บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 โครงสร้างข้อมูลของล็อกไฟล์ของเว็บเซิร์ฟเวอร์

ล็อกไฟล์ของการเข้าถึงเว็บเซิร์ฟเวอร์จะบันทึกกิจกรรมต่างๆที่เกิดขึ้นจากการที่ผู้ชมเข้าถึงไฟล์ที่อยู่บนเซิร์ฟเวอร์ NCSA ได้กำหนดมาตรฐานของรูปแบบของล็อกไฟล์ไว้สองรูปแบบ คือ แบบคอมมอน[7] และแบบคอมไบน์หรือเอ็กซ์เทนด์[8] ตัวอย่างที่ตามมานี้เป็นตัวอย่างของล็อกไฟล์แบบคอมมอน

tarpon.net - - [12/Jan/1996:20:37:55 +0000] "GET index.htm HTTP/1.0" 200 215

ล็อกไฟล์แบบคอนไบน์หรือเอ็กซ์เทนด์มีลักษณะเหมือนกันกับล็อกไฟล์แบบคอมมอนแต่มีฟิลด์ตำแหน่งที่ใช้ลิงก์มายังเว็บเพจและฟิลด์เบราเซอร์และแพล็ตฟอร์มของผู้ชมเพิ่มเติมขึ้นมาที่ท้ายของบรรทัด(อยู่ในเครื่องหมายคำพูด) ตัวอย่างที่ตามมานี้เป็นตัวอย่างของล็อกไฟล์แบบคอมไบน์

tarpon.net - - [12/Jan/1996:20:37:55 +0000] "GET index.htm HTTP/1.0" 200 215 "http://www.webtrends.com/" "Mozilla/2.0b5 (WinNT; I)"

ตารางที่1 : ฟิลด์ของล็อกไฟล์แบบคอมไบน์

ลำดับฟิลด์ที่	ชื่อฟิลด์	คำอธิบาย
1	User Address	ตัวเลข IP address หรือโดเมนของผู้ที่เยี่ยมชมเว็บไซต์
2	Date/Time	วันและเวลาที่เยี่ยมชมไซต์
3	GMT offset	จำนวนชั่วโมงแสดงระยะห่างจากเวลาสากล GMT (ถ้ามีค่า +0000 หมายถึงล็อกไฟล์ในเวลาสากล)
4	Action	วิธีการดำเนินการ (ได้แก่ GET หรือPOST) ของฮิต (ต้องอยู่ในเครื่องหมายคำพูด)
5	URL Stem	ชื่อไฟล์ที่ถูกกระทำ
6	Protocol Version	เวอร์ชันของโปรโตคอล http ที่ใช้
7	Return Code	โค้ดแสดงผลการตอบสนองคำร้องขอ
8	Server to Client bytes	จำนวนไซต์ที่ส่งไปยังไคลเอนท์
9	Referrer	ตำแหน่งที่ผู้ชมใช้ลิงก์มายังไซต์
10	Browser/Platform	เว็บเบราเซอร์และแพลตฟอร์มที่ใช้ในการเยี่ยมชมไซต์

ตารางที่1 อธิบายค่าข้อมูลของฟิลด์แต่ละฟิลด์ในล็อกไฟล์แบบคอมไบน์ เมื่อพิจารณาแล้วจะเห็นว่าล็อกไฟล์ทั้งสองแบบมีแปดฟิลด์แรกเหมือนกัน แต่ล็อกไฟล์แบบคอมไบน์ (เอ็กซ์เทนด์)

จะมีฟิลด์ที่เก้าแสดงตำแหน่งที่ใช้ลิงก์มายังเว็บเพจและฟิลด์ที่สิบแสดงเบราเซอร์และแพลตฟอร์มของผู้ชมเพิ่มเติมขึ้นมา

```
ip                 time                     method                  url          protocol   code  size
looney.cs.umn.edu han – [09/Aug/1996 : 09:53:52 –0500] "GET /~mobasher/courses/cs5106/cs510611.html HTTP/1.0" 200 9370
mega.cs.umn.edu njain – [09/Aug/1996 : 09:53:52 –0500] "GET / HTTP/1.0" 200 3291
mega.cs.umn.edu njain – [09/Aug/1996 : 09:53:53 –0500] "GET /images/backgnds/paper.gif HTTP/1.0" 200 3014
mega.cs.umn.edu njain – [09/Aug/1996 : 09:53:53 –0500] "GET /images/misc/footer.jpg HTTP/1.0" 200 13366
mega.cs.umn.edu njain – [09/Aug/1996 : 09:54:12 –0500] "GET /cgi-bin/Count.cgi?df-CS-horse.dateid-1 HTTP/1.0" 200 646
mega.cs.umn.edu njain – [09/Aug/1996 : 09:54:18 –0500] "GET /~advisor HTTP/1.0" 302
mega.cs.umn.edu njain – [09/Aug/1996 : 09:54:18 –0500] "GET /~advisor/ HTTP/1.0" 200 487
looney.cs.umn.edu han – [09/Aug/1996 : 09:54:28 –0500] "GET /~mobasher/courses/cs5106/cs510612.html HTTP/1.0" 200 14072
mega.cs.umn.edu njain – [09/Aug/1996 : 09:54:31 –0500] "GET /~advisor/csci-faq.html HTTP/1.0" 200 13786
looney.cs.umn.edu han – [09/Aug/1996 : 09:54:47 –0500] "GET /~mobasher/courses/cs5106/princip.html HTTP/1.0" 200 6965
moose.cs.umn.edu mobasher – [09/Aug/1996 : 09:55:50 –0500] "GET /~suharyon/lisa.html HTTP/1.0" 200 654
moose.cs.umn.edu mobasher – [09/Aug/1996 : 09:55:53 –0500] "GET /~suharyon/line/line16.gif HTTP/1.0" 200 1423
moose.cs.umn.edu mobasher – [09/Aug/1996 : 09:55:57 –0500] "GET /~suharyon/jokol.jpg HTTP/1.0" 200 30890
```

