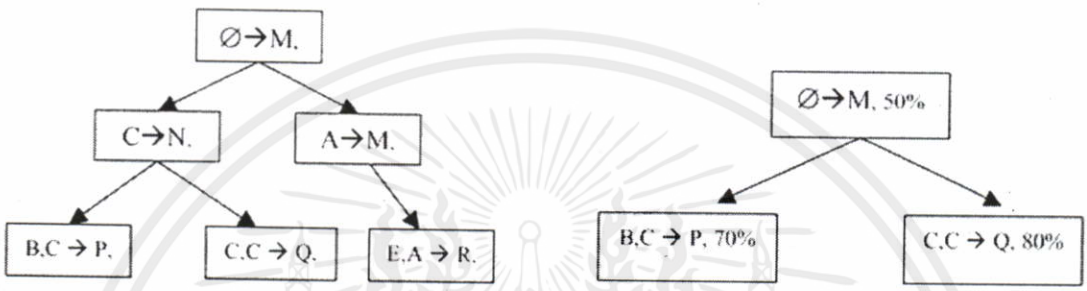
โหนด <E,A> ➔ R ก็จะถูกโปรโมต และเมื่อเปรียบเทียบจะเห็นว่ามีค่า Confidence น้อยกว่าโหนดแม่ ดังนั้นจึงถูกพรุนในลำดับต่อไป

หลักจากเสร็จสิ้นการพรุนจะได้ LSIT ดังรูปที่ 2.15

กรณีทดสอบหาก <C> ➔ ? ซึ่งจะให้การค้นหาโดยเพจ C เมื่อทำการค้นหาแล้วไม่เจอเพจ C ดังนั้นจะได้กฎสุดท้ายที่ได้คือ รูท M ดังนั้นการทำนายจึงให้เพจ M

กรณีทดสอบหาก <A, B, C> ➔ ? จะทำการค้นหาโดยให้เพจ C เมื่อทำการค้นหาไม่เจอต่อไปจะให้เส้นทาง B, C ซึ่งจะพบโหนด P ดังนั้นจึงไม่มีความจำเป็นในการค้นหาความสัมพันธ์ต่อไปแล้ว การทำนายจึงเป็น เพจ P

Rules	Pessimistic Confidence
Ø → M	50%
<C> → N	40%
<A> → M	30%
<B,C> → P	70%
<C, C> → Q	80%
<E, A> → R	40%



รูปที่ 2.16 แสดงข้อมูลและวิธีการสร้างต้นไม้พร้อมทั้งการพรุนโหนดทิ้ง

โดยทั่วไป LSIT จะช่วยลดขนาดในการแบ่งชั้น (Classifier) ซึ่งจากการเปรียบเทียบจะเห็นได้ว่าขนาดของ Latest-substring rules set เมื่อผ่านการ LSIT pruning ขนาดของกฎที่เหลือจะมีขนาดน้อยลง จากงานวิจัยของ Tianyi Li [1] ระบุว่าด้วยวิธีการดังกล่าวนี้กฎต่างๆที่ไม่จำเป็นจะถูกตัดทิ้งเป็นปริมาณสัดส่วนถึง 4/5 จะถูกตัดทิ้งออกไป

จากทฤษฎีที่เกี่ยวข้องที่ได้กล่าวมาข้างต้นนั้น จะเห็นได้ว่าหากข้อมูลที่เป็นลักษณะเว็บไซต์ที่ได้มาจากล็อกไฟล์เซอร์นั้น การเรียงลำดับของเว็บเพจจะมีส่วนสำคัญมาก การจะใช้หลักการ อัลกอริธึมแบบค้นหาความสัมพันธ์ Association Rule โดยผลลัพธ์ด้วยอัลกอริธึมดังกล่าวนี้จะให้ผลลัพธ์เกิดการสลับที่กันของเว็บเพจ หากจะการอัลกอริธึมแบบจัดเรียงลำดับ Sequential Pattern ซึ่งวิธีการดังกล่าวนี้คำนึงเฉพาะลำดับก่อนและหลังเท่านั้น มิได้ให้ความสำคัญกับการเรียกใช้เว็บเพจที่ต่อเนื่องกัน ซึ่งผลที่ออกมาจากการใช้อัลกอริธึมดังกล่าวนี้จะให้ผลลัพธ์ที่มีการกระโดดของเว็บเพจ

ดังนั้นในวิทยานิพนธ์นี้จึงได้คิดวิธีการในการปรับปรุงประสิทธิภาพการค้นหารูปแบบลำดับของเว็บไซต์ โดยมีการแก้ปัญหาการค้นหารูปแบบ โดยใช้หลักการค้นหาลำดับ ซึ่งในการค้นหานั้นจะใช้วิธีการสับกฎเป็นลักษณะ Adaptive windows ที่สามารถค้นหากฎแบบไม่จำกัดขนาด อีกทั้งมีการนำลำดับของข้อมูล ที่มีลักษณะอยู่ติดกัน พร้อมกับข้อมูลเว็บเพจปัจจุบัน เข้ามาพิจารณาด้วย ส่งผลให้กฎ ที่ได้นั้นมีประสิทธิภาพในเรื่องความสัมพันธ์ได้เป็นอย่างดี ในบทต่อไปจะกล่าวถึงงานวิจัยของวิทยานิพนธ์ฉบับนี้

บทที่ 3
อัลกอริทึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์

ขั้นตอนการทำงานที่ใช้ในงานวิจัยนี้ประกอบด้วยกระบวนการต่างๆ ดังต่อไปนี้

- การนำข้อมูลจากล็อกไฟล์เซอร์เวอร์ (Web Log File)
- กระบวนการเตรียมข้อมูล (Data Preprocessing)
  - การแยกแยะข้อมูลที่ไม่ได้ใช้งานทิ้ง
  - กระบวนการระบุการใช้งานของผู้ใช้เว็บเพจแต่ละคน
  - กระบวนการแยกเหตุการณ์ของแต่ละผู้เรียกใช้เว็บเพจ
  - กระบวนการระบุทรานเซ็กชั่น
- อัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์
  - ค้นหารูปแบบ Pattern Discovery (Join Operation)
  - กระบวนการคัดกฎความสัมพันธ์ทิ้ง Reduce Rule (Support Factor)
- กระบวนการสร้างต้นไม้ (Building Tree)
  - กระบวนการพรุ่นนิ่ง Pruning Tree (Confidence Factor)
- กระบวนสร้างโมเดลเพื่อการทำนาย (Prediction Model)

โดยข้อมูลแต่ละเรคคอร์ดที่เซอร์เวอร์ได้บันทึกไว้จะมีข้อมูลที่หลากหลาย แต่จะเห็นว่าข้อมูลบางชนิดเราไม่จำเป็นในการนำมาประมวลผล ดังนั้นเราจึงทำการตัดทิ้งข้อมูลดังกล่าวทิ้งไป กระบวนการดังกล่าวนี้คือ กระบวนการเตรียมการของข้อมูล นอกจากนั้นยังได้กล่าวถึงการสร้างต้นไม้เพื่อการพรุ่นนิ่ง ซึ่งได้กล่าวไว้โดยละเอียดในบทที่ 2 แล้ว

ในบทนี้จะกล่าวถึงการทำงานของอัลกอริทึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์ ถึงการทำงานของอัลกอริธึมจะประกอบไปด้วย 3 กระบวนการหลักๆ ดังต่อไปนี้ คือ

- กระบวนการค้นหารูปแบบความสัมพันธ์ของกฎ
- กระบวนการคัดกฎความสัมพันธ์ทิ้ง
- กระบวนการทำนายเว็บเพจ

โดยข้อมูลต่างๆ ที่เป็นส่วนอินพุทของอัลกอริธึมนั้น จะได้มาจากกระบวนการเตรียมข้อมูล โดยกระบวนการสุดท้าย คือการระบุทรานเซ็กชั่นของผู้เข้าชมเว็บไซต์แต่ละคน ว่ามีการเข้าชมหรือมีพฤติกรรมอย่างไรในการเลือกชมเว็บไซต์ โดยข้อมูลดังกล่าวนี้จะนำมาเป็นข้อมูลทางด้านอินพุทของอัลกอริธึมต่อไป การทำงานของอัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับการเรียกใช้งานเว็บไซต์นั้น จะอธิบายขั้นตอนการทำงานของอัลกอริธึซึ่งจะกล่าวต่อไปและพร้อมทั้งยกตัวอย่างประกอบเพื่อให้เข้าใจการทำงานได้โดยง่าย

3.1 การค้นหารูปแบบความสัมพันธ์ของกฎ

ในการสร้างกฎความสัมพันธ์ของอัลกอริทึม Apriori นั้นจะประกอบด้วยเซตข้อมูล 2 เซต คือ Candidate Itemset (C) และ Frequent Itemset (F) หรือ Large Itemset (L)อธิบายได้ดังนี้

1. Candidate Itemset คือกลุ่มข้อมูลที่เป็นตัวเลือกสำหรับการสร้าง Frequent Itemset แทนด้วยสัญลักษณ์ Ck
2. Frequent Itemset หรือ Large Itemset คือกลุ่มข้อมูลที่ใช้สำหรับการสร้างกฎความสัมพันธ์ โดยแต่ละ Item ที่เป็นสมาชิกของเซตนี้ต้องผ่านค่าสนับสนุนน้อยที่สุดแล้ว แทนด้วยสัญลักษณ์ F หรือ L สำหรับงานวิจัยนี้ใช้ Lk

นอกจากนั้นแล้วจะมีตัวแปรพารามิเตอร์ 2 ตัวแปรด้วยกัน ดังที่ได้ให้คำนิยามไว้ตอนต้น คือ Support Factor และ Confidence Factor โดยที่

1. Support Factor นั้นจะนำมาใช้งานในกระบวนการหา Large Itemset (Lk) ซึ่งค่า Lk นั้นจะเกิดขึ้นมาจากความถี่ของ Ck ที่ผ่านค่า Minimum Support นั้นเอง

2. Confidence Factor นั้นจะนำมาใช้ประโยชน์ในกระบวนการลดทอนกฎหรือ Pruning Process โดยที่กฎที่ผ่านกระบวนการต่างๆของอัลกอริธึมจะมีจำนวนมาก ดังนั้นการลดทอนกฎสามารถกระทำได้โดยการสร้างต้นไม้ และใช้ Confidence Factor ในการตัดทอนกฎที่ไม่จำเป็นทิ้ง

3.1.1. อัลกอริทึมสำหรับค้นหากฎความสัมพันธ์

อัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์ ที่ใช้ในการค้นหารูปแบบความสัมพันธ์ของเว็บเพจ ของงานวิจัยนี้ แสดงดังรูปที่ 3.1 และการ Join Operation ในรูปที่ 3.2

Begin main:

L1 = {large 1-sequences};

For { k=2; $L_{k-1} \neq \varnothing$; k++} do

Begin

C_k New candidate generated from L_{k-1} (See a new candidate generation).

for each web-page sequence c in the web transaction table do

Increment the count of all candidates in C_k that are contained in c.

L_k = Candidates in C_k with minimum support.

End

รูปที่ 3.1 A New Algorithm for Web Sequential Pattern Discovery

The Candidate generate function take as argument L_{k-1}, the set of all large (k-1)-sequences. The function work as follows. First. Join L_{k-1} with L_{k-1}

$$a_1a_2a_3...a_{k-1} \otimes b_1b_2b_3...b_{k-1} = \begin{cases} \phi & \text{Otherwise} \\ a_1\, a_2\, a_3 \ldots a_n & \\ a_2 = b_1, a_2 = b_3, ..., a_{k-1} = b_{k-2} & \end{cases}$$

รูปที่ 3.2 A new Candidate Generations

กระบวนการทำงานของอัลกอริธึมในส่วนของ การค้นหารูปแบบความสัมพันธ์นั้นได้นำแนวคิดจากวิธีการของ Apriori Algorithm มาประยุกต์ใช้งาน ซึ่งในงานวิจัยนี้ ข้อมูลที่ใช้ในการทดลองจะเป็นข้อมูลจากเว็บเซอร์เวอร์ จากปัญหาที่ได้กล่าวไว้ในบทก่อนหน้านี้แล้วว่า ข้อมูลประเภทเว็บเพจ ที่ได้มาจากเว็บล็อกเซอร์เวอร์ นั้นจะต้องให้ความสำคัญกับ ลำดับ การอยู่ชิดติดกันของเว็บเพจ และเว็บเพจสุดท้ายด้วย ดังปัญหาที่ได้กล่าวในเรื่องของการกระโดดของข้อมูลในอัลกอริธึม Mining Sequence Pattern [11] จากปัญหาทำให้มีแนวคิดในการ Join Operation ขึ้นมาใหม่โดยจุดมุ่งหมาย เมื่อใช้งานกับข้อมูลประเภทเว็บเพจนั้นเอง ทำให้ผลลัพธ์ที่ออกมานั้นมีความถูกต้องและแม่นยำมากขึ้น

จากรูปที่ 3.1 แสดงอัลกอริทึมสำหรับค้นหารูปแบบลำดับของเว็บไซต์ ของงานวิจัยนี้เริ่มต้นจากข้อมูลนำเข้าดังนี้ เพื่อความง่ายต่อการเข้าใจวิธีการทำงานของอัลกอริทึมสำหรับค้นหารูปแบบลำดับของเว็บไซต์นี้ ดังนั้นจะแสดงวิธีการค้นหากฎความสัมพันธ์ในแต่ละขั้นตอนพร้อมยกตัวอย่าง

โดยข้อมูลในตารางที่ 3.1 นั้นจะแสดงข้อมูลของทรานเซ็คชั่นต่างๆ พร้อมที่จะเป็นข้อมูลอินพุทในกระบวนการค้นหารูปแบบลำดับของเว็บไซต์ ดังเช่น

ทรานเซ็กชั่นที่ 1 มีการเรียกใช้เว็บเพจ ดังนี้ a.html,b.html,c.html,y.html,z.html,

ทรานเซ็กชั่นที่ 2 มีการเรียกใช้เว็บเพจ ดังนี้ m.html,n.html,o.html,p.html,q.html,

นั้นหมายความว่า แต่ละทรานเซ็กชั่นจะมีการเรียกใช้เว็บเพจตามลำดับ โดยเว็บเพจดังกล่าวนี้ได้ผ่านกระบวนการต่างๆในส่วนการเตรียมข้อมูล เรียบร้อยแล้ว ดังนั้นจะเห็นว่า ข้อมูลเว็บเพจของแต่ละทรานเซ็กชั่น จะไม่ซ้ำกันเลย เมื่อนำข้อมูลจากทรานเซ็กชั่นไปประมวลผลในส่วนแรกของอัลกอริธึมใหม่นี้ จะเป็นการนับจะนวนการเกิดขึ้นของแต่ละเว็บเพจ หรือไอเท็ม นั้นเอง โดยจะมีการนับจำนวนการเกิดของแต่ละไอเท็มทั้งหมด ในฐานข้อมูล เพื่อให้ได้มาซึ่งจำนวนการเกิดที่แท้จริง โดยหากว่าไอเท็มใด มีการเกิดขึ้นมาน้อย นั้นหมายถึงไอเท็มดังกล่าวนั้นจะถูกตัดทอนทิ้ง

