ระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต

WEB SEARCH RESULTS CLUSTERING SYSTEM

โดย

พิศาล สรสิทธิ์

PISAN SORASIT

อาจารย์ที่ปรึกษา

ผศ.ดร. พรฤดี เนติโสภากุล

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2551

WEB SEARCH RESULTS CLUSTERING SYSTEM

PISAN SORASIT

A SYSTEM DEVELOPMENT PROJECT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2/ 2008

Title	Web Search Results Clustering System
Student	Mr. Pisan Sorasit
Student ID.	50066550
Degree	Master of Science
Programme	Information Technology Management
Academic Year	2008
Advisor	Asst.Prof. Dr.Ponrudee Netisopakul

ABSTRACT

Nowadays search engine is very popular because there are lots of internet data that have an increase of exponential growth rate. This increase also causes the increasing amount of search results. Because of a large amount of data, most of users usually not look at the data of searching result that close to the end. Thus, those useful data are usually overlooked and don't be used effectively. As a result, there has a paper represents about grouping data for searching according to contents of the text. This paper presents about Web Search Result Clustering System. It uses Suffix Tree Clustering (STC) Algorithm and LINGO Algorithm which is the way to gather and grouping the snippets from search engine. This technique will make efficient search result displaying and users can also get the data that relevant to their needs conveniencely and more rapidly.

กิตติกรรมประกาศ

โครงงานพัฒนาระบบงานนี้สำเร็จได้ด้วยคำแนะนำ และคำปรึกษาจาก ผศ.ดร. พรฤดี เนติ-โสภากุล ซึ่งเป็นอาจารย์ที่ปรึกษา ข้าพเจ้าขอขอบพระคุณอย่างยิ่งที่ท่านได้ให้ความอนุเคราะห์ด้วยดีเสมอมา จนกระทั่งพัฒนาโครงการนี้ให้สำเร็จลุล่วงไปได้ด้วยดี

ขอกราบขอบพระคุณอาจารย์คณะเทคโนโลยีสารสนเทศ และอาจารย์ต่างคณะ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่เข้ามาให้ความรู้กับข้าพเจ้า

ขอบคุณพี่ๆ เพื่อนๆ IS รุ่น23.2 ทุกคน เพื่อนๆ Com Sci รุ่น 15 พี่ๆ เพื่อนๆ ที่สำนักคอมพิวเตอร์ มหาวิทยาลัยมหิดล และทุกคนที่มีส่วนสนับสนุน ให้กำลังใจและเป็นที่ปรึกษาในทุกๆ เรื่อง

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจและให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี ข้าพเจ้าขอระลึกในพระคุณและขอกราบขอบพระคุณมา ณ ที่นี้

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

พิศาล สรสิทธิ์

สารบัญ

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

สารบัญตาราง(ต่อ)

สารบัญรูป

สารบัญรูป(ต่อ)

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ระบบสืบค้นข้อมูลบนอินเตอร์เน็ต (Search Engine) ปัจจุบันได้รับความนิยมเป็นอย่างสูง เนื่องจากข้อมูลในอินเตอร์เน็ตมีจำนวนมากและยังมีอัตราการเติบโตของข้อมูลที่เพิ่มสูงขึ้นเรื่อยๆ ถึงแม้ว่าการมีจำนวนข้อมูลที่มากนี้จะเป็นสิ่งดี แต่ก็ส่งผลให้การทำงานของระบบสืนค้นข้อมูลเกิดปัญหาขึ้น เช่น ปัญหาการเข้าถึงเนื้อหาที่ตรงตามความต้องการของผู้ใช้งาน เนื่องจากผลการสืบค้นที่ได้มานั้นมีจำนวนมาก อาจทำให้ข้อมูลที่ผู้ใช้งานต้องการใช้นั้นอาจต้องใช้เวลาในการหานานขึ้น หรือข้อมูลที่ได้มาอาจจะไม่ใช้ข้อมูลที่ตรงกับความต้องการทั้งหมด

จากปัญหาที่เกิดขึ้นนี้จึงได้มีแนวคิดที่ต้องการจัดกลุ่มข้อมูลผลการสืบค้นให้อยู่ในรูปแบบกลุ่มผลการสืบค้นตามเนื้อหาที่ผลลัพธ์นั้นๆ เกี่ยวข้อง เพื่อช่วยให้ผู้ใช้ระบบสามารถเข้าไปดูผลสืบค้นข้อมูลได้ตามหมวดหมู่ที่ตนสนใจ ซึ่งจะช่วยให้การแสดงผลการสืบค้นมีประสิทธิภาพมากยิ่งขึ้น ทำให้ช่วยประหยัดเวลาของผู้สืบค้นข้อมูล โดยเลือกจากกลุ่มที่ของเนื้อหาที่สนใจแทนที่จะเลือกสืบค้นที่ละรายการ ทั้งยังเพิ่มโอกาสให้ผลการสืบค้นข้อมูลที่อยู่ในลำดับท้ายถูกนำมาใช้ เพราะถูกนำมาแสดงในกลุ่มเดียวกับผลลัพธ์อื่นๆที่มีเนื้อหาด้านเดียวกัน

การจัดกลุ่มของข้อมูลนั้นสิ่งหนึ่งที่คำนึงถึงเป็นอย่างยิ่งคือป้ายชื่อของกล่มข้อมูล (Label) ที่จัดกลุ่มออกมาได้นั้นสามารถสื่อความหมายให้กับผู้ใช้ได้เพียงใดรวมทั้งเรื่องความเร็วในการจัดกลุ่มและแสดงผลลัพธ์ต่อผู้ใช้ ทั้งหมดนี้จึงเป็นปัจจัยสำคัญที่ช่วยบอกถึงคุณภาพของการจัดกลุ่มข้อมูลได้ ในโครงงานพัฒนาระบบนี้จะนำอัลกอริทึมซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO) เข้ามาประยุกต์ใช้ โดยจะพิจารณาผลลัพธ์ที่ได้จากทั้งสองอัลกอริทึมว่ามีความแตกต่างกันด้านใดและพิจารณาถึงข้อเด่นและข้อด้อยของทั้งสองด้วย

1.2 วัตถุประสงค์

1. เพื่อศึกษาลักษณะการทำงานและการแสดงผลของระบบสืบค้นข้อมูลบนอินเตอร์เน็ต เพื่อให้เข้าใจและทราบถึงจุดเด่น – จุดด้อยของระบบที่มีอยู่ในปัจจุบัน เพื่อหาแนวทางการแก้ปัญหาและการพัฒนาให้มีประสิทธิภาพยิ่งขึ้น

2. เพื่อศึกษากระบวนการทำงานของการจัดกลุ่มข้อมูลด้วยอัลกอริทึมซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO) เพื่อนำมาประยุกต์ใช้กับการจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ตได้อย่างมีประสิทธิภาพ

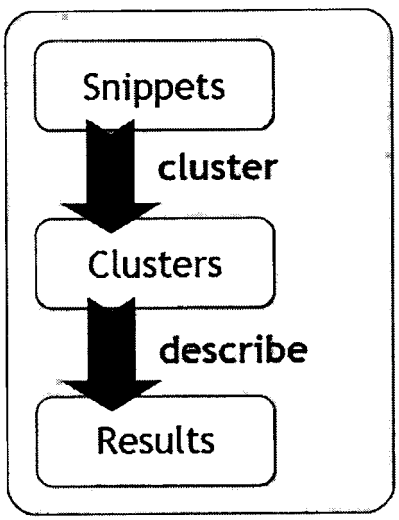
3. เพื่อพัฒนาระบบสืบค้นข้อมูลและการแสดงผลให้มีประสิทธิภาพมากขึ้น โดยการจัดกลุ่มผลการสืบค้นข้อมูล เพื่อต้องการทำให้ผู้ใช้งานเข้าถึงข้อมูลที่ตรงกับความต้องการได้สะดวกและรวดเร็วยิ่งขึ้น

4. เพื่อวิเคราะห์ข้อเด่น – ข้อด้อยของการจัดกลุ่มผลการสืบค้นจากระบบสืบค้นข้อมูลบนอินเตอร์เน็ต โดยใช้อัลกอริทึมซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO)

1.3 ทฤษฎีที่อ้างอิง

ทฤษฎีที่ใช้อ้างอิงในการจัดกลุ่มข้อมูลในระบบนี้มีอยู่ด้วยกัน 2 ทฤษฎีคือ

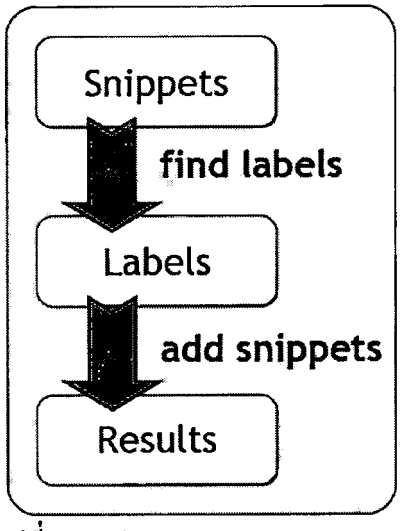
1. ซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) (Zeng, H. et.al. 2004.) เป็นอัลกอริทึมการจัดกลุ่มข้อมูลแบบแฟลทคลัสเตอร์ริง (Flat Clustering) โดยจะทำงานร่วมกับโครงสร้างข้อมูลแบบซับฟิกทรี (Suffix Tree) (R. Baeza-Yates and B. Ribeiro-Neto. 1999) และจะใช้เวลาในการประมวลผลแปรผันไปกับจำนวนเอกสารทั้งหมดที่ถูกรวบรวมไว้ในเอกสารฉบับนี้ หมายถึงคำอธิบายโดยสรุป (snippets) โดยกระบวนการจัดกลุ่มจะพิจารณาจากการใช้กลุ่มคำหรือวลี (phrase) ร่วมกันของเอกสาร STC จะใช้วลีเป็นตัวแทนของเอกสาร โดยที่วลีในที่นี้จะหมายถึงลำดับของคำที่ต่อเนื่องกันตั้งแต่ 1 คำขึ้นไปที่ปรากฏในเอกสาร ซึ่งป้ายชื่อของกลุ่ม (Label) จะได้มาจากวลีที่ถูกพบบ่อยในเอกสารทั้งหมดที่มี รูปที่ 1.1 แสดงหลักการทำงานของ STC