รูปที่ 1 ภาพส่วนหนึ่งของล็อกไฟล์แบบคอมมอน

```
ip              time                 method url protocol code size referrer    browser
9.2.17.16 - - [27/Jun/1997:10:42:03 -0400] "GET / HTTP/1.0" 200 516 "-" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:42:03 –0400] "GET /apache_pb.gif HTTP/1.0" 200 2326 "http://goodwin:8017/" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:42:23 –0400] "GET /manual/index.html HTTP/1.0" 200 2207 "http://goodwin:8017/" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:42:24 –0400] "GET /manual/images/sub.gif HTTP/1.0" 200 6083 "http://goodwin:8017/manual/index.html" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:42:24 –0400] "GET /manual/images/index.gif HTTP/1.0" 200 1540 "http://goodwin:8017/manual/index.html" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:43:56 –0400] "GET /manual/suexec.html HTTP/1.0" 200 19556 "http://goodwin:8017/manual/index.html" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:44:00 –0400] "GET /manual/mod/core.html HTTP/1.0" 200 60679 "http://goodwin:8017/manual/suexec.html#enable" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:44:01 –0400] "GET /manual/images/home.gif HTTP/1.0" 200 1465 "http://goodwin:8017/manual/mod/core.html#group" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:54:20 –0400] "GET /manual/new_features_1_2.html HTTP/1.0" 200 9367 "http://goodwin:8017/manual/index.html" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:54:21 –0400] "GET /manual/cgi_path.html HTTP/1.0" 200 3835 "http://goodwin:8017/manual/new_features_1_2.html" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:54:26 –0400] "GET /manual/new_features_1_1.html HTTP/1.0" 200 8987 "http://goodwin:8017/manual/index.html" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:10:54:29 –0400] "GET /manual/misc/compat_notes.html HTTP/1.0" 200 4691 "http://goodwin:8017/manual/index.html" "Mozilla/3.01 (X11; U; AIX 1)"
9.2.17.16 - - [27/Jun/1997:12:25:20 –0400] "GET /manual/new_features_1_0.html HTTP/1.0" 200 5221 "http://goodwin:8017/manual/index.html" "Mozilla/3.01 (X11; U; AIX 1)"
```

รูปที่ 2 ภาพส่วนหนึ่งของล็อกไฟล์แบบคอมไบน์

ในรูปที่ 1 แสดงตัวอย่างของล็อกไฟล์แบบคอมมอน และในรูปที่ 2 แสดงตัวอย่างของล็อกไฟล์แบบคอมไบน์ โดยแต่ละบรรทัดคือข้อมูลที่เกิดจากการเยี่ยมชมในแต่ละครั้ง เมื่อนำมาเปรียบเทียบกันจะพบว่าล็อกไฟล์แบบคอนไบน์มีลักษณะเหมือนกันกับล็อกไฟล์แบบคอมมอน แต่จะแตกต่างกันเพียงแค่ล็อกไฟล์แบบคอมไบน์จะมีข้อมูลตำแหน่งที่ใช้ลิงก์มายังเว็บเพจกับข้อมูลเบราเซอร์และแพลตฟอร์มของผู้ชมเพิ่มเติมขึ้นมาที่ตอนท้ายของบรรทัด