ตารางที่ 3.1 แสดงข้อมูลทรานเซ็คชั่นของการเรียกใช้เว็บเพจ

ลำดับ	รายการทรานเซ็คชั่น
1.	a.html,b.html,c.html,y.html,z.html,
2.	m.html,n.html,o.html,p.html,q.html,
3.	m.html,n.html,o.html,p.html,
4.	x1.html,c.html,d.html,o.html,p.html,
5.	x1.html,c.html,d.html,e.html,
6.	x.html,y.html,z.html,
7.	x4.html,x5.html,d.html,e.html,y.html,
8.	x4.html,x5.html,d.html,e.html,y.html,
9.	m.html,n.html,o.html,p.html,q.html,

จากที่ได้กล่าวไว้ต้อนต้นแล้วนั้น จะมีตัวแปรพารามิเตอร์หนึ่งตัว ที่ให้ในการกำหนดให้ค่าสนับสนุนน้อยที่สุด (Minimum Support) มีค่าเท่ากับ 0.12 โดยที่หากการเกิดของ Pattern ใดๆ นั้นมีค่าการเกิดน้อยกว่าค่าดังกล่าวนี้ จะถือว่า Pattern ควรที่จะตัดทิ้งออกไป โดยการทำงานของอัลกอริธึมในการค้นหารูปแบบกฎความสัมพันธ์ สามารถอธิบายได้ดังต่อไปนี้

1. ขั้นตอนที่ (1) เป็นการหา Candidate C_0 ซึ่งก็คือการค้นหาว่าแต่ละเว็บเพจนั้นเกิดขึ้นเป็นจำนวนมากน้อยเท่าไหร่ พร้อมทั้งหาค่าสนับสนุนของแต่ละ Itemset แสดงดังตารางที่ 3.2

2. ขั้นตอนที่ (2) เป็นการหา Large Itemset (L_0) ซึ่งกล่าวได้ว่าสมาชิกของ C_0 ที่ผ่านค่าสนับสนุนน้อยที่สุด (Minimum Support) แล้วนั้นเอง แสดงดังตารางที่ 3.3

ตารางที่ 3.2 แสดงจำนวนความถี่ของการเกิดขึ้นของ Pattern ต่างๆ

จำนวนความถี่	Pattern	จำนวนความถี่	Pattern
1	a.html,	4	p.html,
1	b.html,	2	q.html,
3	c.html,	1	x.html,
4	d.html,	2	x1.html,
3	e.html,	2	x4.html,
3	m.html,	2	x5.html,
3	n.html,	4	y.html,
4	o.html,	2	z.html,

ตารางที่ 3.3 แสดง Pattern ที่ผ่านค่า Minimum Support

Support	Pattern	Support	Pattern
0.333	c.html,	0.444	p.html,
0.444	d.html,	0.222	q.html,
0.333	e.html,	0.222	x1.html,
0.333	m.html,	0.222	x4.html,
0.333	n.html,	0.222	x5.html,
0.444	o.html,	0.444	y.html,
		0.222	z.html,

ดังจะเห็นว่า Pattern ที่จะถูกตัดทิ้งไป ซึ่งจะมีการเกิดขึ้นของ Pattern เพียงครั้งเดียวเท่านั้น คือ a.html, b.html, x.html

3. ขั้นตอนที่ (3) เป็นการหา Candidate C_1 ซึ่งเกิดการนำ L_0 มาสร้างนั้นเอง แสดงดังตารางที่ 3.4

ตารางที่ 3.4 แสดงการสร้าง Candidate (C1) จาก L_0

L_0	C_1
c.html,	c.html,d.html
d.html,	c.html,e.html
e.html,	d.html,c.html
…..	d.html,e.html
	e.html,c.html
	e.html,d.html
	…..

ในการหาค่า Candidate (k = 2..n) นั้นอัลกอริธึมจะมีการหาค่า Candidate Generations โดยจะประมวลผลตามรูปที่ 3.2

4. ขั้นตอนที่ (4) เป็นการหา Large Sequence L_1: ซึ่งกล่าวได้ว่าสมาชิกของ C_1 ที่ผ่านค่าสนับสนุนน้อยที่สุด (Minimum Support) แล้วนั้นเอง อีกทั้งในขั้นตอนนี้มีการคำนวณหาค่า Confidence อีกด้วย โดยจะได้ดังตารางที่ 3.5

ตารางที่ 3.5 แสดงข้อมูลของ Large Sequence: L_1

ลำดับ	Support	Confidence	Pattern
1	0.22222	0.66667	c.html,d.html,
2	0.33333	0.75	d.html,e.html,
3	0.22222	0.66667	e.html,y.html,
4	0.33333	1	m.html,n.html,
5	0.33333	1	n.html,o.html,
6	0.44444	1	o.html,p.html,
7	0.22222	0.5	p.html,q.html,
8	0.22222	1	x1.html,c.html,
9	0.22222	1	x4.html,x5.html,
10	0.22222	1	x5.html,d.html,
11	0.22222	0.5	y.html,z.html,

5. ขั้นตอนที่ (5) ซึ่งจะต้องมีการสร้าง Candidate: C_2 จาก Large Sequence $L_1 \otimes L_1$ ซึ่งจะแสดงผลลัพธ์การกระบวนการ Join Operation ดังตารางที่ 3.6 โดยกระบวนการทำงานของอัลกอริธึมนั้นดูได้จากรูปที่ 3.2

ในการสร้าง C_k นั้นเราได้ให้คำนิยามการ Join Operation ใหม่ซึ่งดังจะได้ทราบมาแล้วว่าในลักษณะของ Sequence Pattern นั้นจะเกิดการกระโดดกันของ Item ที่เกิดจากการ Join ดังนั้น Sequence ที่ได้จะขาดลำดับ (Order information) เมื่อเรานำมาประยุกต์ใช้กับเว็บเพจจะทำให้เว็บเพจเกิดการกระโดดได้เช่นกัน ดังนั้นจะส่งผลให้ความแม่นยำลดลงไป หรือการทำนายผิดเพี้ยนไปด้วย ดังตัวอย่างนี้เป็น: A new Candidate Generation

$$D, E \otimes E, D \Rightarrow D, E \rightarrow D$$
$$D, E \otimes E, Y \Rightarrow D, E \rightarrow Y$$
$$E, D \otimes D, E \Rightarrow E, D \rightarrow E$$
$$D, E \otimes E, Z \Rightarrow D, E \rightarrow Z$$

ตารางที่ 3.6 แสดงการ Join Operation $L_1 \otimes L_1 = C_2$

Join Operation ($L_1 \otimes L_1$)	Pattern (C_2)
c.html,d.html, ⊗ d.html,e.html,	c.html,d.html,e.html,
d.html,e.html, ⊗ e.html,y.html,	d.html,e.html,y.html,
e.html,y.html, ⊗ y.html,z.html,	e.html,y.html,z.html,
m.html,n.html, ⊗ n.html,o.html,	m.html,n.html,o.html,
n.html,o.html, ⊗ o.html,p.html,	n.html,o.html,p.html,
o.html,p.html, ⊗ p.html,q.html,	o.html,p.html,q.html,
x1.html,c.html, ⊗ c.html,d.html,	x1.html,c.html,d.html,
x4.html,x5.html, ⊗ x5.html,d.html,	x4.html,x5.html,d.html,
x5.html,d.html, ⊗ d.html,e.html,	x5.html,d.html,e.html,

หลังจากนั้น จะทำการสไลด์ทรานเซ็กชั่นล็อกไฟล์ โดยการเพิ่มขนาดของ Windows = 2 โดย Pattern ที่ได้จากการสไลด์นั้น จะนำไปเปรียบเทียบกับ Pattern ที่ได้จากการทำกระบวนการ Join Operation ซึ่งหากว่า Pattern ที่ได้จากการสไลด์นั้นไม่เหมือนกับใน C_2 ดังนั้น Pattern ที่สไลด์มานั้นจะไม่มีการบันทึกผลนั้นเอง ซึ่งกระบวนการดังกล่าวนี้จะทำช่วยลดการทำงานของระบบในการประมวลผลได้เป็นอย่างดี หากสังเหตุได้จากการผลการทำงานก่อนหน้า จะเห็นว่า ยิ่ง ค่า k มากขึ้น นั้นหมายถึงจะอัลกอริธึมจะทำงานได้เร็วขึ้นนั้นเอง ดังตารางที่ 3.6 แสดงผลการสไลด์ของทรานเซ็กชั่นล็อกไฟล์

ตารางที่ 3.7 แสดงข้อมูลที่ได้จากการสไลด์ที่ Windows = 3

ลำดับ	จำนวนครั้งที่เกิด	Pattern
1	1	a.html,b.html,c.html,
2	1	b.html,c.html,y.html,
3	1	c.html,y.html,z.html,
4	3	m.html,n.html,o.html,
5	3	n.html,o.html,p.html,
6	2	o.html,p.html,q.html,
7	2	x1.html,c.html,d.html,
8	1	c.html,d.html,o.html,

ตารางที่ 3.7 (ต่อ)

ลำดับ	จำนวนครั้งที่เกิด	Pattern
9	1	d.html,o.html,p.html,
10	1	c.html,d.html,e.html,
11	1	x.html,y.html,z.html,
12	2	x4.html,x5.html,d.html,
13	2	x5.html,d.html,e.html,
14	2	d.html,e.html,y.html,

6. ขั้นตอนที่ (6) เป็นการหา Large Sequence L_2: ซึ่งกล่าวได้ว่าสมาชิกของ C_2 ที่ผ่านค่าสนับสนุนน้อยที่สุด (Minimum Support) แล้วนั้นเอง อีกทั้งในขั้นตอนนี้ยังมีการคำนวนหาค่า Confidence อีกด้วย โดยจะได้ดังตารางที่ 3.7

ตารางที่ 3.8 แสดงข้อมูลของ Large Sequence (L_2)

ลำดับ	Support	Confidence	Pattern
1	0.22222	0.66667	d.html,e.html,y.html,
2	0.33333	1	m.html,n.html,o.html,
3	0.33333	1	n.html,o.html,p.html,
4	0.22222	0.5	o.html,p.html,q.html
5	0.22222	1	x1.html,c.html,d.html,
6	0.22222	1	x4.html,x5.html,d.html,
7	0.22222	1	x5.html,d.html,e.html,

ตารางที่ 3.9 แสดงการ Join Operation $L_2 \otimes L_2 = C_3$

Join Operation ($L_2 \otimes L_2$)	Pattern (C_3)
m.html,n.html,o.html, ⊗ n.html,o.html,p.html,	m.html,n.html,o.html,p.html,
n.html,o.html,p.html, ⊗ o.html,p.html,q.html,	n.html,o.html,p.html,q.html,
x4.html,x5.html,d.html, ⊗ x5.html,d.html,e.html,	x4.html,x5.html,d.html,e.html,
x5.html,d.html,e.html, ⊗ d.html,e.html,y.html,	x5.html,d.html,e.html,y.html,

หลังจากนั้น จะทำการสไลด์ทรานเซ็กชั่นล็อกไฟล์ โดยการเพิ่มขนาดของ Windows = 4 โดย Pattern ที่ได้จากการสไลด์นั้น จะนำไปเปรียบเทียบกับ Pattern ที่ได้จากการทำกระบวนการ Join Operation ซึ่งหากว่า Pattern ที่ได้จากการสไลด์นั้นไม่เหมือนกับใน C3 ดังนั้น Pattern ที่สไลด์มานั้นจะไม่มีการบันทึกผลนั้นเอง

ดังตารางที่ 3.6 แสดงผลการสไลด์ของทรานเซ็กชั่นล็อกไฟล์

ตารางที่ 3.10 แสดงข้อมูลที่ได้จากการสไลด์ที่ Windows = 4

ลำดับ	จำนวนครั้งที่เกิด	Pattern
1	1	a.html,b.html,c.html,y.html,
2	1	b.html,c.html,y.html,z.html,
3	3	m.html,n.html,o.html,p.html,
4	2	n.html,o.html,p.html,q.html,
5	1	x1.html,c.html,d.html,o.html,
6	1	c.html,d.html,o.html,p.html,
7	1	x1.html,c.html,d.html,e.html,
8	2	x4.html,x5.html,d.html,e.html,
9	2	x5.html,d.html,e.html,y.html,

ตารางที่ 3.11 แสดงข้อมูลของ Large Sequence (L_3)

ลำดับ	Support	Confidence	Pattern
1	0.33333	1	m.html,n.html,o.html,p.html,
2	0.22222	0.66667	n.html,o.html,p.html,q.html,
3	0.22222	1	x4.html,x5.html,d.html,e.html,
4	0.22222	1	x5.html,d.html,e.html,y.html,

7. ขั้นตอนที่ (7) ซึ่งจะต้องมีการสร้าง Candidate: C_4 จาก Large Sequence $L_3 \otimes L_3$ ซึ่งจะแสดงผลลัพธ์การกระบวนการ Join Operation โดยจะสังเกตุเห็นว่า ขนาดความยาวของความสัมพันธ์ที่ได้นั้นจะมีขนาดที่ยาวมากขึ้น นั้นเองดังตารางที่ 3.11

ตารางที่ 3.12 แสดงการ Join Operation $L_3 \otimes L_3 = C_4$

Join Operation ($L_3 \otimes L_3$)	Pattern (C_4)
m.html,n.html,o.html,p.html, ⊗ n.html,o.html,p.html,q.html, x4.html,x5.html,d.html,e.html, ⊗ x5.html,d.html,e.html,y.html,	m.html,n.html,o.html,p.html,q.html, x4.html,x5.html,d.html,e.html,y.html,

หลังจากนั้น จะทำการสไลด์ทรานเซ็กชั่นล็อกไฟล์ โดยการเพิ่มขนาดของ Windows = 4 โดย Pattern ที่ได้จากการสไลด์นั้น จะนำไปเปรียบเทียบกับ Pattern ที่ได้จากการทำกระบวนการ Join Operation ซึ่งหากว่า Pattern ที่ได้จากการสไลด์นั้นไม่เหมือนกับใน C_4 ดังนั้น Pattern ที่สไลด์มานั้นจะไม่มีการบันทึกผลนั้นเอง ดังตารางที่ 3.13