รูปที่ 1.1หลักการทำงานของ STC

2. ลินโก (LINGO) (Stanislaw Osilski. 2003) เป็นอัลกอริทึมการจัดกลุ่มข้อมูลอีกแบบหนึ่งซึ่งจัดกลุ่มข้อมูลโดยการหาป้ายชื่อของกลุ่ม (Label) ด้วยวิธีการตัดวลีที่พบบ่อยๆ ในกลุ่มเอกสารที่มีอยู่และแยกออกมาเข้ากระบวนการของ Single Value Decomposition (SVD) เพื่อที่จะหา

ป้ายชื่อของกลุ่มของข้อมูล จากนั้นจะนำป้ายชื่อของกลุ่มของข้อมูลที่ได้หามาเอกสารที่สัมพันธ์กับชื่อของกลุ่มของข้อมูล โดยลินโกจะให้ความสำคัญกับป้ายชื่อกลุ่มที่ได้ (Description comes first clustering) คือป้ายชื่อของกลุ่มที่ได้นั้นสามารถสื่อความหมายให้กับผู้ใช้ระบบเข้าใจได้เพียงใด รูปที่ 1.2 แสดงหลักการทำงานของ LINGO



รูปที่ 1.2 หลักการทำงานของ LINGO

1.4 ขอบเขตของระบบ

1. จัดกลุ่มข้อมูลซึ่งได้มาจากผลการสืบค้นข้อมูลส่งมาจากระบบสืบค้นข้อมูลโดยการนำอัลกอริทึมซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO) มาประยุกต์ใช้ในระบบ

2. ข้อมูลที่นำมาใช้ในระบบนั้น จะใช้เพียงคำอธิบายโดยสรุป (snippets) ที่ได้รับจากระบบสืบค้นข้อมูลมาทำการจำแนกกลุ่มเท่านั้น

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เข้าใจหลักการทำงานและการแสดงผลของระบบผลการสืบค้นข้อมูลบนอินเตอร์เน็ต (Search Engine)

2. เข้าใจหลักการและขั้นตอนการทำงานของอัลกอริทึมซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO)

3. เปรียบเทียบข้อเด่น – ข้อด้อยของการจัดกลุ่มผลการสืบค้น โดยใช้อัลกอริทึมซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO)

4. ได้ต้นแบบระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ตเพื่อให้การสืบค้นข้อมูลมีความสะดวก รวดเร็วและตรงกับความต้องการมากขึ้น ทั้งยังทำให้ผลการสืนค้นข้อมูลถูกนำมาใช้ประโยชน์สูงสุด

1.6 ขั้นตอนของการศึกษา

เอกสารฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

1. บทที่ 1 กล่าวถึงความเป็นมาของโครงงาน วัตถุประสงค์ ทฤษฎีที่ใช้ ขอบเขตของโครงงาน ประโยชน์ที่ได้รับและขั้นตอนการศึกษา
2. บทที่ 2 กล่าวถึงทฤษฎีที่เกี่ยวข้องกับการพัฒนาระบบงาน ซึ่งประกอบด้วยหลักการทำงานของระบบสืบค้นข้อมูลบนอินเตอร์เน็ตและหลักการของการจัดกลุ่มเอกสาร รวมทั้งอัลกอริทึมที่ใช้ในการพัฒนาโครงงานนี้ คือซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO)
3. บทที่ 3 กล่าวถึงการวิเคราะห์และออกแบบระบบ
4. บทที่ 4 กล่าวถึงหน้าจอและลักษณะการทำงานของระบบ
5. บทที่ 5 กล่างถึงการสรุปผลและข้อเสนอแนะ โดยจะแสดงผลการเปรียบเทียบในด้านต่างๆ เช่น ผลลัพธ์จากการจัดกลุ่มข้อมูลของทั้งสองอัลกอริทึม เวลาที่ใช้ประมวลผลในการจัดกลุ่มข้อมูล และความถูกต้องของผลลัพธ์ที่ได้จากการจัดกลุ่ม

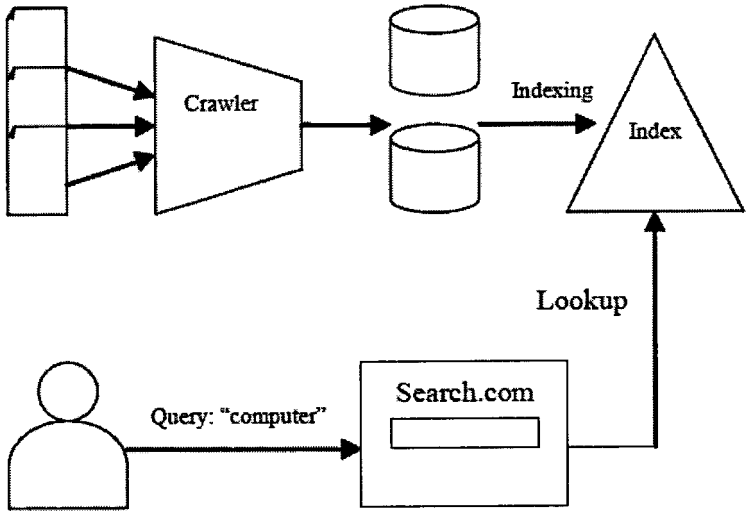
บทที่ 2
ทฤษฎีพื้นฐานที่เกี่ยวข้อง

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานที่เกี่ยวข้องและหลักการทำงานของนำอัลกอริทึมซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO) ซึ่งเนื้อหาในบทนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ของระบบสืบค้นข้อมูลบนอินเตอร์เน็ต (Search Engine) และเทคนิคการจัดกลุ่มข้อมูล (Clustering Method) ได้แก่ กระบวนการทำงาน ประเภทของระบบสืบค้นข้อมูลและเทคนิคการจัดกลุ่มข้อมูลแต่ละประเภท รวมทั้งหลักการทำงานและขั้นตอนการจัดกลุ่มข้อมูลด้วย STC และ LINGO ซึ่งเนื้อหาทั้งหมดนี้จำเป็นสำหรับการพัฒนาระบบงานนี้

2.1 ระบบสืบค้นข้อมูลบนอินเตอร์เน็ต (Search Engines)

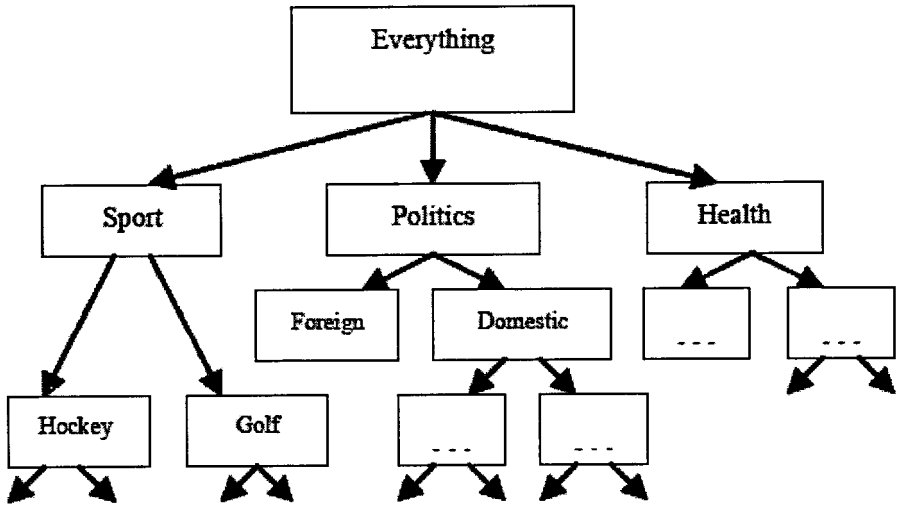
ระบบสืบค้นข้อมูลบนอินเตอร์เน็ต (Search Engine) ในปัจจุบันมีอยู่เป็นจำนวนมาก เช่น Google, Yahoo, MSN, Lycos, DMOZ เป็นต้น โดยสามารถแบ่งได้ 3 ประเภท ซึ่งแต่ละประเภทมีหลักการทำงานที่ต่างกัน และการจัดอันดับการค้นหาข้อมูลก็ต่างกัน (วิริศ ลีลาภัทร, พรฤดี เนติโสภากุล. 2548)

2.1.1 Indexing Search Engine (keyword search) เสิร์จเอ็นจิน เป็นระบบซอฟต์แวร์ที่มีโปรแกรม โรบอต (Robot) ซึ่งบางครั้งเรียกว่า สไปเดอร์ (Spider) หรือ ครอว์เลอร์ (Crawler) ทำหน้าที่เดินทางไปยังเว็บไซต์ต่าง ๆ ในอินเตอร์เน็ตและอ่านเว็บเพจจากไซต์เหล่านั้นเพื่อนำมาสร้างดัชนีรายการของเว็บเพจโดยอัตโนมัติ การทำดัชนีรายการเว็บเพจด้วยเสิร์จเอ็นจินจึงสามารถสร้างดัชนีของเว็บเพจได้เป็นจำนวนที่มากกว่าของไดเรคทอรี เนื่องจากดัชนีของไดเรคทอรีจะถูกจัดการโดยมนุษย์แทนที่จะใช้คอมพิวเตอร์ ถ้าหากเว็บเพจที่ถูกทำดัชนีแล้วเกิดการเปลี่ยนแปลงเสริจ์เอ็นจินจะทราบถึงการเปลี่ยนแปลงและจัดนำเว็บเพจที่มีการเปลี่ยนแปลงนั้นมาสร้างดัชนีใหม่ ซึ่งการสร้างดัชนีใหม่นี้อาจส่งผลต่อการจัดลำดับของเว็บเพจนั้ ตัวอย่างของเสริจ์เอ็นจิน ที่มีอยู่ในปัจจุบัน เช่น HotBot, AltaVista เป็นต้น รูปที่ 2.1 แสดงถึงการสถาปัตยกรรมของ Indexing Search Engine