2.2 แทร็กเกอร์

แทร็กเกอร์ คือ โปรแกรมเล็กๆซึ่งโดยมากจะเป็นโปรแกรมจำพวก CGI ที่ทำงานอยู่ในเว็บเซิร์ฟเวอร์ โปรแกรมเหล่านี้จะไม่ได้ทำงานอยู่ตลอดเวลาแต่จะเริ่มทำงานหรือนับจำนวนผู้ชม ก็ต่อเมื่อมีใครใช้เว็บเบราเซอร์ดึงเว็บเพจขึ้นมาชมดูเท่านั้น เนื่องจากซอร์ซโค้ดสำหรับสั่งให้แทร็กเกอร์ทำงานจะถูกฝังรวมอยู่ในซอร์ซโค้ดของเว็บเพจ ดังนั้นเมื่อมีใครเข้ามาชมเว็บเพจ แทร็กเกอร์ก็จะทำงานโดยบันทึกข้อมูลของผู้ชม เป็นดังนี้ไปเรื่อยๆ แทร็กเกอร์ทำหน้าที่คล้ายกับเคาน์เตอร์แต่ทำได้มากกว่า คือนอกจากนับจำนวนผู้ชมแล้วแทร็กเกอร์ยังเก็บตัวเลขและหลายต่อหลายอย่างเอาไว้แล้วรายงานออกมาเป็นสถิติให้เจ้าของเว็บไซต์ได้วิเคราะห์กัน สถิติที่ว่านี้ก็เช่น จำนวนผู้ชมในแต่ละวัน แต่ละสัปดาห์ แต่ละเดือน แต่ละปี ไปจนถึงการแยกแยะจำนวนผู้ชมที่มาจากแต่ละประเทศแต่ละทวีปก็ยังได้ แทร็กเกอร์ของผู้ให้บริการบางรายไม่เพียงแต่รายงานสถิติเป็นตัวเลขอย่างเดียวเท่านั้น แต่ยังสามารถแสดงสถิติในรูปของกราฟแท่ง กราฟวงกลมอย่างสวยงาม เพื่อให้เจ้าของเว็บไซต์ทำความเข้าใจได้อย่างรวดเร็ว เนื่องจากวัตถุประสงค์หลักของแทร็กเกอร์ไม่ได้อยู่ที่การรายงานผลสถิติให้ผู้ชมเห็นที่เว็บเพจ ดังนั้น เจ้าของเว็บไซต์จึงไม่ค่อยอยากจะให้มีรูปภาพหรือร่องรอยอะไรให้ผู้ชมเว็บเห็นว่ามีแทร็กเกอร์อยู่ในเว็บเพจนี้เลย

ข้อมูลดิบที่นำมาสรุปเป็นสถิติสำหรับแทร็กเกอร์นั้น นอกจากจะมาจากการนับจำนวนผู้ชมแต่ละรายแล้ว ข้อมูลอีกหลายอย่างยังอาจจะมาจากการใช้จาวาสคริปต์เข้าร่วมในซอร์ซโค้ดของแทร็กเกอร์ด้วย ซึ่งจาวาสคริปต์สามารถใช้เพื่อดึงข้อมูลบางอย่างจากเครื่องคอมพิวเตอร์ของผู้ชมเว็บเพจได้ ตัวอย่างง่ายๆเช่น ใช้จาวาสคริปต์เพื่อแสดงวัน-เวลาปัจจุบันจากเครื่องผู้ชมมาแสดงบนเว็บเพจ และเมื่อดึงมาได้ก็ย่อมจะเอาไปรวบรวมเก็บไว้ที่เครื่องของผู้ให้บริการแทร็กเกอร์ได้ด้วยเช่นกัน ยกตัวอย่างเช่น ชื่อและรุ่นของระบบปฏิบัติการในเครื่องคอมพิวเตอร์ของผู้ชม จำนวนสีและความละเอียดของหน้าจอแสดงผล ไปจนถึงการหาแหล่งอ้างอิง (referrer) หรือแหล่งที่มาของผู้ชมเว็บเพจเหล่านี้เป็นต้น เพื่อนำไปสรุปเป็นสถิติแล้วรายงานผลออกมา แต่วิธีนี้มีจุดอ่อนที่เห็นได้ชัดก็คือ ใช้ไม่ได้กับเว็บเบราเซอร์รุ่นเก่าๆที่ไม่สนับสนุนจาวาสคริปต์

2.3 การเรียนรู้กฎความสัมพันธ์[1]

กฎความสัมพันธ์คือกฎที่อธิบายความสัมพันธ์ระหว่างไฟล์ต่างๆที่ผู้ชมดูบนเครื่องเซิร์ฟเวอร์เครื่องหนึ่ง

สำหรับการทำเหมืองเว็บนั้น กฎความสัมพันธ์นี้จะหาได้จากทรานแซคชันของผู้ชมโดยที่แต่ละทรานแซคชันจะประกอบไปด้วยเซตของ URL ที่ผู้ชมคนหนึ่งเข้าเยี่ยมชมเว็บเซิร์ฟเวอร์ในแต่ละครั้ง ตัวอย่างเช่น การค้นหากฎความสัมพันธ์สามารถหาความสัมพันธ์ได้ดังนี้

- 60% ของไคลเอนท์ที่เข้าถึงเว็บเพจที่มี URL /company/products จะเข้าถึงเพจ /company/products/product1.html ด้วย
- 40% ของไคลเอนท์ที่เข้าถึงเว็บเพจ ที่มี URL /company/products /product1.html จะเข้าถึงเพจ /company/products/product2.html ด้วย
- 30% ของไคลเอนท์ที่เข้าถึงเว็บเพจ /company/announcements/special-offer.html จะสั่งซื้อสินค้าแบบออนไลน์ใน /company/products/product1.html

เทคนิคในการค้นหากฎความสัมพันธ์จะเก็บทรานแซคชันของผู้ชมลงในฐานข้อมูล เนื่องจากฐานข้อมูลทรานแซคชันมีขนาดใหญ่มากจึงต้องลดจำนวนทรานแซคชันที่ใช้ในการค้นหาโดยพิจารณจากค่าซัพพอร์ต (ค่าซัพพอร์ต คือ จำนวนครั้งที่ความสัมพันธ์ปรากฏขึ้นภายในฐานข้อมูลทรานแซคชัน)