ตารางที่ 3.13 แสดงข้อมูลที่ได้จากการสไลด์ที่ Windows = 5

ลำดับ	จำนวนครั้งที่เกิด	Pattern
1	1	a.html,b.html,c.html,y.html,z.html,
2	2	m.html,n.html,o.html,p.html,q.html,
3	1	x1.html,c.html,d.html,o.html,p.html,
4	2	x4.html,x5.html,d.html,e.html,y.html,

ตารางที่ 3.14 แสดงข้อมูลของ Large Sequence (L_4)

ลำดับ	Support	Confidence	Pattern
1	0.22222	0.66667	m.html,n.html,o.html,p.html,q.html,
2	0.22222	1	x4.html,x5.html,d.html,e.html,y.html,

จะเห็นว่า Large Sequence (L_4) นั้นไม่สามารถทำการ Join Operation เมื่อไม่มีการสร้าง Candidate ได้ ซึ่งจะเป็นการสิ้นสุดการทำงานของอัลกอริธึมใหม่ในการค้นหารูปแบบลำดับของเว็บไซต์ นั้นเอง ในขั้นตอนต่อไปจะเป็นการนำกฎที่ผ่านค่า Minimum Support ไปสร้างต้นไม้ เพื่อวัตถุประสงค์ในการลดทอนกฎที่ไม่จำเป็นต่อการจัดเก็บทิ้งออกไป จะเหลือกฎทั้งหมดที่จะนำไปใช้สร้างต้นไม้

ตารางที่ 3.15 แสดงกฎที่จะนำไปใช้ในการสร้างต้นไม้

ลำดับ	Confidence	Pattern
1	0.667	c.html,d.html,
2	0.750	d.html,e.html,
3	0.667	e.html,y.html,
4	1.000	m.html,n.html,
5	1.000	n.html,o.html,
6	1.000	o.html,p.html,
7	0.500	p.html,q.html,
8	1.000	x1.html,c.html,
9	1.000	x4.html,x5.html,
10	1.000	x5.html,d.html,
11	0.500	y.html,z.html,
12	0.667	d.html,e.html,y.html,
13	1.000	m.html,n.html,o.html,
14	1.000	n.html,o.html,p.html,
15	0.500	o.html,p.html,q.html,
16	1.000	x1.html,c.html,d.html,
17	1.000	x4.html,x5.html,d.html,
18	1.000	x5.html,d.html,e.html,
19	1.000	m.html,n.html,o.html,p.html,
20	0.667	n.html,o.html,p.html,q.html,
21	1.000	x4.html,x5.html,d.html,e.html,
22	1.000	x5.html,d.html,e.html,y.html,
23	0.667	m.html,n.html,o.html,p.html,q.html,
24	1.000	x4.html,x5.html,d.html,e.html,y.html,

3.2 การคัดกฎความสัมพันธ์ทิ้ง

เรามีความจำเป็นที่จะต้องเก็บกฎที่มีประโยชน์ไว้ โดยเฉพาะอย่างยิ่ง จะพิจารณากฎความสัมพันธ์ที่มีความน่าเชื่อถือด้วย ดั้งนั้นจึงนำกฎนั้นมาสร้างเป็นต้นไม้กฎความสัมพันธ์ 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

ในขั้นตอนต่อไปเราจะนำกฎต่างๆที่ผ่านค่า Minimum Support มาสร้างเป็นลักษณะต้นไม้ (Tree) เพื่อการพรุนนิ่ง โดยจะกล่าวถึงหลักการสร้างดังต่อไปนี้

แต่ละ กฎ เปรียบได้ดัง โหนดหนึ่งโหนด

- โหนดตัวบนแสดงเป็นโหนดแม่ (parent node) ซึ่งโหนดตัวล่างจะเป็นโหนดลูก (children node)
- โหนด (Node) 1 โหนดคือกฎความสัมพันธ์ 1 กฎ
- รูทของ tree จะเป็น default rule ซึ่ง default rule โดยปกติจะเป็นเว็บเพจที่มีการความถี่ในการเรียกใช้งานมากที่สุด หรือเว็บเพจหน้าหลัก Home page

ดังจะเห็นว่า มีการกำหนดค่า Confidence ของ Root ด้วย และในระดับเดียวกันจะเป็นลักษณะโหนด Windows = 2 ซึ่งลูกของโหนดดังกล่าวนั้นจะเป็น Windows = 3 และเรียงลำดับต่อไป จะเห็นได้ว่า การสร้างโหนดนั้นมีความสำคัญอย่างยิ่ง โดยหลักการในการสร้างโหนดนั้น เพจทำนายของแม่จะต้องเหมือนกันตัวลูก

ดังรูปจะเป็นการนำกฎที่เหลือจากการพรุนนำมาสร้างต้นไม้ โดยใช้หลักการที่ได้กล่าวมาแล้วข้างต้น ซึ่งลักษณะของต้นไม้จะเป็นต้นไม้ที่ไม่จำกัดกิ่ง ดังแสดงในรูปที่ 3.3

3.2.2 วิธีคัดกฎความสัมพันธ์ทิ้ง

หลักการในการคัดกฎความสัมพันธ์ทิ้งจากต้นไม้กฎความสัมพันธ์ที่ได้นั้นจะใช้วิธีพิจารณาจากบนลงล่าง โดยพิจารณาจาก Root จนถึงโหนดสุดท้าย (Left node) ดังหลักการดังนี้

1. โหนดแม่มีค่าความเชื่อมั่นมากกว่าหรือเท่ากับค่า Limit confidence ให้เก็บกฎจากโหนดแม่ไว้และเปรียบเทียบต่อไป

2. เมื่อโหนดลูกมีการทำนายเหมือนกับโหนดแม่ ก็จะตัดโหนดลูกทิ้งเช่นกัน
3. โหนดแม่มีค่าความเชื่อมั่นน้อยกว่าค่า Limit confidence ให้พิจารณาโหนดลูกทุก ๆ โหนดของโหนดแม่ ถ้ามีค่าความเชื่อมั่นมากกว่าหรือเท่ากับค่า Limit confidence ให้เก็บกฎนั้นไว้และตัดโหนดลูกของโหนดนั้นทิ้ง แต่หากค่าความเชื่อมั่นของกฎนั้นน้อยกว่าค่า Limit confidence ก็จะต้องดูว่าโหนดนั้นมีลูกหรือไม่ แล้วพิจารณาต่อไปว่ามีค่าความเชื่อมั่นมากกว่า หรือน้อยกว่าค่า Limit confidence

แสดงตัวอย่างการคัดกฎความสัมพันธ์ทิ้งดังรูปที่ 3.3 โดยกำหนดให้ค่า Limit confidence มีค่า 0.250

กระบวนการตัดกฎทิ้ง

พิจารณา c => d ซึ่งมีค่า conf = 0.667

มีค่า conf มากกว่า root ดังนั้นจึงมีการเก็บกฎดังกล่าวนี้ไว้

พิจารณา x1,c => d มีค่า conf = 1.00

เมื่อตรวจสอบกับ node แม่ ปรากฏว่ามีค่า conf มากกว่าก็จริงแต่ว่าเนื่องจากโหนดลูกมีการทำนายเหมือนกับโหนดแม่ ดังนั้นจึงจำเป็นต้องตัดโหนดลูกทิ้งไปนั้นเอง

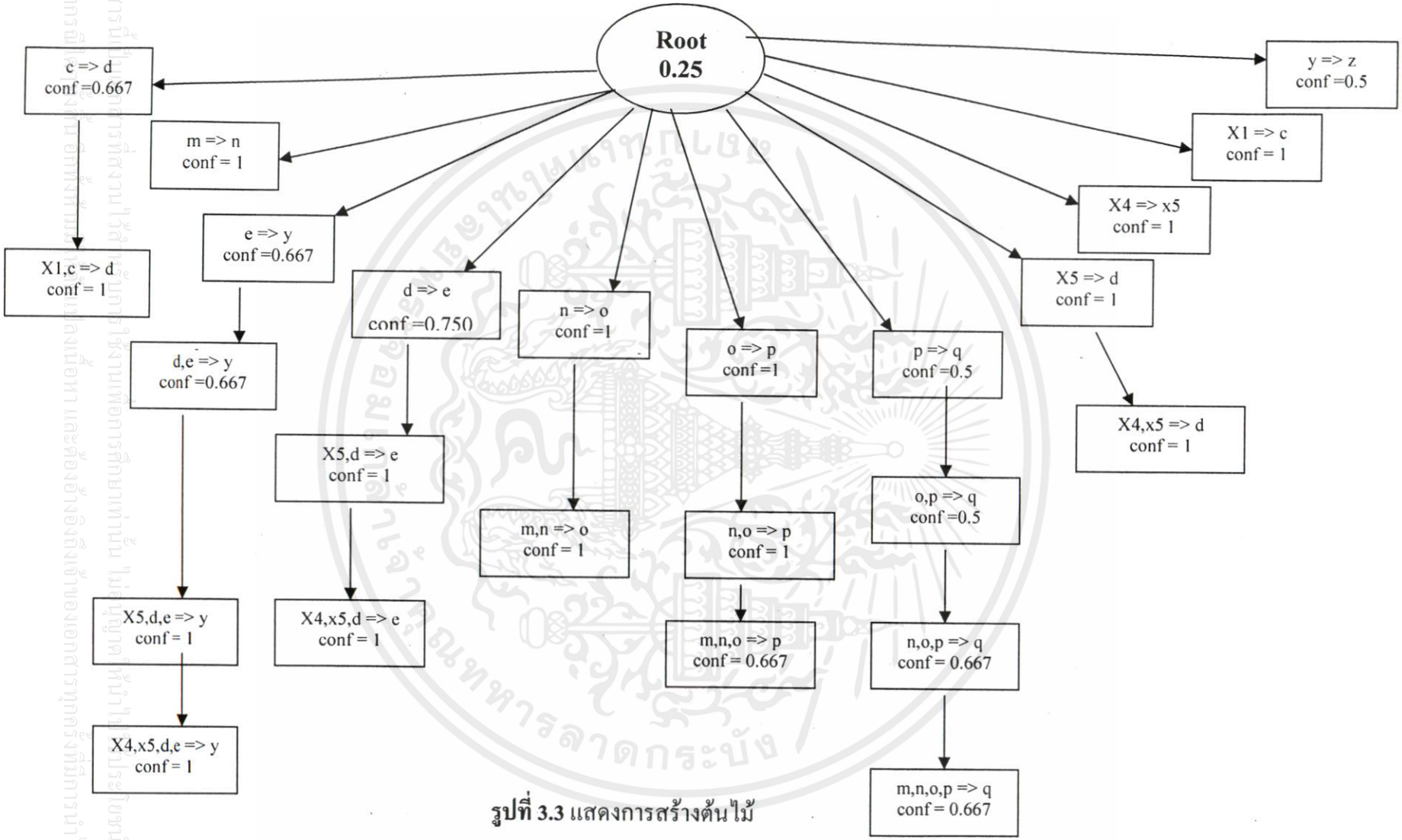
พิจารณา m => n ซึ่งมีค่า conf = 1.00

หากพิจารณาแล้วจะเห็นว่า conf มากกว่า default root ดังนั้นจึงเก็บโหนดดังกล่าวนี้ไว้ โดยเฉพาะอย่างยิ่ง โหนดดังกล่าวไม่มีโหนดลูกอีกด้วย

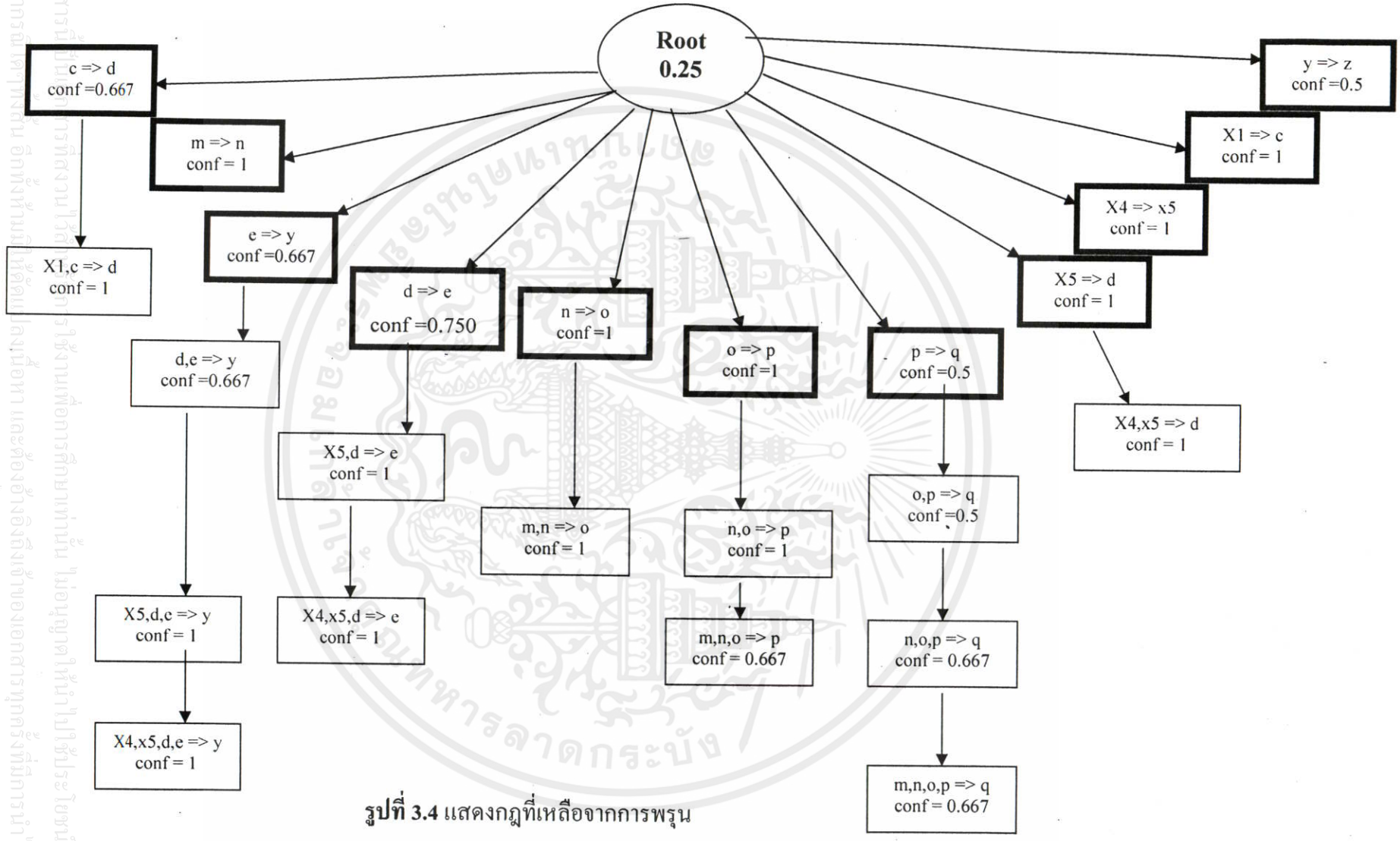
หลักในการพิจารณาโหนดอื่นๆ ก็เช่นเดียวกัน จะพิจารณาโหนดแม่ก่อนแล้วค่อยพิจารณาเหนดลูกต่อไป ซึ่งวิธีการดังกล่าวนี้จะทำให้การตัดกฎที่ไม่จำเป็นออกไปนั้นเอง ดังนั้นผลสุดท้ายก็จะได้กฎที่เหลือจากการพรุ่นนิ่งดังรูปที่ 3.4 คือ ซึ่งประกอบด้วยจำนวนกฎ 11 กฎ ด้วยกัน จะสังเกตุเห็นว่า กฎที่เหลือนั้นจะมีจำนวนเพจที่เคยเยี่ยมชมมาแล้ว (LHS) เพียงเว็บเพจเดียวเท่านั้น โดยลักษณะของกฎดังกล่าวนี้จะเป็นเรียกว่า General Rule หมายถึง สามารถนำไปใช้ประโยชน์ได้ในการทำนายได้หลากหลายกว่านั้นเอง