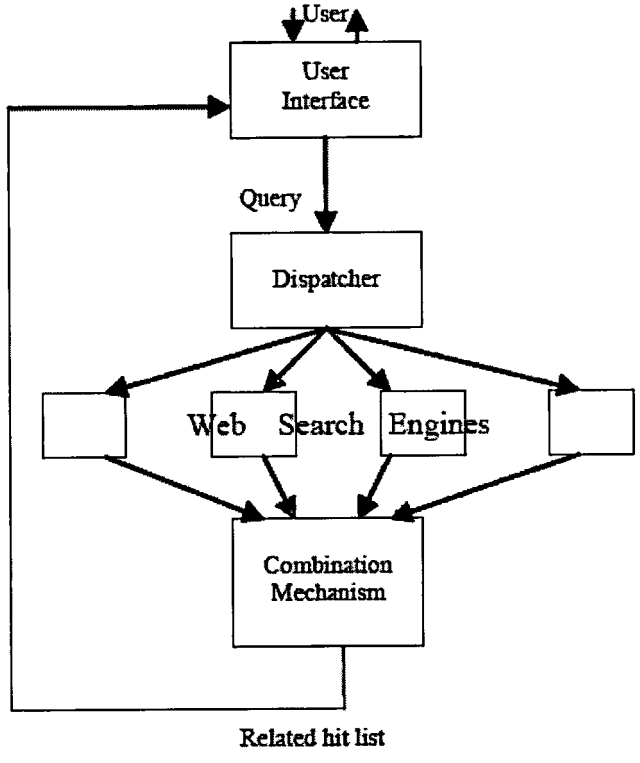
รูปที่ 2. 1 สถาปัตยกรรมของ Indexing Search Engine

2.1.2 Subject Directories มีลักษณะเป็นรายการของเว็บไซต์ หรือ เว็บเพจ ซึ่งได้มีการจัดรวบรวมไว้โดยการแบ่งเป็นหมวดหมู่ตามลักษณะและหัวข้อเนื้อหา เพื่อให้ผู้ค้นหาสามารถเลือกค้นได้ในแต่ละหมวดหมู่หัวข้อเนื้อหา (Subject Categories) จะถูกแบ่งเป็นหมวดหมู่ย่อย (Sub-Categories) ที่จัดเรียงลดหลั่นกันหลายระดับจนกระทั่งถึงรายการชื่อเว็บไซต์ที่นำเสนอเนื้อหาที่สอดคล้องกับหมวดหมู่ย่อยนั้นพร้อมกับมีตัวเชื่อมโยง (Link) เพื่อชี้ไปยังเว็บไซต์นั้น กระบวนการสร้างและจัดทำผ่านการดำเนินการโดยบุคคลหรือกลุ่มบุคคลที่ทำหน้าที่ในการจัดการหมวดหมู่หัวข้อเนื้อหา เลือกและจำแนกเว็บไซต์ลงในหมวดหมู่ รูปที่ 2.2 แสดงถึงการสถาปัตยกรรมของ Subject Directories



รูปที่ 2. 2 สถาปัตยกรรมของ Subject Directories

2.1.3 Multi-Engine Search Tool หรือ Meta-Search คือ Search Engine ที่สามารถสืบค้นข้อมูลจาก Search Engine และหรือ Web Directories ได้มากกว่า 1 ตัวในเวลาเดียวกัน และแสดงผลการสืบค้นที่ได้รับจาก Search Engine เหล่านั้นในเวลาเดียวกัน โดยเสนอผลการสืบค้นในรูปแบบที่สะดวก ซึ่งบางครั้งจะมีการปรับแต่งผลการสืบค้นเหล่านี้เข้ามาเป็นชุดเดียวกัน รูปที่ 2.3 แสดงถึงการสถาปัตยกรรมของ Meta Search Engine คือ Search Engine



รูปที่ 2. 3 สถาปัตยกรรมของ Meta Search Engine คือ Search Engine

แม้ว่า Search Engine แต่ละประเภทจะช่วยในการค้นหาข้อมูลในอินเตอร์เน็ตได้เป็นอย่างดี แต่ปัญหาและข้อจำกัดของ Search Engine แต่ละประเภทก็ยังมีอยู่ เช่น

การชี้ไปยังเอกสารที่ยังไม่มีการปรับปรุงข้อมูลหรือไม่มีข้อมูลตามที่ระบุไว้ (Dead Link)

1. Subject หรือ Directory จะพบกับปัญหาเรื่องปริมาณสาระสนเทศที่เก็บรวบรวมไว้มีจำนวนไม่มากนัก รวมถึงระยะเวลาการจัดเก็บข้อมูลแป็นเวลานานอันเป็นผลมาจากการต้องใช้คนในการตรวจนสอบและจัดเก็บ

2. Indexing Search Engine หรือ Keyword Search จะให้จำนวนผลลัพธ์หรือผลการสืบค้นมากเกินไป ขาดการประเมินและกลั่นกรองสาระของเว็บเพจที่ได้เก็บรวบรวมมา

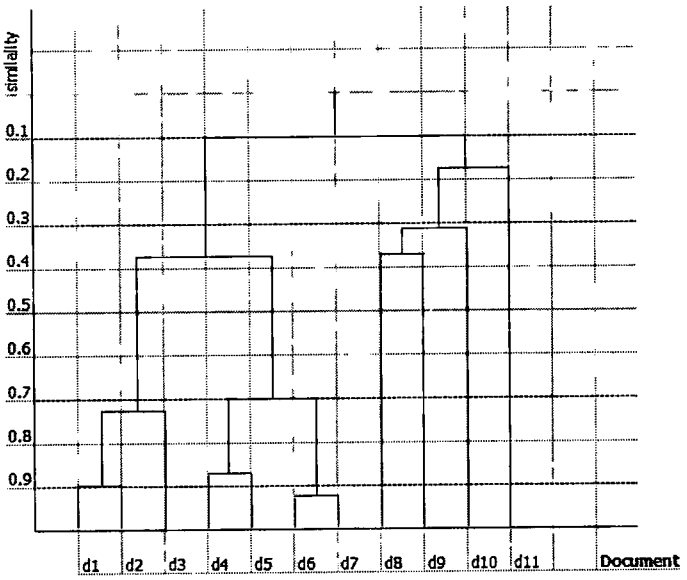
3. ระบบสืบค้นให้ลำดับรายการยาวๆ ทำให้ผู้สืบค้นไม่ได้รับความสะดวกในการเข้าถึงข้อมูลที่ตรงกับความต้องการและเสียเวลาในการค้นหาข้อมูลที่ต้องการ เช่นผู้สืบค้นต้องการสืบค้น

คำว่า "Jaguar" ในความหมายที่สัมพันธ์กับ "เสือ" ผู้สืบค้นอาจจะต้องไปหาในลำดับรายการที่ 10, 11, 32 และ 72 เป็นต้น

2.2 วิธีการจัดกลุ่มข้อมูล (Clustering Methods)

การจัดกลุ่มข้อมูล (Clustering) มีวัตถุประสงค์หลักเพื่อแยกข้อมูลเป็นกลุ่มย่อยๆ ตามลักษณะความเหมือนกันของข้อมูล โดยข้อมูลที่เหมือนกันจะถูกจัดให้อยู่ในกลุ่มข้อมูลเดียวกันซึ่งข้อมูลที่อยู่ในกลุ่มข้อมูลเดียวกันจะมีค่าเหมือนกันมากกว่าข้อมูลที่อยู่ต่างกลุ่มกัน เทคนิคการจัดกลุ่มได้นำไปใช้กับวัตถุที่มีค่าของคุณสมบัติที่เป็นค่าเชิงตัวเลขอย่างแพร่หลาย ซึ่งสามารถจัดกลุ่มเชิงตัวเลขที่เป็นอย่างดี ปัจจุบันได้เริ่มนำเทคนิคการจัดกลุ่มมาใช้กับวัตถุที่มีค่าของคุณสมบัติเป็นค่าเชิงตัวอักษรเพิ่มมากขึ้น การจัดกลุ่มผลการสืบค้นข้อมูลของ Search Engine ในรูปแบบของ Web Search Result Clustering ก็เป็นส่วนหนึ่งของการจัดกลุ่มด้วยค่าคุณสมบัติเป็นเชิงตัวอักษร ซึ่งมีเทคนิคการจัดกลุ่ม 2 รูปแบบดังนี้

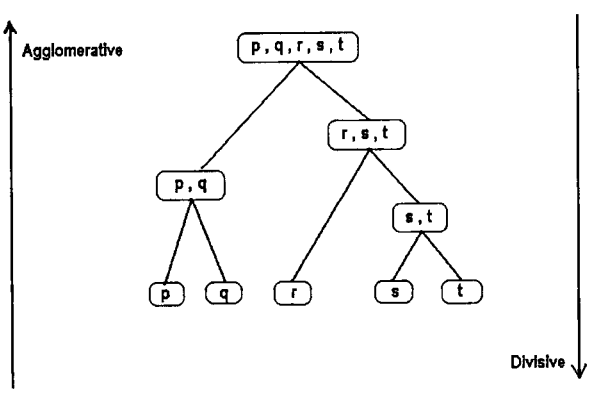
2.2.1 Hierarchical Clustering (Stanislaw Osilski, Dawis Wiess. 2005) มีลักษณะการจัดกลุ่มเป็นลำดับชั้น โดยเริ่มจากข้อมูลทั้งหมดจะอยู่เพียงคลัสเตอร์เดียวในระดับบนสุด แล้วเมื่อผ่านขั้นตอนการจัดกลุ่มข้อมูลจะได้คลัสเตอร์ย่อยๆ จนกระทั่งได้คลัสเตอร์ที่มีข้อมูลเพียงชุดเดียวที่ระดับล่างสุด ในขั้นตอนสุดท้ายของการทำงานของอัลกอริทึมจะได้โครงสร้างต้นไม้ของกลุ่มข้อมูลซึ่งแสดงถึงความสัมพันธ์ของกลุ่มข้อมูลว่ามีความสัมพันธ์กันอย่างไร โดยถ้าทำการตัดโครงสร้างต้นไม้ในระดับที่ต้องการข้อมูลก็จะแยกจากกันเป็นกลุ่มๆ Hierarchical Clustering มีการแสดงผลลัพธ์ด้วยแผนภาพโครงสร้างต้นไม้ หรือที่เรียกอีกอย่างหนึ่งว่า "dendrogram" ดังแสดงในรูปที่ 2. 4 ซึ่งช่วยสร้างความเข้าใจได้ง่ายขึ้น โครงสร้างของ dendrogram จะประกอบด้วยชั้นของ node แสดงถึงการจัดกลุ่มในชั้นนั้นๆ ในแต่ละคลัสเตอร์เส้นที่เชื่อมระหว่าง node แสดงถึงการรวมกันของคลัสเตอร์ใหม่อีกคลัสเตอร์หนึ่ง และถ้าเราตัดแผนภาพ dendrogram ตามขวางในแต่ละระดับชั้นเราจะได้ผลของการทำ clustering ในระดับชั้นนั้นๆ