ตัวอย่างเช่น ถ้าค่าซัพพอร์ตของ /company/products มีค่าต่ำอาจสรุปได้ว่าความสัมพันธ์ระหว่างเว็บเพจ /company/products/product1 และ /company/products/product2 ควรถูกตัดทิ้งเพราะความสัมพันธ์มีจำนวนครั้งที่ปรากฏไม่มากพอ

กฎความสัมพันธ์ที่ค้นพบนี้สามารถบ่งชี้ได้ว่าควรจัดโครงสร้างเว็บไซต์อย่างไรจึงจะเหมาะสมที่สุด ตัวอย่างเช่น

ไคลเอนท์ 80% ที่เยี่ยมชม /company/products และ /company/products/file1.html จะเยี่ยมชม /company/products/file2.html ด้วย แต่มีไคลเอนท์เพียง 30% เท่านั้นซึ่งเยี่ยมชม /company/products/ และยังเยี่ยมชม /company/products/file2.html ด้วย ซึ่งเหมือนกับว่าข้อมูลใน file1.html นำไคลเอนท์ให้ไปเยี่ยมชม file2.html ความสัมพันธ์นี้แนะนำว่าควรจะย้าย file2.html ไปยังระดับที่สูงขึ้น (ได้แก่ /company/products) เพื่อเพิ่มอัตราการเยี่ยมชม file2.html

สำหรับงานของการหากฎความสัมพันธ์นั้น เรานิยามให้ทรานแซคชันคือเซตของล็อกเอ็นทรีทั้งหมดซึ่งเป็นของไคลเอนท์เดียวกัน (มีค่าไอพีแอดเดรสเหมือนกัน) ซึ่งอยู่ภายในช่วงเวลาที่กำหนดโดยผู้ใช้

ให้ L เป็นเซตของล็อกเอ็นทรีของการเข้าถึงเว็บเซิร์ฟเวอร์ ล็อกเอ็นทรี $l \in L$ มีส่วนประกอบดังนี้

- user id ของไคลเอนท์ แทนด้วย l.uid
- URL ของเพจ ที่ถูกไคลเอนท์เยี่ยมชม แทนด้วย l.url และ
- เวลาที่เข้าถึง แทนด้วย l.time

ถึงแม้ว่าจะมีฟิลด์อื่นๆอีกในล็อกเอ็นทรีของเวิลด์ไวด์เว็บแบบคอมมอน เช่น วิธีการร้องขอที่ใช้ (ได้แก่ POST หรือ GET) และขนาดของไฟล์ที่ถูกส่ง แต่ระบบจะให้ความสำคัญกับส่วนประกอบที่ได้กล่าวไปแล้วข้างต้นในล็อกเอ็นทรีเท่านั้น

นิยาม *2.1 แต่ละทรานแซคชัน t ประกอบด้วย 2 ส่วน คือ*

$$t = \langle uid_t, \{l_1^t.url, \ldots, l_k^t.url, \ldots, l_m^t.url\}\rangle$$

โดยที่ $l_k^t \in L$ *และ* $l_k^t.uid = uid_t$, $1 \le k \le m-1$ *และ* $l_{k+1}^t.time - l_k^t.time \le maxtimegap$

uid_t คือ รหัสประจำตัวผู้ชม $l_k^t.time$ คือ เวลาที่เริ่มเยี่ยมชมในครั้งที่ k maxtimegap คือ ช่วงเวลาระหว่างล็อกเอ็นทรี

จากล็อกเอ็นทรีในรูปที่ 1 ถ้าช่วงเวลาระหว่างล็อกเอ็นทรีที่ผู้ใช้กำหนดมีค่าเป็น 1 นาทีแล้ว ทรานแซคชันที่เป็นของไคลเอนท์ njain (เริ่มตั้งแต่เวลา 09/Aug/1996:09:53:52) จะมีเซตของ URL เป็น { / , /~adviser/ , /~adviser.csci-faq.html }

นิยาม *2.2 กำหนดให้ T เป็นเซตของทรานแซคชันความสัมพันธ์ทั้งหมดที่มีรูปแบบ $\langle uid_t, URL_t\rangle$ (ตาม นิยาม 2.1) โดยที่ $URL_t = \{ l_1^t.url, \ldots, l_m^t.url \}$ เรานิยามเว็บสเปซ,WS ของล็อกไฟล์ของการเยี่ยมชมเป็น $WS = \bigcup_{t \in T} URL_t$*

นิยาม *2.3 ให้ U เป็นเซตของเว็บเพจ ($U \subseteq WS$) แล้วเรานิยามค่าซัพพอร์ตเคานท์ (support count) ของ U คือ จำนวนทรานแซคชันของผู้ชมที่เยี่ยมชมเว็บเพจทั้งหมดใน U และใช้สัญลักษณ์แทนด้วย $\delta(U)$ เป็น $\delta(U) = | \{ t | U \subseteq URL_t \} |$ โดยที่ URL_t คือ เซตของเว็บเพจที่เยี่ยมชมในทรานแซคชัน t*