ในรูปที่ 3.5 จะมีการแสดงตัวอย่างเมื่อมีการปรับค่า Root confidence = 0.55 ซึ่งการปรับเพิ่มค่าดังกล่าวนี้หมายความว่า เราต้องการกรั่นกรองเอาเฉพาะกฎที่มีความเชื่อมั่นเกิน 55% เท่านั้น ซึ่งกฎที่ผ่านการพรุ่งนิ่ง จะเหลือเพียง 9 กฎ เท่านั้น ซึ่งผลที่ได้จะแสดงในตารางที่ 3.15

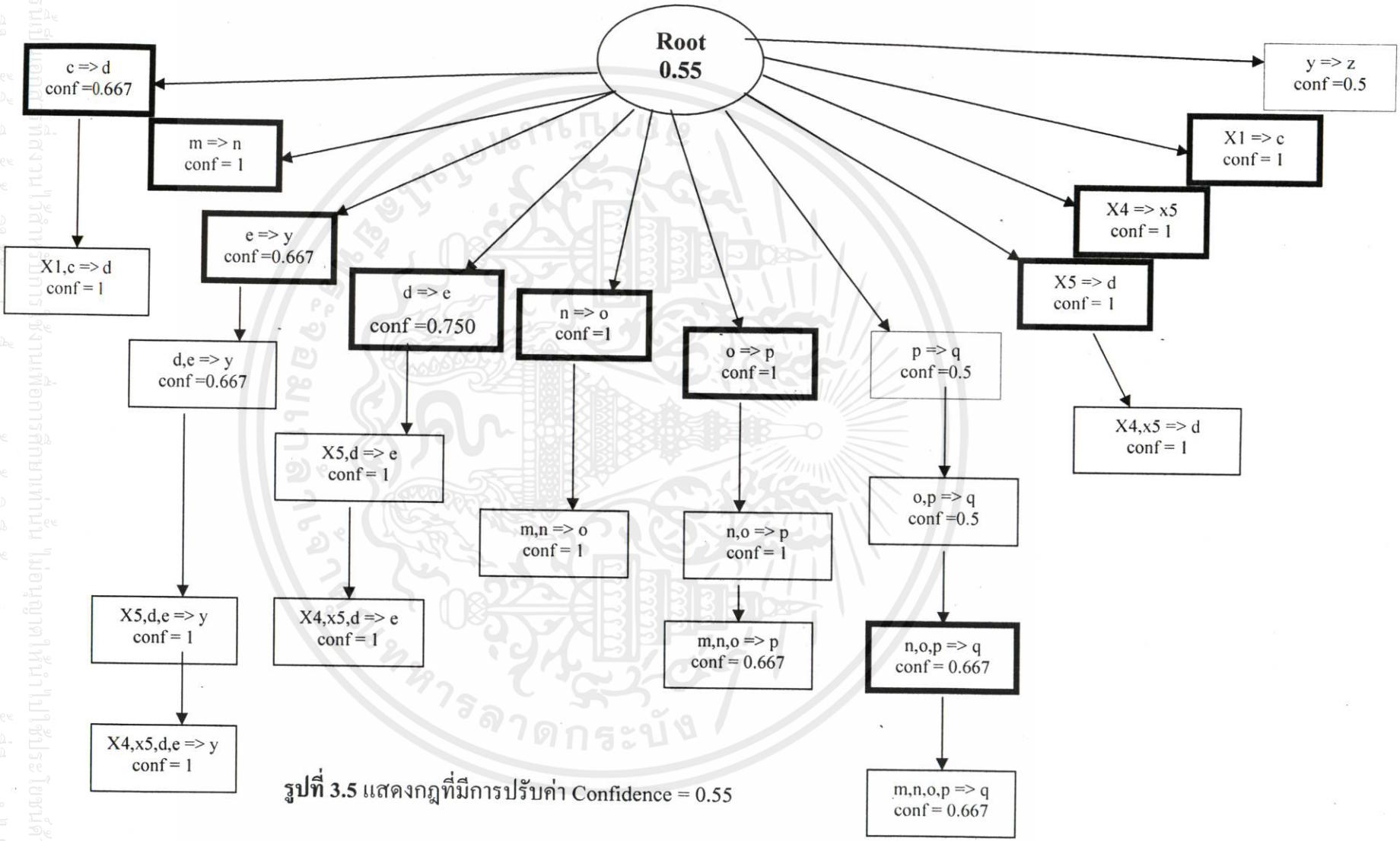
สิ่งที่สังเหตุพบว่า หากเรายิ่งปรับค่า Confidence มากขึ้นเท่าไหร จะส่งผลให้กฎที่เหลือจากการพรุ่นนิ่งนั้นมีความเชื่อมั่นมากขึ้นด้วย ดังรูปที่ 3.5 ผลลัพธ์ที่ได้แสดงดังตารางที่ 3.16 จะเห็นว่า n, o, p ➔ q ซึ่งหากนำไปใช้งานจะให้ความแม่นยำที่สูงกว่า p ➔ q เหตุผลเพราะว่า ในการนำกฎดังกล่าวไปใช้ในการทำนาย เว็บเพจที่เคยเยี่ยมชมมาแล้วจะเป็นปัจจัยสำคัญที่ส่งผลต่อการทำนายด้วย หากเรามีการเรียนรู้เว็บเพจที่ผู้เรียกใช้มีการเยี่ยมชมมาในอดีตหรือที่เยี่ยมชมมาแล้ว มากทำไหร จะทำให้การนายมีความแม่นยำสูงเห็น นั้นเอง



รูปที่ 3.3 แสดงการสร้างต้นไม้



รูปที่ 3.4 แสดงกฎที่เหลือจากการพรุน



รูปที่ 3.5 แสดงกฎที่มีการปรับค่า Confidence = 0.55

ตารางที่ 3.16 กฎที่ผ่านการพรุ่น เมื่อ conf = 0.25

Confidence	Rule
0.66667	c.html,d.html,
0.75	d.html,e.html,
0.66667	e.html,y.html,
1	m.html,n.html,
1	n.html,o.html,
1	o.html,p.html,
0.5	p.html,q.html,
1	x1.html,c.html,
1	x4.html,x5.html,
1	x5.html,d.html,
0.5	y.html,z.html,

ตารางที่ 3.17 กฎที่ผ่านการพรุ่น เมื่อ conf = 0.55

Confidence	Rule
0.66667	c.html,d.html,
0.75	d.html,e.html,
0.66667	e.html,y.html,
1	m.html,n.html,
1	n.html,o.html,
1	o.html,p.html,
1	x1.html,c.html,
1	x4.html,x5.html,
1	x5.html,d.html,
0.667	n.html,o.html,p.html,q.html,

```
(P: user's current clicking sequence; n: minimal path length)
Begin
    For i:=|P| down n do
        If P is an index in hash table Hi Then
            Prediction:=Hi[P];
            Return (Prediction);
        Eng If
        P : = the same sequence with the first element removed;
    End For
    Return ("No Prediction");
End
```

รูปที่ 3.6 แสดงอัลกอริธึมในการทำนายเว็บเพจ [7]

ในการทำนายนั้นจะอ่านข้อมูลจาก ล็อกไฟล์เซอร์เวอร์ทีละเรคคอร์ด
จะทำการลดขนาดของความยาวเพจ จนเหลือเพียงหนึ่งเพจเท่านั้น
และนำเพจที่ลดขนาด ไปทำการเปรียบเทียบกับตารางกฎที่ผ่านการพรุนแล้ว

ตารางที่ 3.17 แสดงล็อกทรานเซ็คชั่น

A.HTML,B.HTML,C.HTML,
O.HTML,P.HTML,Q.HTML,R.HTML,
X.HTML,Y.HTML,Z.HTML,
M.HTML,N.HTML,O.HTML,

ตารางที่ 3.18 แสดงกฎที่ผ่านการพรุ่นแล้ว

B.HTML → C.HTML conf = 0.5
Q.HTML → R.HTML conf = 0.4
Q.HTML → S.HTML conf = 0.3
Y.HTML → Z.HTML conf = 0.5
Y.HTML → W.HTML conf = 0.7
Default Root = O.HTML

ข้อมูลจากตารางที่ 3.17 ซึ่งเป็นข้อมูลที่บันทึกพฤติกรรมของผู้เข้าชมเว็บไซต์ โดยข้อมูลชุดดังกล่าวนี้เมื่อนำมาใช้งาน โดยข้อมูลในตารางที่ 3.18 เป็นกฎที่ผ่านการพรุ่นนิ่งเรียบร้อยแล้ว พร้อมใช้งานในการทำนาย โดยข้อมูลแต่ละกฎจะมีค่าความเชื่อมั่นรวมอยู่ด้วย จะสามารถอธิบายการทำงานของ อัลกอริธึมได้ดังต่อไปนี้

เร็คคอร์ดที่ 1 A.HTML, B.HTML ➔ C.HTML

Pattern Prediction = C.HTML

จากการเปรียบเทียบกับกฎที่พรุ่นแล้ว จะได้ เพจในการทำนายคือ C.HTML

เมื่อเปรียบเทียบ Pattern Prediction = Prune Prediction = C.HTML

ดังนั้นจึงเป็นการทำนายที่ถูกต้อง

เร็คคอร์ดที่ 2 O.HTML,P.HTML,Q.HTML, ➔ R.HTML

Pattern Prediction = R.HTML

Step 1: เปรียบเทียบ O.HTML,P.HTML,Q.HTML กับ ตารางกฎที่พรุ่นแล้ว "NO"

Step 2: เปรียบเทียบ P.HTML,Q.HTML, กับ ตารางกฎที่พรุ่นแล้ว "NO"

Step 3: เปรียบเทียบ Q.HTML, กับตารางกฎที่พรุ่นแล้ว "YES"

Q.HTML ➔ R.HTML conf = 0.4

Q.HTML ➔ S.HTML conf = 0.3

ซึ่งจำเป็นจะต้องเลือกอย่างใดอย่างหนึ่ง โดยการเปรียบเทียบค่า Confidence ดังจะเห็นว่าจำเป็นต้องเลือก Q.HTML ➔ R.HTML

เมื่อเปรียบเทียบ Pattern Prediction = Prune Prediction = R.HTML

ดังนั้นจึงเป็นการทำนายที่ถูกต้อง

เร็คคอร์ดที่ 3 X.HTML,Y.HTML, ➔ Z.HTML,

Pattern Prediction = Z.HTML

Step 1: เปรียบเทียบ X.HTML,Y.HTML กับตารางกฎที่พรุ่นแล้ว "NO"

Step 2: เปรียบเทียบ Y.HTML กับตารางกฎที่พรุ่นแล้ว "YES"

Y.HTML ➔ Z.HTML conf = 0.5

Y.HTML ➔ W.HTML conf = 0.7

ซึ่งจำเป็นจะต้องเลือกอย่างใดอย่างหนึ่ง โดยการเปรียบเทียบค่า Confidence ดังจะเห็นว่าจำเป็นต้องเลือก Y.HTML ➔ W.HTML

เมื่อเปรียบเทียบ Pattern Prediction = Prune Prediction = Z.HTML

ดังนั้นจึงเป็นการทำนายผิดพลาด

เร็คคอร์ดที่ 4 M.HTML, N.HTML, ➔ O.HTML,

Pattern Prediction = O.HTML

Step 1: เปรียบเทียบ M.HTML, N.HTML, กับตารางกฎที่พรุ่นแล้ว "NO"

Step 2: เปรียบเทียบ N.HTML, กับตารางกฎที่พรุ่นแล้ว "NO"

ดังจะเห็นว่าตัวที่ใช้เปรียบเทียบนั้นหมดแล้ว ไม่พบกับกฎในตารางกฎที่พรุ่นด้วย

ดังนั้นจึงมีการให้ค่า Prune Prediction = Default Root = O.HTML

เมื่อเปรียบเทียบ Pattern Prediction = Prune Prediction = O.HTML

โดยในการคิดคำนวณความแม่นยำ [7] นั้นสามารถหาได้จากสูตร ดังนี้
คืออัตราส่วนของจำนวน Pattern ที่ทำนายถูกต้อง ต่อจำนวน Pattern ทั้งหมด

$$Precession = Pr/Pp$$

Precession = ความแม่นยำ

Pr = จำนวนรูปแบบการเข้าชม (Pattern) ทั้งหมดที่มีการทำนายถูกต้อง

Pp = จำนวนรูปแบบการเข้าชม (Pattern) ทั้งหมดของทรานเซ็กชั่นล็อก

ข้อมูลจากตาราง 3.17 ซึ่งเป็นทรานเซ็กชั่น การเข้าชมเว็บไซต์ ประกอบด้วยจำนวนรูปแบบการเข้าชม ทั้งหมด 4 รูปแบบการเข้าชม เมื่อต้องการหาความแม่นยำ โดยนำกฎจากตารางที่ 3.18 มาทดสอบ จะทราบว่า จากการทำนาย จะสามารถทำนายได้ถูกต้อง จำนวน 3 รูปแบบการเข้าชม ดังนี้คือ

A.HTML, B.HTML ➔ C.HTML

O.HTML,P.HTML,Q.HTML ➔ R.HTML

M.HTML,N.HTML ➔ O.HTML

ดังนั้นเมื่อหาค่าความแม่นยำ จะได้

$$Precession = (3/4)$$

ความแม่นยำ Precession = 0.75

บทที่ 4

การทดลอง และผลการทดลอง

ในบทนี้จะกล่าวถึงชุดข้อมูลที่ใช้ในการทดลอง การทดลอง และผลการทดลอง ในการสร้าง Model ต่าง ๆ ในงานวิจัยนี้ใช้โปรแกรม MATLAB 6.5 ในการทดลองทั้งหมด และใช้โปรแกรม MATLAB 6.5, Edit Plus ในการเตรียมข้อมูลก่อนการทดลอง