รูปที่ 2. 4 การจัดกลุ่มแบบ Hierarchical Clustering

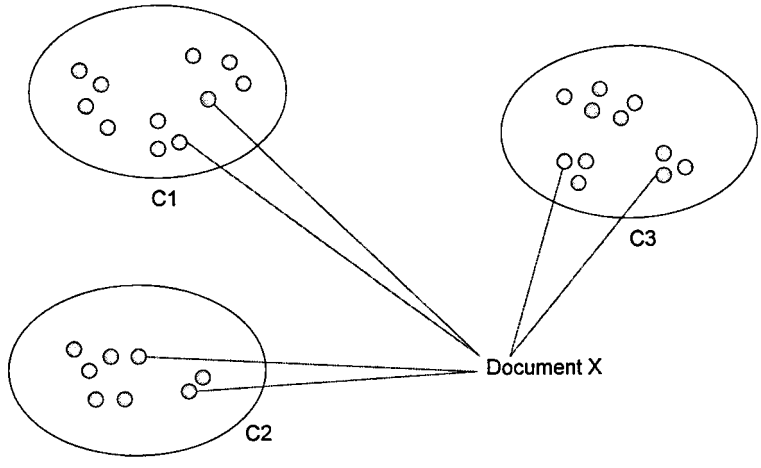
จากรูปที่ 2. 4 ถ้าจัดกลุ่มที่ค่า similarity = 0.1 เอกสารมี 2 กลุ่มคือ {d1, d2, d3, d4, d5, d6, d7}, {d8, d9, d10, d11} และถ้าจัดกลุ่มที่ค่า similarity = 0.7 จะได้เอกสาร 6 กลุ่มคือ {d1, d2, d3}, {d4, d5, d6, d7}, {d8}, {d9}, {d10}, {d11} เป็นต้น

วิธีที่นิยมใช้กันมากที่สุดของ Hierarchical Clustering คือ วิธี agglomerative จะเริ่มจากข้อมูลที่อยู่ต่างคลัสเตอร์และในแต่ละขั้นตอนการทำงานจะรวมข้อมูลที่เหมือนกันหรือคล้ายคลึงกันให้อยู่ในคลัสเตอร์เดียวกัน ทำซ้ำไปเรื่อยๆจนกระทั่งจำนวนของคลัสเตอร์มีค่าน้อยที่สุด และวิธี divisive วิธีนี้จะเริ่มจากรวมข้อมูลทั้งหมดให้อยู่ในคลัสเตอร์เดียว แล้วทำการพิจารณาว่าคลัสเตอร์ใดควรจะถูกแยกออกมาและจะแยกมันออกมาด้วยวิธีใด ซึ่งเป็นสิ่งที่สำคัญที่สุดในกระบวนการทำงานของ divisive ดังแสดงในรูปที่ 2. 5



รูปที่ 2. 5 แผนภาแสดงประเภทของ Hierarchical Clustering

2.2.2 Flat Clustering (Stanislaw Osilski, Dawis Wiess. 2005) เป็นลักษณะการจัดกลุ่มที่ตรงข้ามกับแบบ Hierarchical Clustering เพราะ Flat Clustering จะเป็นลักษณะค่อยๆจัดกลุ่มทีละเอกสารที่เข้ามาตามคุณลักษณะว่าเอกสารนั้นควรจะอยู่ในกลุ่มใด ถ้าไม่มีคุณลักษณะที่ตรงกับกลุ่มที่มีอยู่เดิมก็จะทำการสร้างกลุ่มใหม่ ดังแสดงในรูปที่ 2. 6



รูปที่ 2. 6 แผนภาพ Flat Clustering

จากรูปที่ 2. 6 เมื่อเอกสาร x ถูกนำเข้าระบบเพื่อจัดกลุ่ม จะมีการคำนวณค่า similarity ที่ใกล้เคียงกับแต่ละกลุ่ม คือ C1, C2, C3 ค่าความเหมือนในกลุ่มใดมากที่สุดก็จะนำเอกสาร x เข้าไปอยู่เป็นสมาชิกในกลุ่มนั้น แต่ในกรณีที่การคำนวณค่า similarity ของเอกสาร x กับกลุ่มเอกสารเดิมที่มีอยู่ ปรากฏว่าเอกสาร x ไม่เหมือนหรือคล้ายกับกลุ่มใดเลย เราจะต้องทำการสร้างกลุ่มขึ้นมาใหม่โดยมีเอกสาร x เป็นสมาชิกภายในกลุ่ม

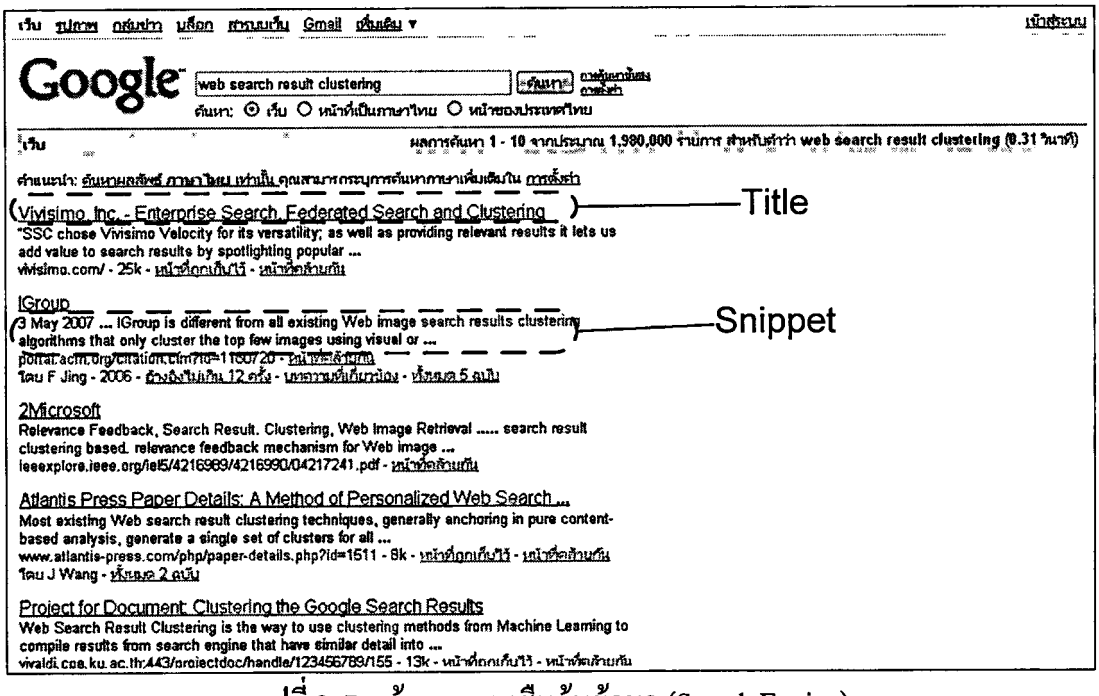
ปัจจุบันการจัดกลุ่มได้ถูกพัฒนาขึ้นมาด้วยหลายๆรูปแบบและวิธีการในการจัดกลุ่ม เพื่อให้ผลการจัดกลุ่มมีความยืดหยุ่นและถูกต้องเหมาะสมมากที่สุด เช่น การนำกระบวนการเรียนรู้ของโครงข่ายประสาทเทียม เข้ามามีส่วนร่วมในการพัฒนาการจัดแบ่งกลุ่มข้อมูล ซึ่งมีผลดีคือสามารถจัดแบ่งกลุ่มของข้อมูลที่มีการกระจายของกลุ่มข้อมูลจำนวนมาก แต่มีข้อมูลจำนวนน้อยๆได้ เช่นวิธี Self-Organizing Maps Clustering หรือการนำเอาคุณสมบัติของ Fuzzy set มาใช้ในการจัดแบ่งกลุ่ม ซึ่ง Fuzzy set จะเพิ่มข้อมูลในส่วนของค่าความเป็นสมาชิก (degree of membership) เข้ามา เช่นวิธี fuzzy C-mean clustering

2.3 การจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต (Web Search Results Clustering)

เนื่องจากผู้สืบค้นได้รับผลการสืบค้นจากระบบสืบค้นบนอินเตอร์เน็ตเป็นรายการยาวๆ ทำให้ผู้สืบค้นต้องเสียเวลาเพื่อค้นหาข้อมูลที่ตรงกับความต้องการอีกครั้งหนึ่ง หรือคำที่ค้นหานั้นมี

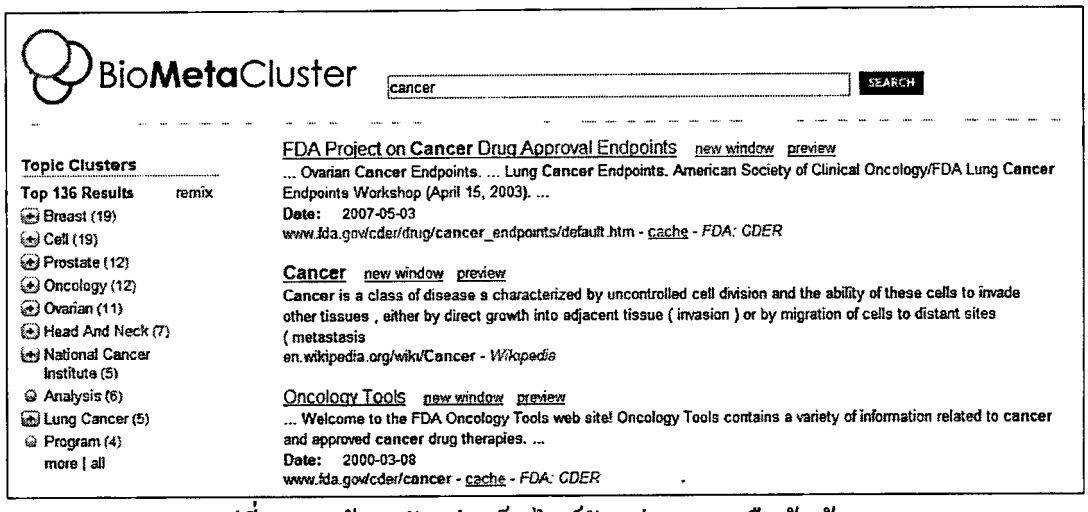
หลายความหมาย เช่น ผู้สืบค้นต้องการค้นหาคำว่า "Jaguar" ในความหมายที่เกี่ยวข้องกับสัตว์ แต่ระบบสืบค้นทั่วๆไปอาจให้ผลการสืบค้นตามที่ผู้สืบค้นต้องการในลำดับที่ 10, 11, 32 หรือ 71 ของลำดับรายการค้นหาที่ได้มา จากตัวอย่างใน