นิยาม *2.4 กฎความสัมพันธ์ คือนิพจน์ที่มีรูปแบบ $X \Rightarrow^{s,\alpha} Y$ โดยที่ $X \subseteq WS$ และ $Y \subseteq WS$ ค่าซัพพอร์ต (support) s ของกฎ $X \Rightarrow^{s,\alpha} Y$ เท่ากับ $\delta(XUY) / |T|$ และค่าความเชื่อมั่น (confidence) α เท่ากับ $\delta(XUY) / \delta(X)$ โดยที่ T คือ เซตของทรานแซคชันทั้งหมด*

งานในการค้นหากฎความสัมพันธ์คือการหากฎ $X \Rightarrow^{s,\alpha} Y$ ทั้งหมดโดยที่ s คือค่าซัพพอร์ต และ α คือค่าความเชื่อมั่น

ตัวอย่าง

{/company/products , /company/products/product1.html} $\Longrightarrow^{0.01,0.75}$
{/company/products/product2.html}

แสดงว่า 75 เปอรเซ็นต์ของไคลเอนท์ที่เยี่ยมชมส่วน "products" ของเว็บไซต์และลิงค์ต่อไปยังเพจ "product1" จะเยี่ยมชมเพจ "product2" ด้วยและการเกิดร่วมกันของเหตุการณ์นี้เกิดขึ้น 1 เปอร์เซ็นต์ของทรานแซคชันทั้งหมด

2.4 การเรียนรู้รูปแบบลำดับ[1]

รูปแบบลำดับคือรูปแบบลำดับการเยี่ยมชมเว็บเพจที่จัดเรียงตามเวลาที่ระบุภายในทรานแซคชัน โดยแต่ละทรานแซคชันจะประกอบไปด้วย URL และเวลาที่เยี่ยมชมของการเยี่ยมชมแต่ละครั้ง

ตัวอย่างเช่น ในการค้นหารูปแบบลำดับสามารถหาลำดับการเยี่ยมชมได้ดังนี้

- 30 เปอร์เซ็นต์ของไคลเอนท์ที่เยี่ยมชม /company/products/product1.html ได้ค้นหาโดยใช้คำสำคัญ w_1 และ w_2 ใน Yahoo ในสัปดาห์ที่ผ่านมา
- 60 เปอร์เซ็นต์ของไคลเอนท์ซึ่งสั่งซื้อสินค้าแบบออนไลน์ใน /company/products/product1.html แล้วยังทำสั่งซื้อสินค้าแบบออนไลน์ใน /company/products/product4.html ภายใน 15 วันด้วย

รูปแบบลำดับที่ค้นพบจะทำให้สามารถทำนายรูปแบบการเยี่ยมชมของผู้ชมได้และช่วยในการกำหนดกลุ่มเป้าหมายในงานโฆษณาจากกลุ่มผู้ชมได้อีกด้วย

สำหรับงานในการหารูปแบบลำดับต้องมีการเก็บข้อมูลเวลาในการเยี่ยมชมสำหรับแต่ละ URL ที่ถูกเยี่ยมชมไว้ภายในทรานแซคชัน นอกจากนี้ระบบยังสนใจพฤติกรรมของไคลเอนท์ที่เกิดขึ้นในล็อกไฟล์ของการเยี่ยมชม ดังนั้นเราจึงได้นิยามทรานแซคชันเป็นเซตที่มีสมาชิกเป็น URL และเวลาในการเยี่ยมชม URL นั้นเป็นของไคลเอนท์คนเดียวกันที่อยู่ภายในช่วงเวลาที่ผู้ใช้กำหนด

นิยาม *2.5 กำหนดให้แต่ละทรานแซคชันทางเวลาประกอบด้วย 2 ส่วน คือ*

$$t = \langle uid_t, UT_t \rangle$$

โดยที่ $UT_t = \langle (l_1^t.url , l_1^t.time) , \ldots , (l_m^t.url , l_m^t.time) \rangle$, $l_{k+1}^t.time - l_k^t.time \leq maxtimegap$ *และ* $1 \leq k \leq m-1$, $l_k^t \in L$ *และ* $l_k^t.uid = uid_t$ *โดยที่* uid_t *คือ รหัสประจำตัวผู้ชม* $l_i^t.time$ *คือเวลาที่เริ่มเยี่ยมชมในครั้งที่ i maxtimegap คือ ช่วงเวลาระหว่างล็อกเอ็นทรี*

ให้ T เป็นเซตของทรานแซคชันทางเวลาทั้งหมด สำหรับแต่ละทรานแซคชันทางเวลา $t=<uid_t, UT_t> \in T$ เรียก UT_t ว่า URL-time set (UT-Set) สำหรับ t ซึ่ง UT-Set คือเซตของ URL-time สำหรับเพจทั้งหมดที่ไคลเอนท์เยี่ยมชมภายในช่วงเวลาที่กำหนด เรายังนิยามเวลาของทรานแซคชันสำหรับ t แทนด้วย time(t) เป็น $time(t) = \max_{1 \le i \le m} l_i^t.time$