4.1 ชุดข้อมูลที่ใช้ในการทดลอง

ในส่วนนี้จะกล่าวถึงชุดข้อมูลที่ใช้ในการทดลอง โดยแบ่งออก 2 ส่วนคือ ลักษณะของชุดข้อมูลที่ใช้ในสร้างกฎหรือ Training Data Source และข้อมูลในการทดสอบ Testing Data Source

4.1.1 ชุดข้อมูล ล็อกไฟล์

Input Pattern	
จำนวน	Pattern
100	1_A.html,1_B.html,1_C.html,1_D.html,
100	2_A.html,2_B.html,2_C.html,2_D.html,
100	3_A.html,3_B.html,3_C.html,3_D.html,
400	4_A.html,4_B.html,4_C.html,4_D.html,4_E.html,
200	5_A.html,5_B.html,5_C.html,5_D.html,5_E.html,
200	6_A.html,6_B.html,6_C.html,6_D.html,6_E.html,
200	7_A.html,7_B.html,7_C.html,7_D.html,7_E.html,
200	8_A.html,8_B.html,8_C.html,8_D.html,8_E.html,
200	9_A.html,9_B.html,9_C.html,9_D.html,9_E.html,
200	10_A.html,10_B.html,10_C.html,

รูปที่ 4.1 แสดงข้อมูลส่วน Input Pattern

การเตรียมการข้อมูล ประกอบด้วยขั้นตอนต่างๆดังนี้

- ทำการสร้างชุดข้อมูล ขึ้นมาจำนวน 2000 ทรานเซ็คชั่น
- ขนาดของแต่ละทรานเซ็คชั่นจะประกอบด้วยการเยี่ยมชมจำนวน 7 เพจ
- เมื่อสร้างข้อมูลเสร็จเรียบร้อยแล้วจะทำการสลับตำแหน่งของแต่ละทรานเซ็คชั่น

8k.html,11p.html,8_a.html,8_b.html,8_c.html,8_d.html,8_e.html,
20s.html,16o.html,15b.html,1_a.html,1_b.html,1_c.html,1_d.html,
20d.html,13v.html,16q.html,6j.html,10_a.html,10_b.html,10_c.html,
12w.html,9v.html,8_a.html,8_b.html,8_c.html,8_d.html,8_e.html,
3i.html,6z.html,9_a.html,9_b.html,9_c.html,9_d.html,9_e.html,
4i.html,10s.html,19x.html,2_a.html,2_b.html,2_c.html,2_d.html,
8y.html,16x.html,4_a.html,4_b.html,4_c.html,4_d.html,4_e.html,
8i.html,6r.html,14m.html,3_a.html,3_b.html,3_c.html,3_d.html,
4p.html,7b.html,15j.html,2_a.html,2_b.html,2_c.html,2_d.html,
16l.html,8a.html,6_a.html,6_b.html,6_c.html,6_d.html,6_e.html,
12o.html17a.html,10j.html,3u.html,1g.html,13e.html,9d.html,
11d.html,16u.html,7_a.html,7_b.html,7_c.html,7_d.html,7_e.html,
8e.html,14n.html,13g.html,15y.html,10_a.html,10_b.html,10_c.html,
9v.html,15d.html,9_a.html,9_b.html,9_c.html,9_d.html,9_e.html,
13z.html,20a.html,17f.html,1_a.html,1_b.html,1_c.html,1_d.html,
14a.html,11f.html,6_a.html,6_b.html,6_c.html,6_d.html,6_e.html,
18c.html,12x.html,2a.html,8u.html,10_a.html,10_b.html,10_c.html,
16o.html,4w.html,5_a.html,5_b.html,5_c.html,5_d.html,5_e.html,
14u.html,8a.html,7_a.html,7_b.html,7_c.html,7_d.html,7_e.html,
6j.html17r.html,2a.html,11k.html,9h.html,10z.html,3s.html,

รูปที่ 4.2 แสดงตัวอย่างข้อมูลที่ได้จากการสร้างชุดข้อมูลล็อกไฟล์

เมื่อได้ข้อมูลล็อกไฟล์ จำนวนทั้งสิ้น 2000 ทรานเซ็คชั่นแล้วในการบวนการต่อไป จะนำชุดข้อมูลเหล่านี้ไปผ่านกระบวนการของอัลกอริธึมใหม่สำหรับการค้นหารูปแบบลำดับของเว็บไซต์ ซึ่งในการประมวลผลอัลกอริธึมใหม่ดังกล่าวนี้ จำเป็นอย่างยิ่งที่จะต้องมีการทดลองนำผลลัพธ์ ไปตรวจสอบความแม่นยำของการทำงานของอัลกอริธึมนั้นเอง ดังนั้นโดยขั้นตอนดังกล่าวนี้ จำเป็นจะต้องมีการแยกข้อมูลออกเป็นสองส่วน ด้วยกัน ดังนี้คือ

- ข้อมูลส่วนที่ 1 ซึ่งคือ Data Training จำนวน 1000 ทรานเซ็คชั่น เพื่อใช้ในการประมวลผลของอัลกอริธึม โดยผลลัพธ์จะนำไปทดสอบความแม่นยำกับข้อมูลชุดที่ 2
- ข้อมูลส่วนที่ 2 คือ Data Testing จำนวน 1000 ทรานเซ็คชั่น เพื่อใช้ในการทดสอบกับกฎที่ผ่านการประมวลผลของอัลกอริธึม โดยผลลัพธ์ของกระบวนการนี้จะให้ค่าเป็นความแม่นยำในการทำนาย ซึ่งเป็นการทดสอบการทำงานของอัลกอริธึม

ตารางที่ 4.1 แสดงการบันทึกผลการทดลอง ในการปรับค่าพารามิเตอร์ Support Factor

Support	กฎที่บันทึกทั้งหมด	กฎที่ผ่านการ Prune	มี Pattern		ไม่มี Pattern		ความแม่นยำ (1)	ความแม่นยำ (2)
			ถูก	ผิด	ถูก	ผิด		
0.01	73	35	955	-	-	45	96%	100%
0.05	67	32	904	-	-	96	90%	100%
0.09	62	26	807	-	-	193	80%	100%
0.1	13	6	287	-	-	713	28%	100%
0.2	9	4	196	-	-	804	20%	100%

วิธีคิดคำนวณหาค่าความแม่นยำ

ความแม่นยำ (1) สามารถคำนวณได้ดังตัวอย่าง

กรณีคิดที่ค่า Support = 0.01

$$\frac{\text{ข้อมูลที่ทำนายถูกต้อง}}{\text{ข้อมูลทรานเซ้กชั่นทั้งหมด}}$$

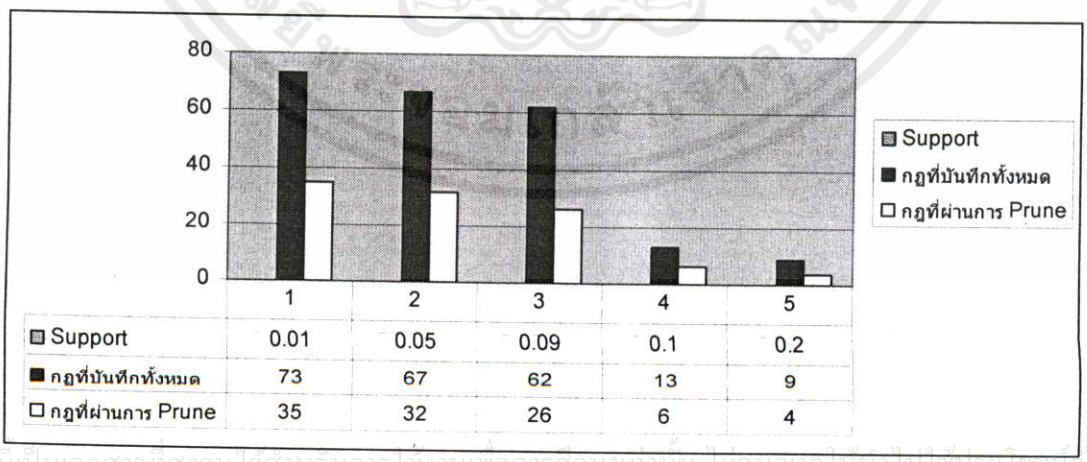
$$\frac{955}{1000}=96\%$$

ความแม่นยำ (2) สามารถคำนวณได้จากใน กรณีที่ทำนายเฉพาะที่มีกฎเท่านั้น ส่วน Pattern ไหนที่ไม่มีในกฎก็จะไม่นำมาทำนายและไม่นำ Pattern ดังกล่าวมาคิดคำนวนหาค่าความแม่นยำ

กรณีคิดที่ค่า Support = 0.2

$$\frac{\text{ข้อมูลที่ทำนายถูกต้อง}}{\text{ข้อมูลทรานเซ้กชั่นทั้งหมด}-\text{ข้อมูลทำนายผิด}}$$

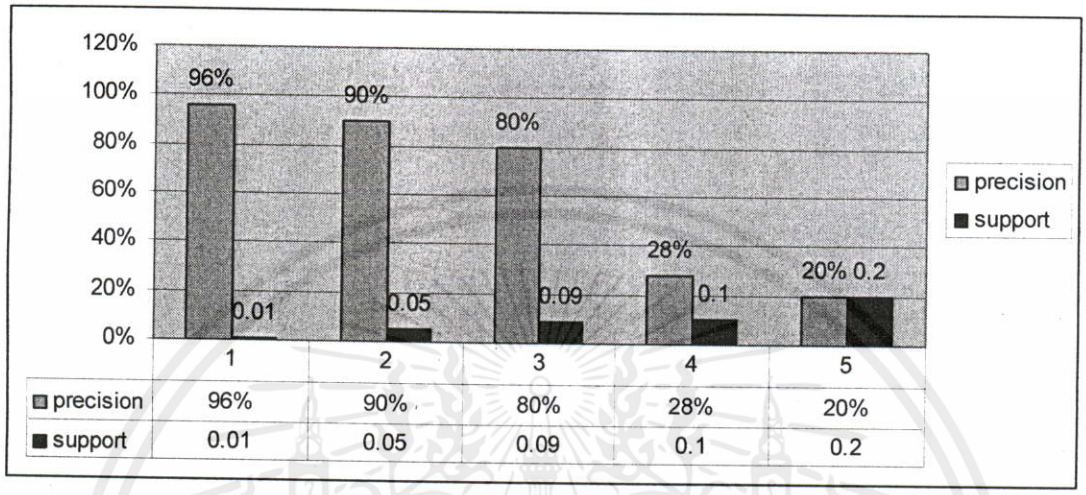
$$\frac{196}{1000-804}=100\%$$



รูปที่ 4.3 แสดงการปรับค่า support ที่มีผลต่อปริมาณกฎที่ค้นพบ

กฎที่พบ หมายถึงกฎดังกล่าวเมื่อเปรียบเทียบกับค่า Minimum Support แล้วกฎดังกล่าวมีค่า Support มากกว่า ค่า Minimum Support นั้นเอง

ในกระบวนการ Prune คือกระบวนการลดทอนกฎ โดยวิธีการสร้าง Tree ซึ่งค่า Root Confidence = 0



	1	2	3	4	5
precision	96%	90%	80%	28%	20%
support	0.01	0.05	0.09	0.1	0.2

รูปที่ 4.4 แสดงความแม่นยำเมื่อมีการปรับค่า Support ในขณะที่ค่า Confidence คงที่

4.1.2 ชุดข้อมูลจาก U.S Environmental Protection Agency (WWW.EPA.GOV)

ทำการสุ่มเพื่อใช้ในการทดลอง 142 patterns แบ่งแยกข้อมูล เป็น สองชุด ส่วน Training 70 pattern และ Testing 72

U.S Environmental Protection Agency (WWW.EPA.GOV)
141.243.1.172,[29:23:53:25],"GET,/Software.html,HTTP/1.0",200,1497
query2.lycos.cs.cmu.edu,[29:23:53:36],"GET,/Consumer.html,HTTP/1.0",200,1325
tanuki.twics.com,[29:23:53:53],"GET,/News.html,HTTP/1.0",200,1014
wpbfl2-45.gate.net,[29:23:54:15],"GET,/,HTTP/1.0",200,4889
wpbfl2-45.gate.net,[29:23:54:16],"GET,/icons/circle_logo_small.gif,HTTP/1.0",200,2624
wpbfl2-45.gate.net,[29:23:54:18],"GET,/logos/small_gopher.gif,HTTP/1.0",200,935
140.112.68.165,[29:23:54:19],"GET,/logos/us-flag.gif,HTTP/1.0",200,2788
wpbfl2-45.gate.net,[29:23:54:19],"GET,/logos/small_ftp.gif,HTTP/1.0",200,124
wpbfl2-45.gate.net,[29:23:54:19],"GET,/icons/book.gif,HTTP/1.0",200,156

รูปที่ 4.5 แสดงตัวอย่างของข้อมูล จาก U.S Environmental Protection Agency

ข้อมูลเมื่อนำไปผ่านกระบวนการ Pre-processing แล้วจะเป็นข้อมูลอินพุทในการประมวลผลอัลกอริธึม

/software.html,/docs/airsexec/overview.html,/docs/rrel/metaepa.txt.html,
/owow/publist/publist1.html,/owow/publist/pubcon.html,
/docs/oppts_harmonized/metadata.html,/docs/oppts_harmonized/abguide.txt.html,
/docs/gcdoar/particip.html,/docs/gcdoar/greenlights.html,
/docs/whatshot.html,/docs/freesfw/description.html,
/enviro/html/cerclis/cerclis_overview.html,/enviro/html/cerclis/cerclis_query.html,/rules.html,
/owow/sec9/index.html,/owow/publist/pubcon.html,/owow/publist/publist1.html,
/docs/whatsnew.html,/docs/whatshot.html,
/owow/publist/publist1.html,/owow/publist/pubcon.html,
/offices.html,/docs/welcome/epa.html,
/docs/whatshot.html,/docs/freesfw/description.html,
/docs/major.html,/enviro/html/ef_home.html,
/docs/whatsnew.html,/rules.html,/docs/epa-air/1995/august/day-29/pr-959.html,
/epahome/progs-init.html,/docs/negp/intro.txt.html,/docs/negp/metadata.asc.html,/epahome/epa.html,/browner/a
/docs/whatsnew.html,/docs/whatshot.html,
/region10/www/r10.html,/region10/www//gisapps.html,/region10/www//gisapps/site.html,/rules.html,/standard
/pic.html,/docs/whatsnew.html,/docs/whatshot.html,
/docs/major.html,/docs/idaho.html,/docs/ngispr/natgispr.html,/docs/welcome/epa.html,/information.html,/pic.ht
/docs/internet.html,/docs/welcome/welcome.html,/docs/welcome/epa.html,
/owow/sec7/index.html,/owow/sec7/page1.html,
/software.html,/docs/airsexec/overview.html,
/info.html,/docs/hrmd/vacancy/metavac.txt.html,/docs/hrmd/vacancy/5-all.txt.html,/docs/environment.html,/ru