รูปที่ 2. 7 เป็นผลการสืบค้นคำว่า "Web Search Results Clustering" จะเรียงลำดับความสำคัญของเอกสาร นั่นคือผลที่ได้มามีจำนวนมากเกินความต้องการของผู้สืบค้น ส่งผลให้ผู้สืบค้นต้องทำการค้นหาด้วยตนเองอีกครั้งหนึ่งว่าเอกสารที่ตนเองต้องการนั้นอยู่ในลำดับที่เท่าไร โดยดูจาก title และ snippets ที่ระบบจะแสดงให้ จากตัวอย่างเป็นระบบสืบค้นของ google.com ให้ผลการสืบค้นทั้งหมด 1,980,000 รายการ



รูปที่ 2. 7 หน้าจอระบบสืบค้นข้อมูล (Search Engine)

จากปัญหาดังกล่าวทำให้มีการนำผลการสืบค้นมาจัดกลุ่ม เพื่ออำนวยความสะดวกให้กับผู้สืบค้น ค้นหาเอกสารที่ต้องการตามกลุ่มต่างๆ และสามารถเข้าถึงข้อมูลที่ต้องการได้สะดวกรวดเร็วมากยิ่งขึ้น ดังแสดงในรูปที่ 2. 8 เป็นการจัดกลุ่มผลการสืบค้นของระบบสืบค้นชื่อ vivisimo.com ผู้ค้นหาสามารถเลือกค้นหาในกลุ่มที่ต้องการได้



รูปที่ 2. 8 หน้าจอตัวอย่างเว็บไซต์จัดกลุ่มผลการสืบค้นข้อมูล

ลักษณะการจัดกลุ่มผลการสืบค้นแบ่งเป็น 4 รูปแบบ คือ

2.3.1 Single Word and Flat Clustering (Stanislaw Osilski, Dawis Wiess. 2005) เป็นการใช้ Single Word เป็นตัวกำหนด feature ของเอกสาร (ใช้คำคำเดียวเป็นตัวแทนของเอกสาร) เพื่อนำไปจัดกลุ่มตามรูปแบบของ Flat Clustering

2.3.2 Sentence and Flat Clustering (Stanislaw Osilski, Dawis Wiess. 2005) เป็นการใช้ Sentence เป็นตัวกำหนด feature ของเอกสาร (ใช้ sentence เป็นตัวแทนเอกสาร) เพื่อนำไปจัดกลุ่มตามรูปแบบของ Flat Clustering

2.3.3 Single Word and Hierarchical Clustering (Stanislaw Osilski, Dawis Wiess. 2005) เป็นการใช้ Single Word เป็นตัวกำหนด feature ของเอกสารเพื่อนำไปจัดกลุ่มตามรูปแบบของ Hierarchical Clustering เช่น ระบบของ Frequent Item Hierarchical Clustering (FIHC) ใช้ความถี่ของคำ (vector model) ในการจัดกลุ่ม

2.3.4 Sentence and Hierarchical Clustering (Stanislaw Osilski, Dawis Wiess. 2005) ใช้ Sentence เป็นตัวกำหนด feature ของเอกสาร เพื่อจัดกลุ่มตามแบบของ Hierarchical Clustering เช่น ระบบ SNAKET นำเสนอแนวทางากรพัฒนาการจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต โดยชื่อว่า Semantic, Hierarchical, Online Clustering (SHOC) ซึ่งมีการนำ Suffix Array ซึ่งเป็นโครงสร้างข้อมูลรูปแบบหนึ่งมาใช้ในการค้นหา phrase แล้วนำ phrase เป็นป้ายชื่อ (Label) มาจัดกลุ่มด้วย Singular Value Decomposition (SVD) ร่วมกับ Latent Semantic Indexing (LSI)

ปัจจุบันมีหลายๆระบบที่พัฒนา Web-Snippets Clustering ในลักษณะของ (meta)search engine ประกอบด้วย vivisimo.com, clusty.com, kartoon.com, mooter.com, copernic.com, iboogie.com, groxis.com, dogpile.com ในปี 2001-2003 searchenginewatch.com จัดงาน "best meta search engine award" ในส่วนของการจัดกลุ่มที่มีประสิทธิภาพ รางวัลนี้ได้แก่ vivisimo.com และ

ในเดือนมกราคม 2005 google.com เลือกให้ vivisimo.com เป็นระบบที่มีการจัดเตรียมผลการสืบค้นที่ดีที่สุด ดังนั้น Google และ Microsoft ได้ให้ความสนใจในการจัดกลุ่มผลการสืบค้น เพราะในอนาคตมันจะเป็นเทคโนโลยีของการจัดลำดับความสำคัญของผลการสืบค้น ("clustering technology is the PAGERANK of the future") จากรายงานดังกล่าวแสดงให้เป็นถึงการให้ความสนใจและความสำคัญของการจัดกลุ่มผลการสืบค้นจากปี 2001-2005 เรื่อยมา แต่เทคนิคของการจัดกลุ่มระบบสืบค้นในปัจจุบันยังมีข้อบกพร่องอยู่มาก เช่น ความชัดเจนของป้ายชื่อของกลุ่ม (label) ความซ้ำซ้อนของข้อมูลในแต่ละกลุ่ม (overlap) และ ความสามารถในการแสดงผลของข้อมูลทั้งหมดหลังจากการจัดกลุ่ม (coverage) เป็นต้น

2.4 ซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC)

ซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) (Zamir, O and Etzioni O. 1997 และ Zamir, O and Etzioni O. 1998.) เป็นอัลกอริทึมการจัดกลุ่มข้อมูลแบบแฟลทคลัสเตอร์ริง (Flat Clustering) โดยจะทำงานร่วมกับโครงสร้างข้อมูลแบบซับฟิกทรี (Suffix Tree) (R. Baeza-Yates and B. Ribeiro-Neto. 1999) และจะใช้เวลาในการประมวลผลแปรผันไปกับจำนวนเอกสารทั้งหมดที่ถูกรวบรวมไว้ในเอกสารฉบับนี้หมายถึงคำอธิบายโดยสรุป (snippets) โดยกระบวนการจัดกลุ่มจะพิจารณาจากการใช้กลุ่มคำหรือวลี (phrase) ร่วมกันของเอกสาร STC จะใช้วลีเป็นตัวแทนของเอกสาร โดยที่วลีในที่นี้จะหมายถึงลำดับของคำที่ต่อเนื่องกันตั้งแต่ 1 คำขึ้นไปที่ปรากฏในเอกสาร ซึ่งป้ายชื่อของกลุ่ม (Label) จะได้มาจากวลีที่ถูกพบบ่อยในเอกสารทั้งหมดที่มี ขั้นตอนการทำงานของ STC สามารถอธิบายได้จากรหัสเทียม (Pseudo-Code) ดังตารางที่ 2. 1

ตารางที่ 2. 1 รหัสเทียม (Pseudo-Code) ของอัลกอริทึม STC

```
Search result fetching and split snippet
STEP 1 : Preprocessing
     - The non-word tokens are strip (HTML Tag,
       punctuation, number etc.)
     - Remove Stop words , if the English word are stem
       tokens to "root" word

STEP 2 : Discovering Base Clusters
     2.1  Creation of a suffix tree with n-gram technique
          of all sentence
          for each sentence {
            split sentence into n-gram (phrase)
            for each phrase{
```

ตารางที่ 2.1 (ต่อ)

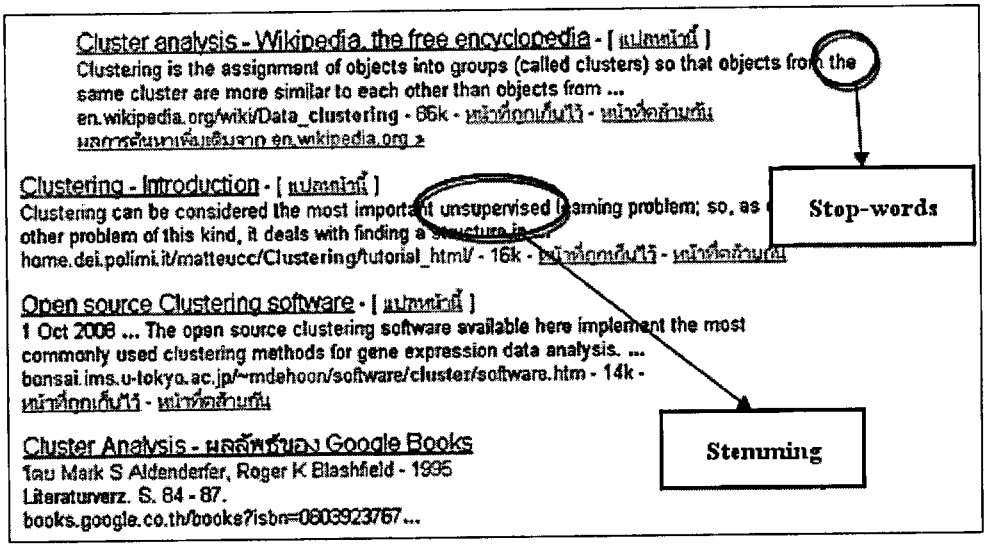
```
                insert phrase into each node of suffix tree
                update internal node with the index to current
                document while rearranging the tree  have
                common phrase, number of document
            }
        }
    2.2   Build base cluster if each node in tree have
        common phrase, number of document > 1

STEP 3 : Discovering true common phrases using a join phrase
         algorithm
  for each base cluster {
    3.1  Delete cluster B if phrase cluster B is subset of
        phrase cluster A or delete document of cluster B if
        phrase of cluster B length = 1 and Overlap with
        cluster A
  }
  for each base cluster {
    3.2  Joint base cluster A and B if phrase cluster A and
        B length > n-gram and number of document in cluster
        is subset and word's position in document in pair
        cluster is join
 }
```