นิยาม *2.6 UT-sequence เป็นลิสต์ของ UT-Set ที่เรียงลำดับตามเวลาของทรานแซคชัน หรืออีกนัยหนึ่งคือ เมื่อกำหนดเซต* $T' = \{ t_i \in T \mid 1 \le i \le k \}$ *ของทรานแซคชัน แล้ว UT-sequence S สำหรับ T' คือ* $S=<UT_{t1}, \ldots, UT_{tk}>$ *โดยที่* $time(t_i) < time(t_{i+1})$ *สำหรับ* $1 \le i \le k-1$

นิยาม *2.7 กำหนดให้ไคลเอนท์ c มีค่า user id เท่ากับ u ให้* T_c *เป็นเซตของทรานแซคชันทางเวลาทั้งหมดที่เกี่ยวข้องกับไคลเอนท์ นั่นคือ*

$$T_c = \{ t \in T \mid uid_t = u \}$$

UT-sequence ของ T_c *เป็นลำดับพิเศษ แทนด้วย* S_c *เรียกว่าลำดับของลูกค้า (client-sequence) สำหรับไคลเอนท์ c ซึ่งประกอบด้วย UT-Set ทั้งหมดของทรานแซคชันที่เกี่ยวข้องกับไคลเอนท์ c หรืออีกนัยหนึ่งคือ* $S_c = <UT_{t1}, UT_{t2}, \ldots, UT_{tn}>$ *โดยที่* $1 \le i \le n$, $t_i \in T_c$

นิยาม *2.8 UT-sequence* $A=<a_1, a_2, \ldots, a_n>$ *เป็นลำดับย่อยของ UT-sequence* $B=<b_1, b_2, \ldots, b_n>$ *เขียนแทนด้วย* $A \subseteq B$ *ถ้ามีตัวเลขจำนวนเต็ม* $i1 < i2 < \ldots < in$ *ซึ่ง* $a_1 \subseteq b_{i1}$ $a_2 \subseteq b_{i2}$, ... , $a_n \subseteq b_{in}$

นิยาม *2.9 กำหนดให้ ID เป็นเซตของไคลเอนท์ทั้งหมดในล็อกไฟล์แล้วเรานิยามค่าซัพพอร์ตสำหรับ UT-sequence S แทนด้วย* $\delta(S)$ *เป็น* $\delta(S) = | \{S_c | c \in ID \text{ และ } S \in S_c\} |$ *ซึ่งหมายความว่าค่าซัพพอร์ตสำหรับ UT-sequence S คือ จำนวนของลำดับลูกค้าทั้งหมดที่มี S เป็นลำดับย่อย*

นิยาม *2.10 กำหนดให้ลำดับ S=X.Y เป็น UT-sequence (โดยที่ " . " แทนการต่อกันของลำดับ 2 ลำดับ) รูปแบบลำดับคือนิพจน์ที่อยู่ในรูปแบบของ* $X \overset{s,c}{\Rightarrow} Y$ *โดยที่ค่าซัพพอร์ต s ของลำดับมีค่าเท่ากับ* $\delta(X.Y)/|ID|$ *โดยที่ ID คือ เซตของไคลเอนท์ทั้งหมด และค่าความเชื่อมั่นของลำดับ c มีค่าเท่ากับ* $\delta(X.Y)/\delta(X)$

ในการทำเหมืองเว็บ การเรียนรู้รูปแบบลำดับคือการหาลำดับ $X \overset{s,c}{\Rightarrow} Y$ ทั้งหมด โดยที่ค่าซัพพอร์ต s ของลำดับจะต้องมีค่าไม่ต่ำกว่าค่าซัพพอร์ตขั้นต่ำที่กำหนด