รูปที่ 4.6 แสดงข้อมูลเมื่อผ่านกระบวนการแยกแยะทรานเซ็กชั่น

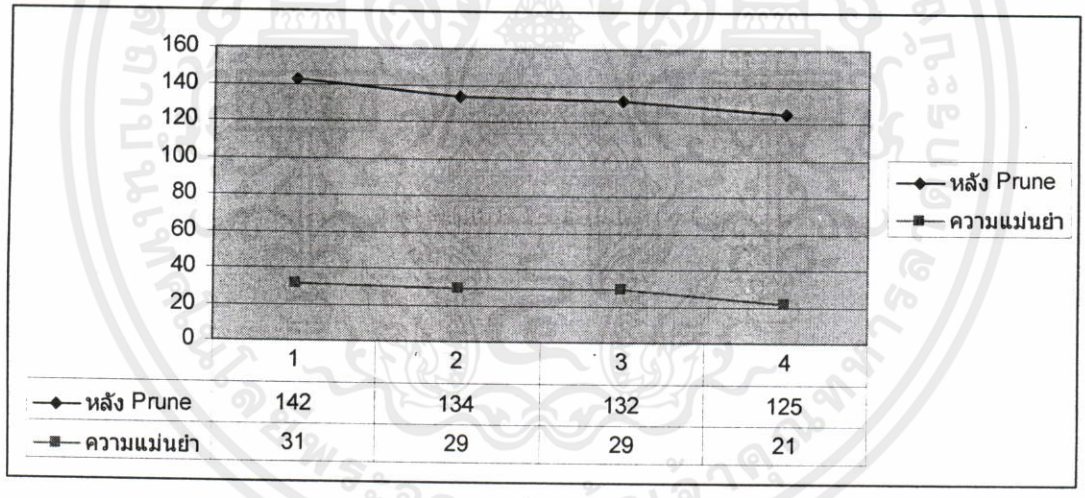
ตารางที่ 4.3 แสดงการเปรียบเทียบค่าความแม่นยำที่ ค่า Support ต่าง และ Confidence = 0

Support	กฎที่บันทึกทั้งหมด	กฎที่ผ่านการ Prune	มี Pattern		ไม่มี Pattern		ความแม่นยำ (1)	ความแม่นยำ (2)
			ถูก	ผิด	ถูก	ผิด		

0.01	569	124	41	30	-	6	57%	63%
0.02	26	26	26	45	-	24	36%	55%
0.03	10	10	21	50	-	36	30%	66%
0.04	10	10	21	50	-	36	30%	66%

ตารางที่ 4.4 แสดงการปรับค่า Confidence โดยกำหนดค่า Support คงที่

Confidence	**0.1**	**0.2**	**0.3**	**0.4**
กฎที่บันทึกทั้งหมด	567	567	567	567
กฎที่ผ่านการ Prune	142	134	132	125
ทำนายถูกต้อง	31	29	29	21
ค่าความแม่นยำ (1)	43%	40%	40%	29%
ค่าความแม่นยำ (2)	47%	45%	45%	30%



รูปที่ 4.7 แสดงกราฟเปรียบเทียบกฎที่เหลือจากการ Prune และความแม่นยำ

ผลการทดลองโดยการใช้วิธีการกำหนดขนาดการสไลด์วินโดร์แบบจำกัด โดยมีขนาดในการสไลด์วินโดว์เท่ากับ 4 ซึ่งการใช้วิธีการดังกล่าวนี้จะส่งผลเสียคือกฎบางกฎอาจจะไม่ถูกค้นพบนั้นเอง ซึ่งได้อธิบายไว้ก่อนหน้านี้แล้ว ข้อมูลที่ใช้ประกอบด้วย Data Training 70 Pattern และใช้ Data Testing 72 Pattern

ตารางที่ 4.5 แสดงการเปรียบเทียบค่าความแม่นยำที่ ค่า Support ต่าง และ Confidence = 0 เมื่อมีการจำกัดขนาดของการสไลด์วินโดว์ให้มีขนาดเท่ากับ 4 เท่านั้น

Support	กฎที่บันทึกทั้งหมด	กฎที่ผ่านการ Prune	มี Pattern		ไม่มี Pattern		ความแม่นยำ (1)	ความแม่นยำ (2)
			ถูก	ผิด	ถูก	ผิด		
0.01	248	76	21	50	-	18	30%	37%
0.02	8	4	12	59	-	55	17%	77%
0.03	2	2	8	63	-	62	11%	87%
0.04	2	2	8	50	-	62	11%	87%

กำหนดให้ในการสไลด์ข้อมูลนั้นมีการจำกัดของวินโดว์ โดยเริ่มต้นที่การปรับขนาดเท่ากับ 2 และมีความยาวสูงสุดที่ 4 เพื่อหาความแม่นยำของข้อมูล จะเห็นได้ว่าในกรณีที่กำหนดขนาดการสไลด์ของวินโดว์ บางครั้งอาจจะมีผลต่อความแม่นยำด้วยเพราะกฎบางตัวอาจจะไม่สามารถค้นพบนั้นเอง

ตารางที่ 4.6 แสดงความแม่นยำที่ ค่า Support ต่างๆ เมื่อมีการจำกัดขนาดวินโดว์เริ่มต้น ที่ 2 และมีขนาดสูงสุดที่ 4

Support	กฎที่บันทึกทั้งหมด	กฎที่ผ่านการ Prune	มี Pattern		ไม่มี Pattern		ความแม่นยำ (1)	ความแม่นยำ (2)
			ถูก	ผิด	ถูก	ผิด		
0.01	360	148	41	30	-	6	58%	37%
0.02	26	26	26	45	-	24	37%	77%
0.03	10	10	21	50	-	36	30%	87%
0.04	10	10	21	50	-	36	30%	87%

กำหนดให้ในการสไลด์ข้อมูลนั้นมีการจำกัดของวินโดว์ โดยเริ่มต้นที่การปรับขนาดเท่ากับ 2 และมีความยาวสูงสุดที่ 3 เพื่อหาความแม่นยำของข้อมูล

ตารางที่ 4.7 แสดงความแม่นยำที่ ค่า Support ต่างๆ เมื่อมีการจำกัดขนาดวินโดว์เริ่มต้นที่ 2 และมีขนาดสูงสุดที่ 3

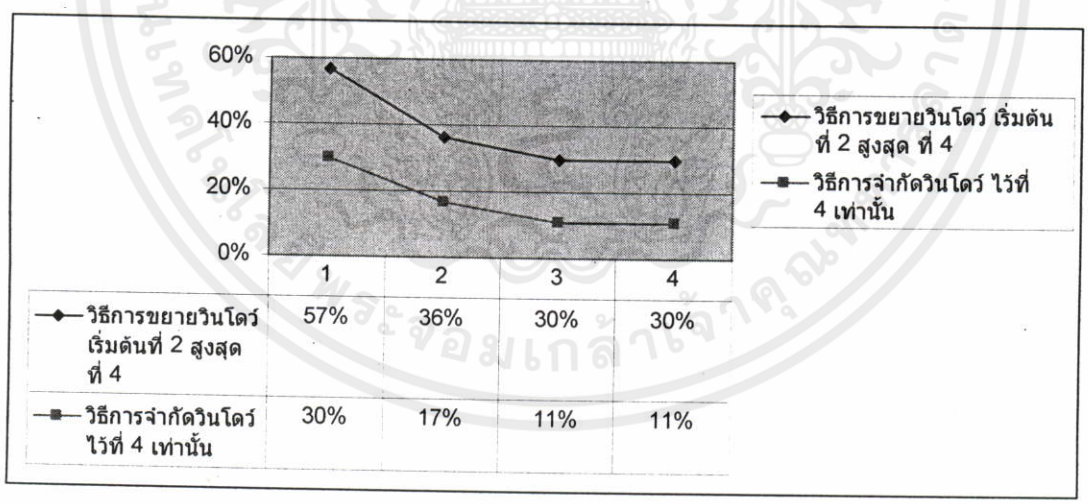
Support	กฎที่บันทึกทั้งหมด	กฎที่ผ่านการ Prune	มี Pattern		ไม่มี Pattern		ความแม่นยำ (1)	ความแม่นยำ (2)
			ถูก	ผิด	ถูก	ผิด		

0.01	237	148	41	30	-	6	58%	37%
0.02	26	26	26	45	-	24	37%	77%
0.03	10	10	21	50	-	36	30%	87%
0.04	10	10	21	50	-	36	30%	87%

กำหนดให้ในการสไลด์ข้อมูลนั้นมีการจำกัดของวินโดว์ มีความยาวสูงสุดที่ 2

ตารางที่ 4.8 แสดงความแม่นยำที่ ค่า Support ต่างๆ เมื่อมีการจำกัดขนาดวินโดว์มีขนาดสูงสุดที่ 2

Support	กฎที่บันทึกทั้งหมด	กฎที่ผ่านการ Prune	มี Pattern		ไม่มี Pattern		ความแม่นยำ (1)	ความแม่นยำ (2)
			ถูก	ผิด	ถูก	ผิด		
0.01	237	148	41	30	-	6	58%	37%
0.02	26	26	26	45	-	24	37%	77%
0.03	10	10	21	50	-	36	30%	87%
0.04	10	10	21	50	-	36	30%	87%



รูปที่ 4.8 แสดงกราฟความแม่นยำของวิธีจำกัดและไม่จำกัดขนาดของการสไลด์

ตารางที่ 4.9 แสดงความแม่นยำระหว่างวิธีจำกัดและไม่จำกัดขนาดของการสไลด์วินโดว์

Support	0.01	0.02	0.03	0.04
วิธีการขยายวินโดว์ เริ่มต้นที่ 2 สูงสุด ที่ 4	57%	36%	30%	30%
วิธีการจำกัดวินโดว์ ไว้ที่ 4 เท่านั้น	30%	17%	11%	11%

จากกราฟ จะเห็นว่า วิธีการจำกัดวินโดว์ไว้ที่ 4 นั้นจะทำให้การค้นพบกฎน้อยลง ซึ่งจะส่งผลให้เมื่อนำกฎเหล่านั้นไปใช้ในการทำนาย ความแม่นยำก็จะลดน้อยลงไปด้วย

บทที่ 5
สรุปผลการวิจัย และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

เนื่องจากในงานวิจัยเรื่องอัลกอริธึมใหม่สำหรับการค้นหารูปแบบเว็บไซต์ นั้นมีด้วยกันหลัก 3 ส่วนหลักด้วยกัน ดังนี้คือ

- การเตรียมการข้อมูลในการประมวลผล Preprocessing Process
- อัลกอริธึมในการค้นหารูปแบบเว็บไซต์ Sequential Pattern Discovery
- การลดทอนกฎ Pruning Process
- การสร้าง Prediction model

ซึ่งแต่ละส่วนมีการทำงานดังต่อไปนี้

Preprocessing Process ประกอบด้วยขั้นตอน

- การนำข้อมูลล็อกไฟล์เข้ามาประมวลผล
- การทำการ Cleaning Process โดยการตัดทิ้งข้อมูลหรือ Request ที่ไม่มีการใช้งานทิ้งไป ดังเช่น Request ที่เป็นลักษณะภาพ, เสียง, หรือสิ่งที่ไม่จำเป็นอื่นๆ ในงานวิจัย
- ขั้นตอนการจัดเรียงว่า User แต่ละคนมีการเรียกใช้เว็บเพจอะไรบ้าง
- Session Identification การแยกแยะว่า User แต่ละคนมีการเรียกใช้เว็บเพจอะไรบ้าง ภายในระยะเวลาที่กำหนด โดยในงานวิจัยได้กำหนดไว้ที่ 120 นาที ซึ่งผลของกระบวนการดังกล่าวนี้จะนำไปสู่การสร้างทรานเซ็กชั่น
- Transaction Identification การบ่งชี้ว่า User แต่ละท่านมีการเรียกใช้งานเว็บเพจอย่างไร โดยมีการนำอัลกอริธึม Maximum Forward Reference เข้ามาทำการวิเคราะห์ โดยแนวคิดของอัลกอริธึมดังกล่าวนี้จะพิจารณาเมื่อมีการคลิกหรือกดปุ่ม Back นั้นเอง หรือกล่าวได้ว่าจะเป็นการทราบว่าจุดลึกสุดของเว็บเพจที่ User เข้าเรียกใช้งานนั้นเอง

Sequential Pattern Discovery ประกอบด้วยขั้นตอน

- การค้นหากฎ โดยใช้แนวคิดของ Sequential Pattern Discovery ซึ่งจะเป็นลักษณะการทำงานของ Association Rule โดยกระบวนการดังกล่าวนี้ได้มีการเปรียบเทียบการทำงานกับ Sequential Algorithm ซึ่งอัลกอริธึมดังกล่าวนี้หากนำมาใช้งานกับเว็บไซต์จะเกิดการกระโดดของเว็บเพจ ดังนั้นจะส่งผลในการทำนายมีประสิทธิภาพต่ำลงนั้นเอง
- การสร้าง Operation Join จากผลลัพธ์ที่ใช้ Sequential Algorithm เมื่อข้อมูลเป็นเว็บล็อกไฟล์นั้น ทำให้เกิดแนวคิดในการแก้ปัญหาการกระโดดของข้อมูล ขึ้นมากซึ่งตามที่ได้กล่าวถึงรายละเอียดการทำงานในบทก่อนหน้า ดังนั้นจะเห็นว่าสร้าง Operation Join

ขึ้นมาซึ่งได้มีการปรับปรุง Operation Join ขึ้นมาใหม่ให้มีความเหมาะสมกับข้อมูลที่เป็นลักษณะเว็บเพจนั้นเอง

Pruning Process ประกอบด้วยขั้นตอน

- กระบวนการสร้างต้นไม้ เมื่อผ่านการระบวนการค้นหากฎต่างๆเป็นที่เรียบร้อยแล้ว จะได้กฎจำนวนมากมาใช้งาน ดังนั้นจะเห็นว่าหากกฎจำนวนมากนำไปใช้งานในการทำนาย ซึ่งกฎบางกฎ อาจจะไม่จำเป็นต้องเก็บก็ได้เพราะอาจจะเป็นการใช้งานที่ซ้ำซ้อน ยกตัวอย่างเช่น