จากตารางที่ 2. 1 สามารถอธิบายกระบวนการทำงานได้ดังดังนี้

2.4.1 การเตรียมข้อมูล (Preprocessing) เริ่มต้นจากเมื่อได้ผลการสืบค้นจากระบบสืบค้นบนอินเตอร์เน็ต (Snippets) จะต้องผ่านกระบวนการกรองคำที่จะใช้จัดกลุ่ม ดังนี้

1. ตัดคำที่ไม่ต้องการออกจากข้อความ เช่น Tag HTML เว้นวรรค ตัวเลข หรือเครื่องหมายต่างๆ เป็นต้น

2. กรณีที่เป็นภาษาอังกฤษ จะตัดคำจำพวกที่เป็น Stop Word เช่น a, and, the เป็นต้น และลดรูปคำให้อยู่ในรูปแบบรากศัพท์ ตัวอย่างของ Stop-Word และ Stemming ดังรูปที่ 2. 9



รูปที่ 2. 9 ตัวอย่างของ Stop-Word และ Stemming

2.4.2 การค้นหา Base Cluster (Discovering Base Clusters) นำคำที่ได้จากการผ่านกระบวนการเตรียมข้อมูล (Preprocessing) มาค้นหา Base Cluster โดยมีกระบวนการทำงานดังนี้

1. **สร้างรายการลำดับของคำ** เป็นขั้นตอนการนำข้อมูลในเอกสาร (Snippet) มาสร้างรายการลำดับของคำที่ต่อเนื่องกัน (phrase) ก่อนที่จะนำไปสร้าง Suffix Tree โดยคาดว่าลำดับของคำที่ถูกสร้างขึ้นนั้นมีความน่าจะเป็นว่าจะปรากฏเป็นวลีในภายหน้า ซึ่ง STC จะใช้วลีเป็นตัวแทนของเอกสาร เทคนิคในการใช้สร้างรายการลำดับคำของเอกสารที่ใช้ก็คือ เทคนิค N-gram

N-gram เป็นการสร้างรายการลำดับของ n คำ ซึ่งใช้ลักษณะความน่าจะเป็นว่ามันจะปรากฏเป็นวลีในภายหน้า ดังตัวอย่างการแตกประโยคไปเป็นกลุ่มคำที่อยู่ติดกันตามความยาวของ n "I don't know what to say"

ตารางที่ 2. 2 ตารางแสดงการประเภทของ N-gram

ประเภท	ข้อความที่ได้
n=1; 1-gram (unigram)	I , don't , know , what , to , say
n=2; 2-gram (bigram)	I don't , don't know , know what , what to , to say
n=3; 3-gram (trigram)	I don't know , don't know what , what to say
N-gram	.....

จากตารางที่ 2. 2 จะเห็นได้ว่าการใช้เทคนิค N-gram ในการสร้างรายการลำดับของคำ จะทำให้จำนวนของคำในวลี ซึ่งอาจจะถูกเลือกเป็นป้ายชื่อของกลุ่มนั้นมีความเป็นระเบียบ เพราะจะมีจำนวนของคำในวลีได้ไม่เกินจำนวนขนาดของ N-gram

2. **สร้าง Suffix Tree** ในขั้นตอนนี้จะเป็นการนำข้อมูลในเอกสารทั้งหมดที่ผ่านการสร้างรายการลำดับของคำด้วยเทคนิค N-gram แล้วมาทำการสร้าง Suffix Tree โดย Suffix Tree จะมีส่วนประกอบต่างๆดังต่อไปน

 - สัญลักษณ์ "สี่เหลี่ยม" แทน leaf ของ Suffix Tree โดยภายใน leaf จะประกอบไปด้วยตัวเลขคู่ลำดับ 2 หมายเลข หมายเลขแรกจะระบุถึง หมายเลขของเอกสารที่วลีนั้นๆปรากฏอยู่ และหมายเลขที่ 2 จะระบุถึงลำดับของรายการลำดับคำที่ปรากฏในเอกสารนั้นๆ

 - สัญลักษณ์ "เส้นตรง" แทน edge ของ Suffix Tree โดยแต่ละ edge จะมีป้ายชื่อ (label) กำกับอยู่ ซึ่งป้ายชื่อจะเป็นคำคำหนึ่งที่ประกอบอยู่ในรายการลำดับคำ (phrase) ของเอกสาร

 - สัญลักษณ์ "วงกลม" แทนโหนด (node) ของ Suffix Tree

Suffix Tree จะมีลักษณะพิเศษคือ edge ที่ออกไปจากโหนดเดียวกัน จะต้องมีป้ายชื่อของ edge ไม่ซ้ำกัน ในการสร้าง Suffix Tree จะเป็นการนำคำทีละคำในวลีแทน ป้ายชื่อที่กำกับ edge ของ Suffix Tree มาสร้างเรียงลำดับกันไปโดย leaf จะอยู่ส่วนท้ายสุดของ Suffix Tree เพื่อระบุถึงหมายเลขเอกสารที่วลีนั้นปรากฏอยู่และลำดับของวลีในเอกสาร

ตัวอย่างการสร้าง Suffix Tree จากเอกสาร 3 เอกสารดังต่อไปนี้

ตารางที่ 2. 3 ตัวอย่างการสร้าง Suffix Tree จากเอกสาร 3 เอกสาร

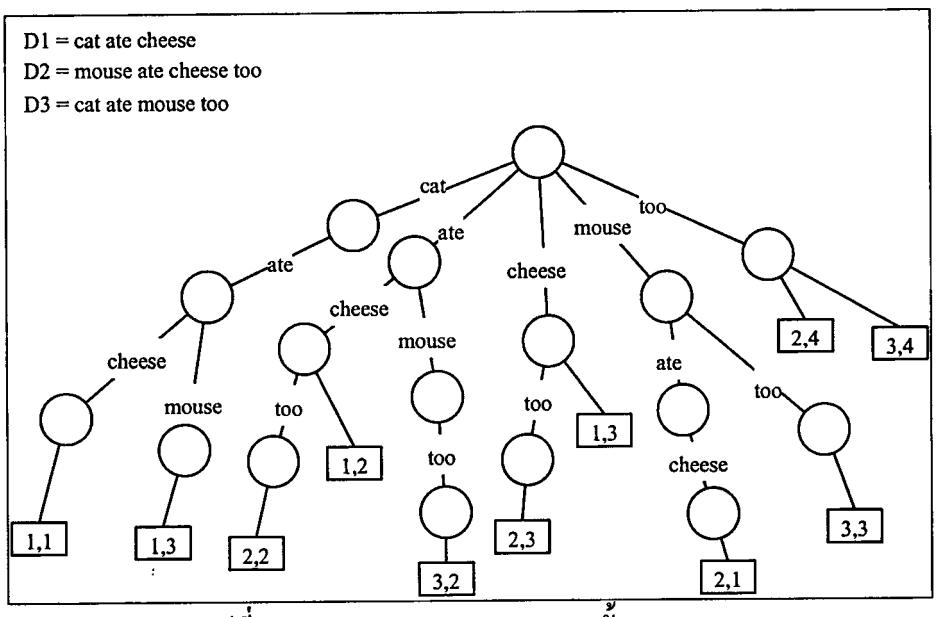
เอกสาร	ข้อมูล
D1	cat ate cheeses
D2	mouse ate chesses too
D3	cat ate mouse too

จากเอกสาร D1, D2, D3 เมื่อนำไปสร้างรายการลำดับของคำด้วยเทคนิค N-gram จะได้รายการลำดับของคำโดยใช้ N-gram ≤ 3 ดังต่อไปนี้

ตารางที่ 2. 4 การสร้าง Suffix Tree ด้วย N-gram ≤ 3

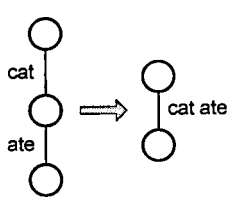
เอกสาร	ข้อความที่ได้จาก N-gram ≤ 3
D1	cat ate cheeses, ate cheeses, cheeses
D2	mouse ate cheese, ate cheeses too, cheeses too, too
D3	cat ate mouse, ate mouse too, mouse too, too

สามารถสร้าง Suffix Tree จากข้อมูลในเอกสารทั้ง 3 เอกสารได้ ดังรูปที่ 2. 10



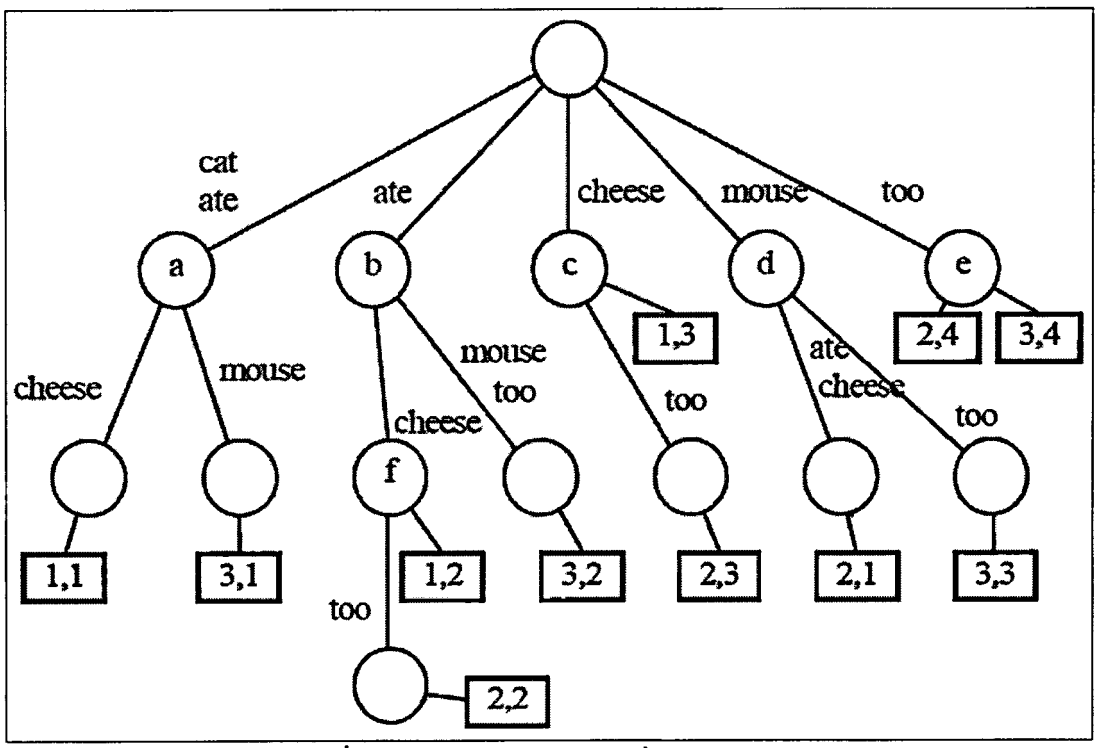
รูปที่ 2. 10 Suffix Tree ของเอกสารทั้ง 3 เอกสาร