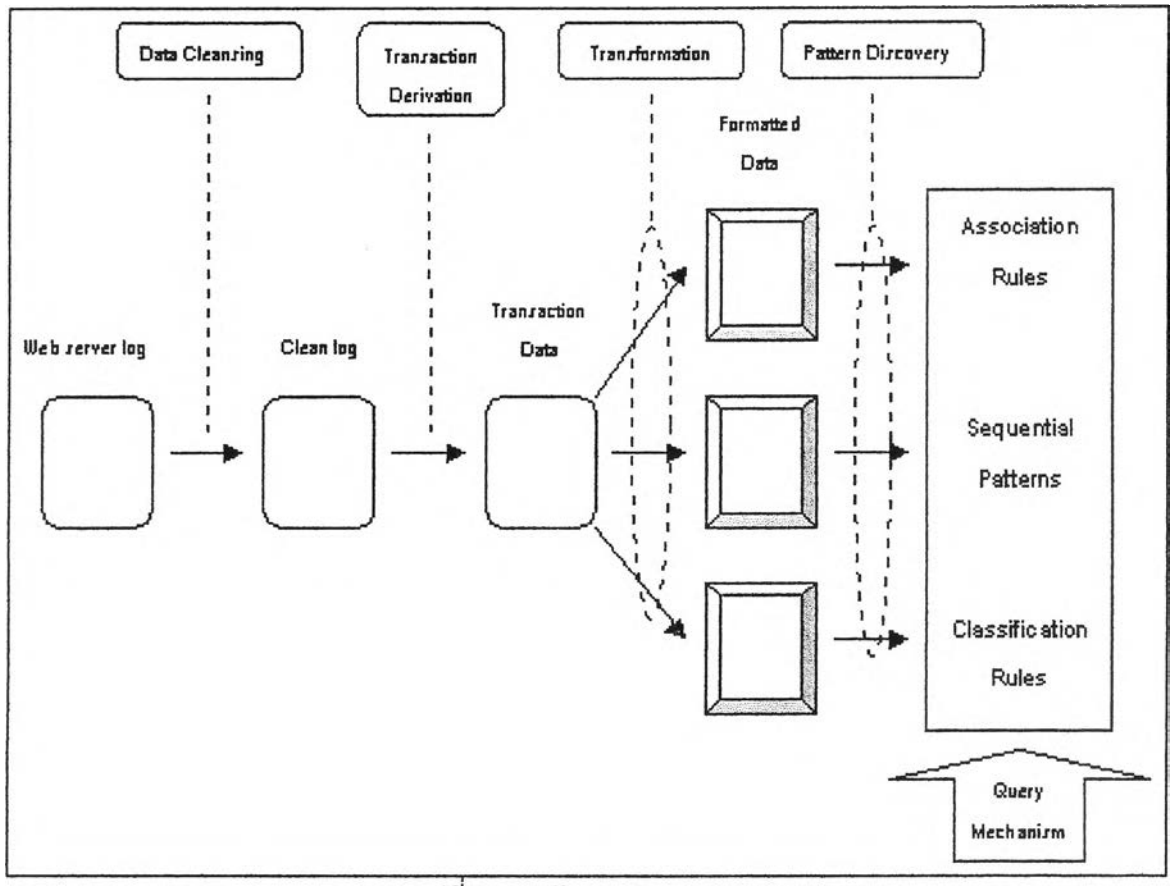
ถ้าค่าช่วงเวลาระหว่างล็อกเอ็นทรีของทรานแซคชันทางเวลามีค่าเป็น 0 แล้วสมาชิกของแต่ละ UT-sequence จะมีเพียงค่าเดียว (ซึ่งคือมีล็อกเอ็นทรีเพียงล็อกเอ็นทรีเดียวในแต่ละ UT-sequence) ถึงแม้ว่านิยามของทรานแซคชันทางเวลากำหนดให้แต่ละทรานแซคชันประกอบด้วยล็อกเอ็นทรีหลายล็อกเอ็นทรีแต่ในการประยุกต์ใช้งานส่วนมากเราสนใจเพียงแค่รูปแบบลำดับที่สร้างจาก URL เพียงค่าเดียวไม่ใช่เซตของ URL โดยที่ค่าช่วงเวลาระหว่างล็อกเอ็นทรีเป็น 0

2.5 งานวิจัย WEBMINER[2]

WEBMINER เป็น ระบบสำหรับค้นหารูปแบบจากเวิลด์ไวด์เว็บทรานแซคชันซึ่งพัฒนาโดย B. Mobasher , N. Jain , Eui-Hong (Sam) Han and, J. Srivastava

งานวิจัยนี้เกี่ยวข้องกับการพัฒนากรอบงานสำหรับการทำเหมืองข้อมูลการเยี่ยมชมเว็บซึ่งเป็นการนำเทคนิคการทำเหมืองข้อมูลมาประยุกต์ใช้เพื่อหาความสัมพันธ์ของเว็บเพจที่ถูกเยี่ยมชม ได้แก่การค้นหากฎความสัมพันธ์และรูปแบบลำดับการเยี่ยมชม กรอบงานนี้อธิบายถึงสถาปัตยกรรมสำหรับกระบวนการทำเหมืองเว็บซึ่งแบ่งแยกงานในการแปลงรูปแบบของข้อมูลออกจากแบบจำลองของข้อมูลและทรานแซคชัน โดยความรู้ที่ค้นพบสามารถนำไปใช้งานต่างๆได้ เช่น ใช้ในการปรับโครงสร้างของเว็บไซต์เพื่อเพิ่มประสิทธิผล ใช้สำหรับจัดการการสื่อสารของกลุ่มงานในอินทราเน็ตให้ดีขึ้นและใช้ในการวิเคราะห์รูปแบบการเยี่ยมชมของผู้ชมหรือใช้เพื่อนำเสนอข้อมูลแบบพลวัตให้เหมาะสมกับกลุ่มผู้ชม ซึ่งระบบ WEBMINER มีพื้นฐานอยู่บนกรอบงานนี้และยังอยู่บนพื้นฐานของแบบจำลองของข้อมูลเพื่อทำการค้นหากฎความสัมพันธ์,รูปแบบลำดับและกฎการจำแนกจากข้อมูลเวิลด์ไวด์เว็บอีกด้วย รูปที่ 3 แสดงสถาปัตยกรรมของ WEBMINER