  Pattern 1: A.HTML,B.HTML,C.HTML,D.HTML → E.HTML

  Pattern 2: X.HTML,Y.HTML,C.HTML,D.HTML → E.HTML

  Pruning Rule : 1. C.HTML,D.HTML → E.HTML

  2. B.HTML,C.HTML,D.HTML → E.HTML

  ซึ่งความจำเป็นในการเก็บกฎ ที่ 2 อาจจะไม่จำเป็นเลยเพราะว่า การเก็บกฎที่ 1 จะสามารถมีประโยชน์ในการใช้งานได้กว้างขวางกว่า นั้นเอง จะเห็นได้ว่า หากเราเก็บ Pruning Rule 1: เราสามารถใช้ในการทำนาย ได้ทั้งสอง Pattern นั้นเอง ดังนั้นจึงได้คิดหาแนวทางในการลดทอนกฎ ซึ่งวิธีที่นำมาใช้งานคือ การสร้างต้นไม้ โดยที่แต่ละโหนดจะแทนด้วย กฎ และจะมีค่าจำเพาะคือค่า Confidence ประจำโหนดด้วย โดยแต่ละโหนดจะใช้การเปรียบเทียบเพจทำนายและค่าความเชื่อมั่น ซึ่งผลลัพธ์จากกระบวนการดังกล่าวนี้จะทำให้เหลือกฎที่มีประสิทธิภาพในการทำนายมากที่สุดนั้นเอง
- การลดทอนกฎ จะใช้วิธีการเปรียบเทียบค่าความเชื่อมั่น Confidence และมีการเปรียบเทียบเพจที่ใช้ในการทำนาย โดยจะเปรียบเทียบโหนดแม่กับโหนดลูกนั้นเอง หากโหนดแม่มีค่าความเชื่อมั่นมากกว่า ก็จะตัดโหนดลูกทิ้งทั้งหมด

Prediction Model ประกอบด้วยขั้นตอน

เมื่อเราได้กฎจากการสร้างต้นไม้เสร็จสิ้นแล้ว ต่อจากนั้นจะนำข้อมูลอินพุทหรือ Testing Data Source เพื่อทำการตรวจสอบความแม่นยำว่า กฎที่ผ่านการพรุ่นนิ่งนั้นมีความแม่นยำมากน้อยเพียงใด

- โดยการทำงานนั้นจะเป็นการลดขนาดของ Input Pattern แล้วไปเปรียบเทียบกับกฎที่ผ่านการพรุ่น ซึ่งหากไม่เจอกฎดังกล่าวนี้ก็จะทำการลดขนาดไปเรื่อยๆ จนเหลือเท่ากับ 1 หากยังไม่เจออีกจะนำ Default Root เป็นเพจในการทำนาย ในกรณีที่มีเพจเหมือนกัน จะเลือกเพจที่ใช้ในการทำนาย โดยการตรวจสอบค่าความเชื่อมั่นนั้นเอง
- การหาค่าความแม่นยำ โดยผลลัพธ์จากการทำนายจะมีการหาอัตราส่วนของ Pattern ที่ทำนายถูกต่อ Pattern ทั้งหมด

5.2 ประสิทธิภาพของอัลกอริธึม

ในงานวิจัยนี้เรามุ่งเน้น ในการประมวลผลของอัลกอริธึมเป็นหลัก และให้ความสำคัญมากที่สุด แต่จากปัจจัยดังกล่าวข้างต้นแล้วนั้น เมื่อเราได้นำอัลกอริธึมมาร่วมกระบวนการทั้งหมด เพื่อที่จะหาความแม่นยำในการทำนาย สิ่งหนึ่ง ที่มีปัจจัยหรือมีผลต่อความแม่นยำของอัลกอริธึม นั้นคือ กระบวนการในการเตรียมข้อมูล

ดังจะเห็นว่าอัลกอริธึมใหม่สำหรับการค้นหารูปแบบเว็บไซต์ นั้นจะมีความเหมาะสมและทำงานได้อย่างสมบูรณ์กับข้อมูลหรือเว็บล็อกไฟล์ที่มีการเกิดของ Pattern ที่มีความแตกต่างกันน้อย หากใช้กับข้อมูลที่เป็นลักษณะการเกิดของ Pattern ที่มีความหลากหลายจะส่งผลให้ประสิทธิภาพน้อยลงนั้นเอง

5.3 ข้อเสนอแนะ

5.3.1 การกำหนดระยะเวลาของ Session Identification

การกำหนดระยะเวลาของการแยกแยะ Session นั้นจะมีผลต่อ Transaction ที่ได้รับด้วย ทั้งนี้ขึ้นอยู่กับข้อมูลที่นำเสนอของแต่ละเว็บเพจนั้น หากมีการกำหนดให้เหมาะสมจะทำให้ข้อมูลที่ได้มีความแม่นยำมากยิ่งขึ้น

5.3.2 ควรมีการทดลองกับข้อมูลต่างๆให้มากขึ้น อีกทั้งควรเน้นย้ำในกระบวนการเตรียมการของข้อมูลอีกด้วย

บรรณานุกรม

[1] J. Srivastava, R. Cooley, M. Deshpande and P. Tan, "**Web Usage mining: Discovery and Applications of Usage Patterns from Web Data**", ACM SIGKDD , Jan 2000.

[2] U. Fayyad,G. Piatetsky-Shapiro,and P. Smyth. **From data mining to knowledge discovery: An overview.** InProc. ACM KDD, 1994.

[3] M.S. Chen, J. Han, and P.S. Yu. **Data mining: An overview from a database perspective.** IEEE Transactions on Knowledge and Data Engineering, 8(6):866-883, 1996.

[4] R. Agrawal and R. Srikant. **Fast algorithms for mining association rules.** In Proc. of the 20th VLDB Conference, Santiago, Chile, 1994.

[5] J.Pitkow and P.Pirolli. **Mining Longest Repeating Subsequences to Predict World Wide Web Surfing.** In Second USENIX Symposium on Internet Technologies and Systems, C0, 1999.

[6] lan Tian Yi Li. **Web-document prediction and presending using association rule sequential classifiers**, Simon Fraser University, S2001

[7] Z. Su, Q. Yang, Y. Lu, and H. Zhang. **Whatnext: A Prediction System for Web Requests Using N-gram Sequence Models.** In Proc. of the First Int'l Conf. on Web Information Systems and Engineering Conference, Hong Kong June 2000.

[8] Q. Yang, H. Zhang, and T. Li.**Mining Web Logs for Prediction Models in WWW Caching and Prefetching.** In Proc. ACM SIGKDD, 2001.

[9] I.Zukerman, D.W.Albrecht, and A.E.Nicholson. **Predicting User's Request on the WWW.** In Proce., Monash University, June 1999.

[10] R. Agrawal and R. Srikant. **Fast algorithms for mining associatin rules**. In Proc. Of the VLDB Conference, Santiago, Chile, September 1994. Expanded version available as IBM Research Report RJ9839, June 1994.

[11] R. Agrawal and R. Srikant. **Mining Sequential patterns.** Research Report RJ 9910, IBM Almaden Research Center, Sanjose, California, October 1994.

ภาคผนวก.

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. Bunyat Thipmoud and Worapoj Kreesuradej, "**A New Sequential Pattern Discovery Algorithm for Web Usage Mining**", WSEAS Transactions on Computers, Salzburg, Austria ,3 July 2004, pp. 801-806.

2. Worapoj Kreesuradej and Bunyat Thipmoud, "**A New Algorithm for Web Sequential Pattern Discovery**", The 2004 International Technical Conference on Circuits/Systems, Computers and Communication (ITC-CSCC2004), Hotel Taikanso, Sendai/Matsshima, Miyagi-Pref, JAPAN, July 6-8 2004.

61

WSEAS TRANSACTIONS on COMPUTERS

Issue 3, Volume 3, July 2004
ISSN 1109-2750 http://www.wseas.org

802 WSEAS TRANSACTIONS ON COMPUTERS, Issue 3, Volume 3, July 2004 ISSN: 1109-2750

It is noticeable that the problem stated above as rule of 1,2,3..n sequential pattern rule could not reach decision criteria. It, then, could not be considered in any further stages. The problem solving in this paper that the size of slide window, is adaptive window size could confidentially solve the problem described above.

2. Rule Representation Method

Given a web log. We consider web log data as a sequence of distinct web pages where subsequences, such as users' sessions can be observed within unusually long gaps between consecutive requests.

Time	User ID	Requested Document
00:00:01	U1	A
00:00:02	U2	B
00:00:03	U2	C
00:00:04	U3	D
00:00:05	U1	E
.....	.....	.....

Table 1 Example user visit sequence

This sequence is divided into user sessions according to user Ids Shown in table 2.

User ID	Session Sequence
U1	A, E,...
U2	B,C,...
U3	D,..
....	....

Table 2 Extracted user sessions

We now discuss how to extract rules of the form LHS→RHS from the log table. As it was mentioned before in this paper, the RHS in each association rule is the next page requested by the user. This is a different method to extract rule.

The representation is called the *latest-substring* rules. This is also known as n-gram rules in some literature [8]. These rules are not only take into account the order and adjacency information, but also the recent information about the LHS string.

LHS-RHS	RHS	Extracted Rules
A, B, C	D	<A, B, C> → E, <B, C> → D, <A> → D, <C> → D

Table 3 The latest-substring rule

For each rule of the form LHS-RHS, we define the *support* and *confidence* as follows:

$$sup = \frac{count(LHS, RHS)}{count(Table)} \qquad conf = \frac{sup(LHS, RHS)}{sup(LHS)}$$

$$sup(LHS) = \frac{count(LHS)}{count(Table)}$$

For the equations above, the function Count (Table) is the record set in the log table. The way of pruning here is exactly the same as in all association mining algorithms [4]. The approaches of association rule mining, two parameters: *minimum support* and *minimum confidence* can be pre-defined to filter out some rules with very low support and/or minimum confidence [4].

The latest substring representation interested references the experiment such as [6].

2.1 Extract processing of log file table:

- Firstly, the system will read all data from Log file then transaction log file incurred as Transaction Log File $Tlog = \{R1, R2, R3, ...Rn\}$.
- Data cleaning is the next step. The method of cleaning server log is to eliminate irrelevant items of web log analysis. The output processed by cleaning model will result in untruth data the best use of cleaning model applied HTML log file.
- The output clean model could be interpreted as web transaction table. Web transaction table is written as $WebT = \{R1, R2, R3, ...Rn\}$.
- Lastly, reading each records as $Read \leftarrow \{P1, P2, ...Ps\}$ goes by IP number. The result is kept in variable.

3. Algorithm

Definitions:

Support: the support of a rule "X → Y" is the probability (or attribute sets) X and Y occurring together in the same transaction

Confidence: the confidence of a rule "X → Y" is defined as the probability of occurrence of X and Y together in all transaction in which X already occurs.

Minsup: Minsup is an input parameter to the algorithm for generating association rules

Minconf: Minconf is an input parameter that defines the minimum level of confidence that a rule must process

Web Transaction Table: is where all transactions on web that were already passed cleaning model

Moving window pair (LHS-RHS): To capture the sequential and time limited nature of predictions that composted of two adjacent windows.

Antecedent window (LHS): It holds all visited

Y → X	2	1	0.142	0.5
Y → X1	2	1	0.142	0.5
X1 → X2	4	1	0.142	0.25

Table 5 Full Table1

Based on this Table, LHS, LHS - RHS, Support, and Confidence are recoded with the obviously that C → D has Confidence = 100%. This means whenever Access User starts at C, it would absolutely result in Consequent Window (RHS) as D without any questions. When data of Full Table 1 completed the process of Reduce Rule (filter out), data in Trim Table 1 appeared which some rules that value of support is less then threshold will filter our (if support ≤ 0.142) , such as X4 → X 5, X5 → D, Y → X, Y → X1, X1 → X2 while the rest rules are recorded in Trim Table 1 would be D → E, E → D, E → Y, X1 → C, E → Z as shown in Table 6.

Trim Table1	RULE	LHS	LHS-RHS	Sup	Conf
D → E	C, D→E	3	3	0.428	1
	E, D→E	2	2	0.285	1
	X5, D→E	1	1	0.142	1
E → D	D, E→D	8	4	0.428	0.125
E → Y	D, E→Y	8	4	0.428	0.5
X1→C	X1→C	2	1	0.142	0.5
E → Z	D, E→Z	8	3	0.428	0.375

Table 6 Full Table 2

Table 6 works as following procedure. When iCW2 slide window with the value that compares with rule recorded in Table of iCW2 1 or Trim Table 1, LHS-RHS from slide window would be split into 3 major parts called LHS, LHS - RHS and A_R. Where as LHS and LHS – RHS applied in support and confidence value calculation, A_R takes an important role to record value within itself compared to rule. If the result of this comparison states the same value, it means LHS – RHS currently be able to process in the next step. In case that A_R could not find any value in Trim Table 1, shift window moving pair to {P1, P2, ... , Pn} continues to incur.

The last value recorded in Full Table 2 will be filter out and then the rest are kept in Trim Table 2 for further next process. It is obviously seen that the rest value is quite small with the Rule of D, E → Y, D, E → Z

	RULE	LHS	LHS-RHS	Sup	Conf
E→Y	D, E→Y	4	8	0.5714	0.5
E→Z	D, E→Z	8	3	0.428	0.375

Table 7 Trim Table 2 and Full Table 3

As you may noticed that the more difficulties access pattern has, the less of support value incurred, Table 6 with additional iCW=4, compared with Trim Table 3 resulted in none identical value hence no Rule in Full Table 4. This could be called as the end of process.

Trim Table2	RULE	LHS	LHS-RHS	Sup	Conf
D E → Y	E, D, E → E	2	2	0.285	1
	X5, D, E → Y	1	1	0.142	1
D E → Z	C,D, E → Z	3	3	0.428	1

Table 8 Full Table 3

	RULE	LHS	LHS-RHS	Sup	Conf
E, D, E → E	-	-	-	-	-
C,D, E → Z	-	-	-	-	-

Table 9 Trim Table 3

We construct the Tree according to the following relations:

- Each rule is represented by a node in the Tree.
- The node representing the direct parent rule is the parent node of the node(s) representing the direct children rule(s).
- The root of the Tree representing the default rule.