3. ยุบรวมโหนดของ Suffix Tree เมื่อสร้าง Suffix Tree จากข้อมูลของทุกเอกสารเรียบร้อยแล้ว ทำการยุบรวมโหนดของ Suffix Tree ที่เป็นไปได้เพื่อรวมป้ายชื่อของแต่ละ edge เพื่อให้ได้วลีที่มีจำนวนคำมากที่สุด เนื่องจาก STC จะให้ความสำคัญกับจำนวนคำในวลียิ่งมีจำนวนคำมากจะยิ่งให้ความสำคัญมาก ซึ่งจะได้อธิบายต่อไปในขั้นตอนต่อไป และยังเป็นการลดจำนวนของโหนดที่อาจจะถูกเลือกให้เป็น Base Cluster ลง ทำให้เวลาที่ใช้ในการค้นหาและระบุ Base Cluster นั้นน้อยลงอีกด้วย หลักในการยุบรวมโหนดของ Suffix Tree สามารถแสดงได้ดังรูปที่ 2. 11



รูปที่ 2. 11 แสดงการยุบรวมโหนดของ Suffix Tree

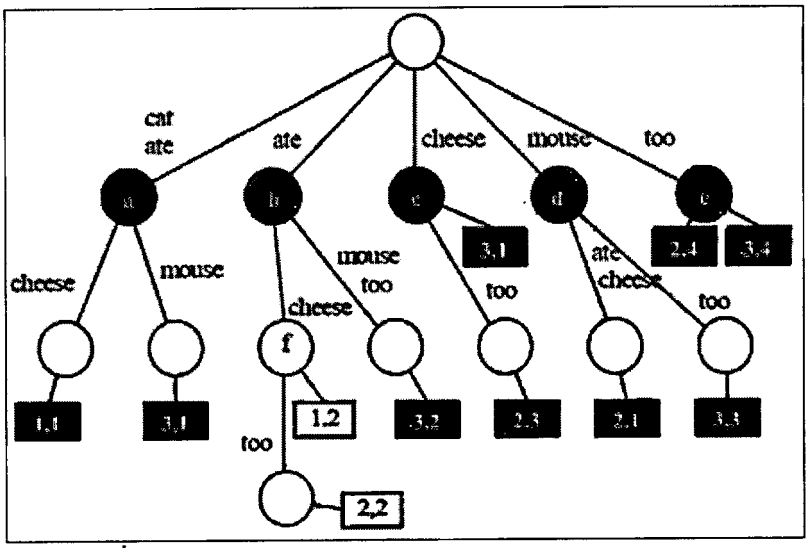
จากรูปที่ 2. 10 เมื่อทำการยุบรวมโหนดของ Suffix Tree แล้ว ป้ายชื่อของ edge จะเป็นการนำเอาป้ายชื่อของ โหนดที่ถูกยุบรวมมาเชื่อมต่อกัน ให้เป็นวลีที่มีจำนวนคำมากขึ้น เมื่อทำการยุบรวมโหนดแล้วสามารถ แสดง Suffix Tree ได้ดังรูปที่ 2. 12



รูปที่ 2. 12 แสดง Suffix Tree ที่ยุบรวมโหนดแล้ว

4. ค้นหาและระบุ Base Clusters การระบุ Base Clusters พิจารณาจากโครงสร้างข้อมูล Suffix Tree และจะมีหลักเกณฑ์ในการเลือก Base Clusters คือ จำนวนของเอกสาร หรือ จำนวน leaf ที่ปรากฏภายใน sub-tree ของโหนดใดๆ มีตั้นแต่ 2 เอกสาร หรือ 2 leaf ขึ้นไปโหนดนั้นจะถูกเลือกให้เป็น Base Cluster โดยที่ป้ายชื่อ (phrases) ของ Base Cluster คือ การนำ edge-label จากเส้นทางตั้งแต่ root จนถึงโหนดนั้นๆ มารวมกัน จะถูกกำหนดให้เป็นป้ายชื่อของกลุ่ม (phrase) และเอกสารที่ปรากฏจะถูกกำหนดให้เป็นเอกสารภายในกลุ่มของ Base Cluster นั้นๆ เช่น จากรูปที่ 2. 12 Base Cluster ของโหนด f จะมีป้ายชื่อของกลุ่มคือ ate + cheeses = ate cheese และ เอกสารในกลุ่มของ Base Cluster "f" คือ เอกสารที่ 1 และ เอกสารที่ 2

จากรูปที่ 2. 13 แสดงการระบุ base cluster จากโครงสร้างข้อมูล สามารถระบุ Base Cluster ได้ทั้งหมด 6 Base Cluster ซึ่งสามารถแสดงข้อมูลของ Base Cluster ดังตารางที่ 2. 5



รูปที่ 2. 13 แสดงการระบุ base cluster จากโครงสร้างข้อมูล

ตารางที่ 2. 5 แสดงผลการทำงานในขั้นตอนระบุ Base Cluster

Node	Phrase	Documents	S(B)
a	cat ate	1,3	(2*2) = 4
b	ate	1,2,3	(3*0) = 0
c	cheese	1,2	(2*0) = 0
d	mouse	2,3	(2*0) = 0
e	too	2,3	(2*0) = 0
f	ate cheese	1,2	(2*2) = 4

5. **คำนวณคะแนนของแต่ละ Base Cluster** ในขั้นตอนนี้จะเป็นการคำนวณคะแนนของ Base Clusters (S(B)) เพื่อนำคะแนนนี้ไปใช้ในการคัดเลือกตัวแทนของกลุ่มและคำนวณคะแนนในการจัดลำดับความสำคัญของการแสดงผลในขั้นตอน Combining Base Clusters ต่อไป

$$s(B) = |B| * f(|P|)$$
$$f(|P|) = \begin{cases} 0, & |P| = 1 \\ |P|, & if\ 2 < |P| < 6 \\ \alpha, & if\ |P| > 6 \end{cases} \qquad (2.1)$$

เมื่อ B คือ base cluster

$|B|$ คือ จำนวนของเอกสารใน B

P คือ วลีที่จะนำมาใช้เป็นป้ายชื่อของ base cluster

$|P|$ คือ จำนวนของคำที่อยู่ในวลี

α คือ ค่าคงที่

จากสูตร 2.1 S(B) จะเท่ากับจำนวนของเอกสารใน Base Cluster คูณกับค่าของ $f(|P|)$ ซึ่งค่าของ $f(|P|)$ จะมีค่าเป็น 0 เมื่อจำนวนคำในวลีซึ่งเป็นป้ายชื่อของ Base Cluster มีจำนวนคำเท่ากับ 1 คำ (single word) และจะมีค่าเท่ากับ 2 ถึง 6 เมื่อจำนวนคำในวลีมากกว่า 6 คำขึ้นไป การคำนวณคะแนนของ Base Cluster สามารถแสดงตัวอย่างการคำนวณคะแนนได้ดังนี้

จากตารางที่ 2. 5 Base Cluster "a" มีป้ายชื่อของกลุ่ม (phrase) คือ "cat ate" ที่มีจำนวนคำภายในวลีเท่ากับ 2 คำ และมีจำนวนของเอกสารที่ปรากฏใน Base Cluster 2 เอกสารคือ เอกสารที่ 1 และเอกสารที่ 3 จากสูตร

$$s(B) = |B| * f(|P|)$$

$$|B| = 2$$

$$f(|P|) = 2$$

ดังนั้น S(B) = 2*2 = 4

จากการให้คะแนนของ Base cluster จะเห็นว่า STC จะให้ความสำคัญของวลีที่มีจำนวนของคำที่ประกอบกันภายในวลีหรือ ป้ายชื่อของกลุ่มมากกว่า 1 คำ มากกว่าวลีที่มีจำนวนคำภายในวลีเพียงคำเดียว (single word) สังเกตได้จากการคิดคะแนน S(B) คือ เมื่อจำนวนคำในวลีมีเพียง 1 คำ จะให้ค่า $f(|P|)$ เป็น 0 ส่งผลให้ S(B) มีค่าเป็น 0 ทันที และผลของการคำนวณคะแนนของ Base Cluster จะแสดงให้เห็นในตารางที่ 2. 5

2.4.3 รวม Base Cluster จากผลการทำงานในขั้นตอนค้นหาและระบุ Base Clusters แสดงให้เห็นว่าเอกสารอาจมีการใช้วลีร่วมกันมากกว่า 1 วลี ส่งผลให้เอกสารสามารถปรากฏอยู่ได้ในหลายๆ Base Cluster ทำให้ Base Cluster บางกลุ่มมีความเหมือนกันสูงมาก ในขั้นตอนนี้จึงเป็นการรวม Base Cluster ที่มีความเหมือนกัน โดยจะพิจารณาจากการใช้เอกสารร่วมกัน เมื่อรวม Base Cluster ที่คล้ายกันแล้ว จะได้ Merged Cluster ซึ่งในขั้นตอนรวม Base Clusters มีหลักการทำงานดังนี้

1. **คำนวณค่า similarity ของแต่ละ Base Cluster** เพื่อที่จะค้นหาว่า Base Cluster ใดบ้างที่มีความคล้ายกัน ถ้า Base Cluster คู่ใดมีค่า similarity เป็น 1 ก็ให้ทำการรวม Base Cluster เข้าด้วยกัน แล้วเลือก Base Cluster ที่มีคะแนน S(B) สูงที่สุดเป็นตัวแสดงผลเป็นตัวแทนของ Base Cluster ทั้งหมดใน Merged cluster โดยที่ป้ายชื่อของกลุ่มก็คือป้ายชื่อ (phrase) ของ Base Cluster ที่มีคะแนน S(B) สูงสุด และเอกสารที่ปรากฏใน Merged Cluster นั้นคือเอกสารทั้งหมดที่ปรากฏใน Base Cluster ทุก Base Cluster ที่เป็นสมาชิกของ Merged Cluster นั้น