ดังที่แสดงในรูปที่ 3 ก่อนที่กระบวนการค้นพบความรู้จะนำบันทึกการเข้าใช้ของเว็บเซิร์ฟเวอร์ (web server log) ไปประมวลผล บันทึกการเข้าใช้จะถูกส่งผ่านไปยังขั้นตอนการทำความสะอาดข้อมูลก่อนเพื่อกำจัดข้อมูลที่ไม่เกี่ยวข้องหรือซ้ำซ้อนออกไป ต่อจากนั้นจะนำบันทึกการเข้าใช้ที่ทำความสะอาดแล้ว (clean log) มาจัดรูปแบบให้อยู่ในรูปแบบที่เหมาะสมกับการประยุกต์ใช้งาน (กฎความสัมพันธ์และรูปแบบลำดับต้องการข้อมูลเข้าในรูปแบบที่แตกต่างกัน) และท้ายที่สุดจะนำข้อมูลที่จัดรูปแบบแล้วมาวิเคราะห์เพื่อหากฎความสัมพันธ์ระหว่างเว็บเพจ รูปแบบลำดับและกฎการจำแนก แล้วจึงนำไปเก็บลงในฐานความรู้เพื่อทำการแสดงผลต่อไป



รูปที่ 3 สถาปัตยกรรมของ WEBMINER

2.6 โปรแกรมวิเคราะห์บันทึกการเข้าใช้เว็บไซต์

โปรแกรมวิเคราะห์บันทึกการเข้าใช้เว็บไซต์ (web log analyzer) เป็นโปรแกรมที่ทำหน้าที่วิเคราะห์บันทึกการเข้าใช้เว็บเซิร์ฟเวอร์เพื่อรายงานสถิติของผู้ชมออกมาในรูปแบบ HTML ซึ่งสามารถดูได้โดยผ่านทางเว็บเบราเซอร์ เช่น โปรแกรม Webalizer[6]

ข้อมูลสถิติที่รายงานมีดังต่อไปนี้

- สถิติแยกตามวัน-เวลาเข้าชม เป็นสถิติแบบง่ายๆที่มาจากการบันทึกวัน-เวลาที่มีผู้ชมเข้ามาในเว็บเพจแต่ละครั้งเอาไว้ ตัวอย่างเช่น จำนวนผู้ชมในแต่ละวัน แต่ละสัปดาห์ แต่ละเดือน เป็นต้น
- สถิติแยกตามที่อยู่หรือที่มาหรือโดเมนเนมของผู้ชม นอกจากวัน-เวลาแล้วโปรแกรม Webalizer ยังแสดงโดเมนเนมหรือไอพีแอดเดรสของเครื่องผู้ชมได้ด้วย ซึ่งจะช่วยให้รู้แหล่งที่มาของผู้ชม เช่น ถ้ารู้ว่าโดเมนเนมของผู้ชมคือ something.co.th ก็สามารถเดาได้ว่าเป็นผู้ที่อยู่ในประเทศไทย จากนั้นก็อาจนำไปสรุปเป็นสถิติในลักษณะต่างอาทิที่อยู่ของผู้ชม 20 รายล่าสุดหรือจำนวนผู้ชมจากแต่ละทวีป ฯลฯ

- สถิติแยกตามเครื่องคอมพิวเตอร์ของผู้ชม โปรแกรมสามารถแสดงสถิติแยกตามชนิดของเบราเซอร์หรือระบบปฏิบัติการในเครื่องคอมพิวเตอร์ของผู้ชม เช่น Windows, MacOS, UNIX หรืออื่นๆ ข้อมูลเหล่านี้เป็นประโยชน์มากในการพัฒนาเว็บไซต์ของคุณให้เข้ากันได้ดีกับเครื่องของผู้ชมส่วนใหญ่ ยกตัวอย่างง่ายๆเช่นชื่อและรุ่นของเว็บเบราเซอร์ยอดนิยมในหมู่ผู้ชมของคุณ จะช่วยให้ตัดสินใจในการออกแบบเว็บเพจให้เข้ากับเว็บเบราเซอร์ที่มีผู้ชมใช้มากที่สุด และจะทำให้เขียนโค้ด HTML ได้ง่ายขึ้นอีกด้วย
- สถิติแยกตามแหล่งอ้างอิง (referrer) คำว่าแหล่งอ้างอิงหรือ referrer นั้นพูดง่ายๆก็คือ จุดหรือตำแหน่งต้นทางที่ผู้ชมเคยอยู่ ก่อนเข้ามาเยี่ยมชมเว็บเพจของเรา จุดหรือตำแหน่งต้นทางที่ว่านี้อาจจะเป็นเว็บเพจอื่น หรืออีเมลหรืออะไรก็ตามที่มีไฮเปอร์ลิงก์โยงมาถึงเว็บเพจของเรา แหล่งอ้างอิงนี้จะอยู่ในรูป URL(Universal Resource Locator) เช่น www.yourfriend.com/yourlink.html ซึ่งแทร็กเกอร์สามารถนำ URL ดังกล่าวไปวิเคราะห์แล้วตีความเป็นสถิติออกมาว่า มีผู้ชมกี่รายมาจากเว็บเพจนี้ หรือแม้แต่บอกว่าเว็บเพจที่เป็นแหล่งต้นทางนั้นเป็นเซิร์ชเอ็นจินหรือไม่ ถ้าเป็นเป็นเซิร์ชเอ็นจินใด และผู้ชมคนนั้นใช้คำอะไรเพื่อค้นหาในเซิร์ชเอ็นจินจนเจอและเข้ามาในเว็บเพจของเรา