The pruning process just traverses the tree using post-order traverse. If a not has a lower confidence than that of its direct parent nodes, or it predicts the same class as its direct parent node, it can be pruned and all of its children nodes (if any) promoted to be the children of its direct parent node [6]. Figure 5 is a diagram tree created from Table 4 after Discovery and reduce rule completed. Rules left as show in Figure 6 that every single nodes have value of Conf. According to the concept of pruning tree method stated earlier, tree that completed the process will be as Figure 7 while;

- $E \rightarrow Y_{Conf} < Root_{Conf}$ then pruned and promote D,E→Y that node will not be pruned because the value of conf is more than root. It is essential that E,D,E→Y was pruned because prediction is same as prediction of parent that is node D, E→ Y.
- $E \rightarrow D_{Conf} < Root_{Conf}$ was pruned.
- E,D→E and C,D→E was pruned because the prediction is same as parent prediction.
- $E \rightarrow Z_{Conf} < Root_{Conf}$ was pruned then promoted D,E→ Z to be parent of C,D,E→Z.
- $D,E \rightarrow Z_{Conf} < Root_{Conf}$ was pruned then promoted C,D,E→Z.

adjacency information, but also the recent information about the LHS string.

LHS-RHS	RHS	Extracted Rules
A, B, C	D	<A, B, C>→ D, <B, C> → D, <C>→ D

Table 2 The latest-substring rule

For each rule of the form LHS-RHS, we define the *support* and *confidence* as follows:

$$\sup = \frac{count(LHS, RHS)}{count(Table)} \qquad conf = \frac{\sup(LHS, RHS)}{\sup(LHS)}$$

$$\sup(LHS) = \frac{count(LHS)}{count(Table)}$$

For the equations above, the function Count (Table) is the record set in the log table. The way of pruning here is exactly the same as in all association mining algorithms [4]. The approaches of association rule mining, two parameters: *minimum support* and *minimum confidence* can be pre-[illegible] to filter out some rules with very low support and/or [illegible] confidence [4]. The latest substring representation interested references the experiment such as [6].

2.1 Extract processing of log file table:

- Firstly, the system will read all data from Log file then transaction log file incurred as Transaction Log File
  Tlog = {R1,R2,R3, ...Rn}.
- Data cleaning is the next step. The method of cleaning server log is to eliminate irrelevant items of web log analysis. The output processed by cleaning model will result in untruth data, the best use of cleaning model applied to HTML log file.
- The output clean model could be interpreted as web transaction table. Web transaction table is written as
  WebT = {R1,R2,R3, ... Rn}.
- Lastly, reading each records as *Read* ← *{P1,P2,....PN}* goes by IP number. The result is kept in variable.

3. Algorithm

Begin main:
Open Web Log File;
- Extract web log record in to a main table, Data Cleaning
- Extract Web Transaction according to IP.
- Grouping web transaction based on web page visited. The record will be stamped according to the time consequence.
- Web transaction Table according to IP separation resulted from previous grouping. This table is in order of IP that visit web page consequently, according to order of web page.

L_1 = {large 1-sequences};
For (k=2; $L_{k-1} \neq \emptyset$; k++) **do**
Begin
C_k New candidate generated from L_{k-1}
(See a new candidate generation)
- **for each** web-page sequence c in the web transaction table **do**
Increment the count of all candidates in C_k that are contained in c.
L_k = Candidates in C_k with minimum support.
End

Figure 1 A New Algorithm for Web Sequential Pattern Discovery

Notation. In all the algorithm, L_k denotes the set of all large k-sequences, and C_k the set of candidate k-sequences.
The algorithm is given in figure 1. In each pass, we use the large sequences from the previous pass to generate the candidate sequence and then measure their support by making a pass over the web log transaction. At the end of pass, the support of the candidates is used to determine the large sequence. In the first pass, the output of the large phase is used to initialize the set of large 1-sequences.

The Candidate generate function take as argument L_{k-1}, the set of all large (k-1)-sequences. The function work as follows. First. Join L_{k-1} with L_{k-1}

$$a_1a_2a_3...a_n \otimes b_1b_2b_3...b_m = \begin{cases} \phi & \text{if } a_n \neq b_1 \\ a_1\, a_2\, a_3 \ldots a_n & \\ b_1 = a_n & \end{cases}$$

Figure 2 A new Candidate Generations

In order to create Ck, we have defined new meaning of Join Operation as referring to [10], the output is created from the candidate generation procedure, results the sequence pattern to overstep. When the application of web based transaction incurred, it is absolutely unappreciated on the reason that the process of prediction model is related to next request web page. If the output of sequential pattern, which is overstepped, is applied, it will decrease the accuracy. For example: A new Candidate Generation

D E ⊗ E D => D E→D : D E ⊗ E Y => D E→Y
E D ⊗ D E => E D → E : D E ⊗ E Z => D E → Z

4. Experiments

The rules established at the beginning of Algorithm experiment. These are D,E → Y, X1 → C and C, D, E → Z. These rules were set to compile with log table then additional information is filled out. Lastly algorithm principle mentioned in this paper was applied and resulted as following.

Table 3: Web Transaction Table, illustrates all data of 7 IP. Given each users have different access pattern calling as P1 to P5, all of these data were already processed Cleaning Model. After we have data as shown in Web Transaction Table 3. The Moving Window Pair which has same identification that is incurred from same user by IP inspection method would be recorded as one record. The repeated access pattern would be discarded.

IP	P1	P2	P3	P4	P5
1	D	E	D	E	Y
2	X4	X5	D	E	Y
3	X1	C	D	E	Z
4	X1	C	D	E	Z
5	X1	C	D	E	Z
6	E	D	E	Y	X
7	D	E	Y	X1	X2

Table 3 Web transaction table

For example: IP1 has Access Pattern; D → E, E → D, D → E, E → Y. It is noticeable that D → E incurred twice, and then access pattern of the repeated on is discarded. The count method is only one time; LHS →RHS = 1 and LHS = 2. The benefit of unrepeated count is described earlier, hence the end of this Algorithm generated data showing in Table 4:

RULE	LHS	LHS-RHS	Sup	Conf
D → E	8	7	1	0.875
E → D	9	2	0.285	0.222
E → Y	9	4	0.571	0.444
X1 → C	4	3	0.428	0.75
C → D	3	3	0.428	1
E → Z	9	3	0.428	0.333
Y → X1	2	1	0.142	0.5

Table 4 Candidate Generation C_2

Based on this Table, LHS, LHS – RHS, Support, and Confidence are recoded with the obviously that C → D has Confidence = 100%. This means whenever Access User starts at C, it would absolutely result in Consequent Window (RHS) as D without any questions.

When data of Candidate Generation C_2 Table completed the process of Reduce Rule (filter out), data in Table appeared which some rules that value of support is less then threshold will filter our (if support ≤ 0.142) , such as Y →X1 while the rest rules are recorded in Table would be D → E, E → D, E → Y, X1 → C, C → D, E → Z. Then some rule will pass the A new Candidate Generations for join operation, the rules will be

D E ⊗ E D => D E→D : E D ⊗ D E => E D →E
D E ⊗ E Y => D E→Y : X1 C ⊗ C D => X1 C →Y

RULE	LHS	LHS-RHS	Sup	Conf
D, E→D	8	1	0.1428	0.1251
D, E→Y	8	4	0.571	0.5
D, E → Z	8	3	0.428	0.375
E, D→E	2	2	0.285	0.125
X1,C → D	3	3	0.428	1
C, D→E	3	3	0.428	1

Table 4 Large Sequence L_2

The last value recorded in Table 5 Large Sequence L_3 will be filter out and then the rest are kept in Table 6 for further next process.

RULE	LHS	LHS-RHS	Sup	Conf
D, E, D→E	1	1	0.1428	1
E, D, E→D	2	0	0	0
E, D, E → Y	2	2	0.285	1
E, D, E → Z	2	0	0	0
X1, C, D → E	3	3	0.428	1
C, D, E → D	3	0	0	0
C, D, E → Y	3	0	0	0
C, D, E →Z	3	3	0.428	1

Table 5 Candidate Generation C_3

RULE	LHS	LHS-RHS	Sup	Conf
E, D, E → Y	2	2	0.285	1
X1, C, D → E	3	3	0.428	1
C, D, E →Z	3	3	0.428	1

Table 6 Large Sequence L_3 (Reduced)

RULE	LHS	LHS-RHS	Sup	Conf
X1,C,D,E→ Z	3	3	0.428	1

Table 6 Candidate Generation C_4

No candidate is generated for the fourth pass as you [illegible] would be called as the end of process.

Next step We construct the Tree according to the following relations:

- Each rule is represented by a node in the Tree.
- The node representing the direct parent rule is the parent node of the node representing the direct children rule(s).
- The root of the Tree representing the default rule.

The pruning process just traverses the tree using post-order traverse. If a not has a lower confidence than that of its direct parent nodes, or it predicts the same class as its direct parent node, it can be pruned and all of its children nodes (if any) promoted to be the children of its direct parent node [6]. Figure 3 is a diagram tree created from Table 3 after Discovery and reduce rule completed. Rules left as show in Figure 4 that every single nodes have value of Conf. According to the concept of pruning tree method stated earlier, tree that completed the process will be as Figure 5;

- $E \rightarrow Y_{Conf} < Root_{Conf}$ then pruned and promote D,E→Y that node will not be pruned because the value of conf is more than root. It is essential that E,D,E→Y was pruned because prediction is same as prediction of parent that is node D, E→ Y.
- $E \rightarrow D_{Conf} < Root_{Conf}$ was pruned.
- E,D→E and C,D→E was pruned because the prediction is same as parent prediction.
- $E \rightarrow Z_{Conf} < Root_{Conf}$ was pruned then promoted D,E→ Z to be parent of C,D,E→Z.
- $D,E \rightarrow Z_{Conf} < Root_{Conf}$ was pruned then promoted C,D,E→Z.

5. Conclusion

This algorithm introduced in this paper is a discovery rule used in prediction model. The idea of reduction irrelevant data and counting precisely could show the capacity of the rule. While the reduce rule could be done at the same time as searching. The rule is reduced, by

confidence and support. Last procedure is pruning rule that will discard redundant and inconsistent rule [6] then the rest will have high accuracy. This prediction helps in either e-commerce to provide client's choice of products or offer the fast view of next page to satisfy those who visit website applying this algorithm.

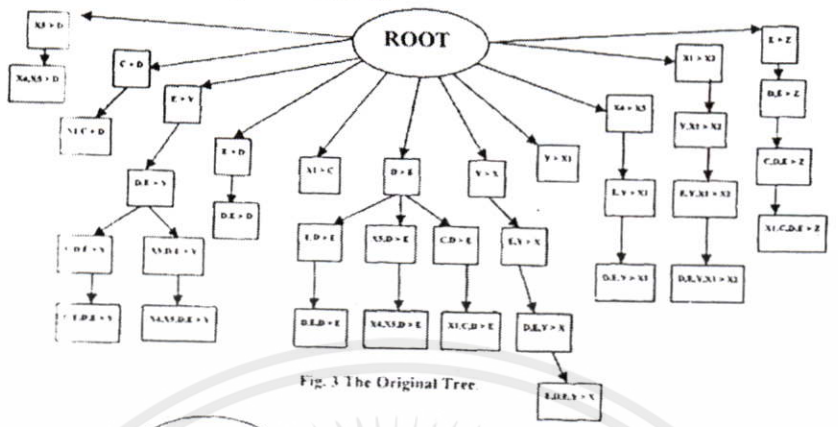


Fig. 3 The Original Tree

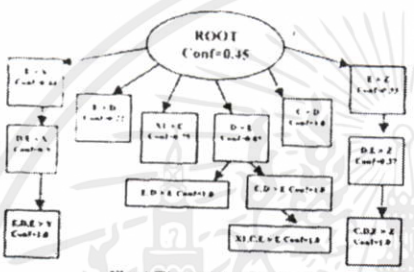


Fig. 4 The tree was reduced

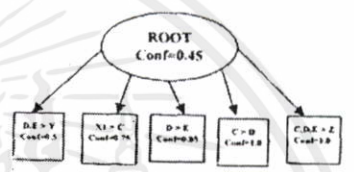


Fig. 5 The tree was pruned

References:

[1] J. Srivastava, R. Cooley, M. Deshpande and P. Tan. "*Web Usage mining: Discovery and Applications of Usage Patterns from Web Data*", ACM SIGKDD , Jan 2000.

[2] U. Fayyad,G. Piatetsky-Shapiro,and P. Smyth. *From data mining to knowledge discovery: An overview*. InProc. ACM KDD, 1994.

[3] M.S. Chen, J. Han, and P.S. Yu. *Data mining: An overview from a database perspective*. IEEE Transactions on Knowledge and Data Engineering. 8(6):866-883, 1996.

[4] R. Agrawal and R. Srikant. *Fast algorithms for mining association rules*. In Proc. of the 20th VLDB Conference, Santiago, Chile, 1994.

[5] J.Pitkow and P.Pirolli. *Mining Longest Repeating Subsequences to Predict World Wide Web Surfing*. In Second USENIX Symposium on Internet Technologies and Systems, CO, 1999.

[6] Ian Tian Yi Li. *Web document prediction and presending using association rule sequential classifiers*, Simon Fraser University, S2001

[7] Z. Su, Q. Yang, Y. Lu, and H. Zhang. Whatnext: *A Prediction System for Web Requests Using N-gram Sequence Models*. In Proc. of the First Int'l Conf. on Web Information Systems and Engineering Conference, Hong Kong June 2000.

[8] Q. Yang, H. Zhang, and T. Li.*Mining Web Logs for Prediction Models in WWW Caching and Prefetching*. In Proc. ACM SIGKDD, 2001.

[9] I.Zukerman, D.W.Albrecht, and A.E.Nicholson. Predicting User's Request on the WWW. In Proce., Monash University, June 1999.

[10] R. Agrawal and R. Srikant. Fast algorithms for mining associatin rules. In Proc. Of the VLDB Conference, Santiago, Chile, September 1994. Expanded version available as IBM Research Report RJ9839, June 1994.

[11] R. Agrawal and R. Srikant. Mining Sequential patterns. Research Report RJ 9910, IBM Almaden Research Center, Sanjose, California, October 1994.

ประวัติผู้เขียน

ชื่อ	บัญญัติ ทิพย์หมัด
วัน เดือน ปีเกิด	26 มกราคม 2518
ที่อยู่	เลขที่ 19 หมู่ 2 ต.ตลาดไชยา อ.ไชยา จ.สุราษฎธานี
ประวัติการศึกษา	2540 เทคโนโลยีอุตสาหกรรม
	สาขาเทคโนโลยีอิเล็คทรอนิคส์กำลัง
	สถาบันเทคโนโลยี พระจอมเกล้า พระนครเหนือ
ประวัติการทำงาน	
	เจ้าหน้าที่คอมพิวเตอร์ ระดับ 2 กรมสรรพากร
	เจ้าหน้าที่ดูแลระบบคอมพิวเตอร์
	ภาควิชาเทคโนโลยี สารสนเทศ สถาบันเทคโนโลยีนานาชาติ
	สิรินธร มหาวิทยาลัยธรรมศาสตร์ (รังสิต)
	วิศวกรระบบ (System Engineer: Batching System)
	พนักงานบริษัท โบรอล(ประเทศไทย) จำกัด