สูตรในการคำนวณค่า similarity

$$similarity(B1,B2) - \begin{cases} 1, & if \left(\frac{|B_1 \cap B_2|}{B_1}\right) > \partial \\ & and \left(\frac{|B_1 \cap B_2|}{B_2}\right) > \partial \\ 0, & otherwise \end{cases} \quad (2.2)$$

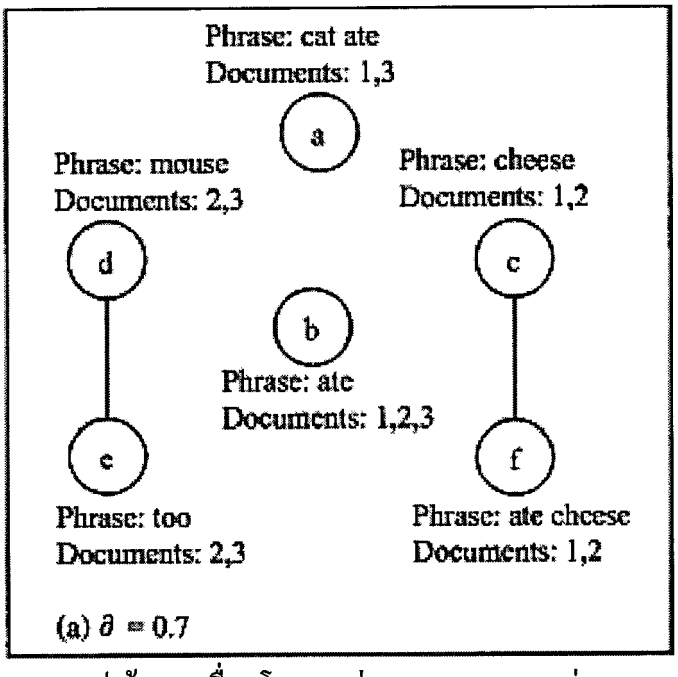
เมื่อ B คือ Base Cluster

∂ คือ ค่า Minimum Support

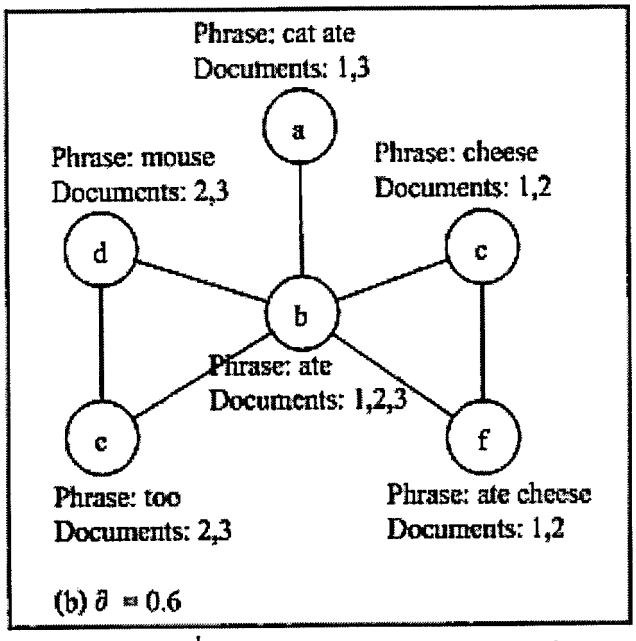
จากสูตร 2.2 ค่า similarity จะเท่ากับ 1 ก็ต่อเมื่อ จำนนสมาชิกของ B_1 กับ B_2ที่ซ้ำกัน (จำนวนของเอกสารที่ Base Cluster ทั้งสองใช้ร่วมกัน) หารด้วยจำนวนสมาชิกของ B_1 และ B_2 (จำนวนของเอกสารที่ปรากฏใน Base Cluster1 และ Base Cluster2) มีค่ามากกว่าค่าความเหมือนกันขั้นต่ำ (Minimum Support หรือ ∂) ที่กำหนดทั้งสองกรณี ในกรณีอื่นๆจะถือว่าค่า similarity มีค่าเป็น 0 ซึ่งโดยปกติค่าของ Minimum Support จะมีค่าตั้งแต่ 0.5 - 0.8 แล้วแต่ผู้พัฒนาจะกำหนด

จากการคำนวณค่า similarity สามารถนำมาวาดเป็นกราฟของความเหมือนกันของ Base Cluster ดังแสดงในรูปที่ 2. 14 และรูปที่ 2. 15

จากรูปที่ 2. 14 และรูปที่ 2. 15 Base Cluster ใดที่มีเส้นเชื่อมกันจะหมายความว่ามีค่าของ similarity =1 ซึ่งมีความคล้ายกันจะถูกรวมเข้าไว้เป็นกลุ่มเดียวกัน



รูปที่ 2. 14 กราฟเส้นการเชื่อมโยงตามค่า similarity ของแต่ละ Base Cluster



รูปที่ 2. 15 กราฟเส้นการเชื่อมโยงตามค่า similarity ของแต่ละ Base Cluster

จากรูปที่ 2. 14 ระบบกำหนดให้ค่า ∂ = 0.7 นั่นคือ จำนวนสมาชิกต้องเหมือนกัน (จำนวนของเอกสารที่ Base Cluster ทั้ง 2 ใช้ร่วมกัน) 70% ของสมาชิกภายในกลุ่มทั้งสอง จากเดิมจะได้ Base Cluster = 6 กลุ่ม คือ Base Cluster a, b, c, d, e, f เมื่อคิดค่า similarity แล้วทำการยุบรวมกัน จะเหลือ Merged Cluster = 4 กลุ่ม คือ Merged Cluster {a},{b},{c, f},{d, e} ดังตารางที่ 2. 6

จากรูปที่ 2. 15 ระบบกำหนดให้ค่า ∂ =0.6 นั่นคือ จำนวนสมาชิกต้องเหมือนกัน (จำนวนของเอกสารที่ Base Cluster ทั้ง 2 ใช้ร่วมกัน) 60% ของสมาชิกภายในกลุ่มทั้งสอง จากเดิมจะได้ Base Cluster = 6 กลุ่ม คือ Base Cluster a, b, c, d, e, f เมื่อคิดค่า similarity แล้วทำการยุบรวมกัน จะเหลือ Merged Cluster = 1 กลุ่ม คือ Merged Cluster {a, b, c, d, e, f} ดังตารางที่ 2. 6

ตารางที่ 2. 6 แสดงผลการทำงานของขั้นตอนการรวม Base Cluster

Figure	Cluster Number	Base Cluster	Documents	S(C)
(a)	1	a	1,3	4
	2	b	1, 2, 3	0
	3	d, e	2, 3	0
	4	c, f	1, 2	4
(b)	1	a, b, c, d, e, f, g	1, 2, 3	8

2. คำนวณคะแนนของ Merged Cluster S(*C*) โดยในท้ายที่สุดแล้วการจัดกลุ่มข้อมูลซึ่งเป็นผลสืบค้นที่ได้รับจากอินเตอร์เน็ต อาจจะได้กลุ่มของข้อมูลเป็นจำนวนมาก ดังนั้นจึงต้องทำการ

จัดลำดับความสำคัญในการแสดงผล โดยการคำนวณคะแนนของ Merged Cluster (S(C)) เพื่อเป็นคะแนนในการจัดลำดับความสำคัญในการแสดงผลให้กับผู้ใช้ โดยสูตรที่ใช้ในการคำนวณคะแนนของ Merged Cluster (S(C)) คือ

$$S(c) = \sum_{b \in c} S_b \qquad (2.3)$$

เมื่อ **b** คือ Base Cluster ที่ถูกรวมเข้ามาไว้ใน Merged Cluster

จากสูตร 2.3 S(C) จะเป็นผลรวมคะแนนของ Base Cluster หรือ S(B) ทั้งหมดที่อยู่ใน Merged Cluster เมื่อคำนวณค่าของ S(C) เรียบร้อยแล้ว ก็จะทำการแสดงผล Merged Cluster

2.5 ลินโก (LINGO)

ลินโก (LINGO) (Stanislaw Osilski. 2003 และ Stanislaw Osilski, Dawis Wiess. 2004) เป็นอัลกอริทึมการจัดกลุ่มข้อมูลอีกแบบหนึ่งซึ่งจัดกลุ่มข้อมูลโดยการหาป้ายชื่อของกลุ่ม (Label) ด้วยวิธีการตัดวลีที่พบบ่อยๆ ในกลุ่มเอกสารที่มีอยู่และแยกออกมาเข้ากระบวนการแยกด้วยค่าเจาะจง Single Value Decomposition (SVD) (Stanislaw Osilski. 2003) เพื่อที่จะหาป้ายชื่อของกลุ่มของข้อมูล จากนั้นจะนำป้ายชื่อของกลุ่มของข้อมูลที่ได้หามาเอกสารที่สัมพันธ์กับชื่อของกลุ่มของข้อมูล โดย LINGO จะให้ความสำคัญกับป้ายชื่อของกลุ่มที่ได้ (Description comes first clustering) (Stanislaw Osilski, Dawis Wiess. 2005) คือป้ายชื่อของกลุ่มที่ได้นั้นสามารถสื่อความหมายให้กับผู้ใช้ระบบเข้าใจได้เพียงใด ขั้นตอนการทำงานของ LINGO สามารถอธิบายได้จากรหัสเทียม (Pseudo-Code) จากตารางที่ 2. 7 ได้ดังนี้

ตารางที่ 2. 7 รหัสเทียม (Pseudo-Code) ของอัลกอริทึม LINGO

```
Search result fetching and split snippet
D ← input documents (or snippets)
Step 1 : Preprocessing
      - The non-word tokens are strip (HTML Tag,
        punctuation, number etc.)
      - Remove Stop words , if the English word are stem
        tokens to "root" word

STEP 2 : Frequent Phrase Extraction
      Concatenate all documents;
      Pc ← discover complete phrases;
```

ตารางที่ 2.7 (ต่อ)

$P_f \leftarrow p : \{p \in P_c \wedge \text{frequency}(p) > \text{Term Frequency Threshold}\}$

STEP 3 : Cluster Label Induction

A ← term-document matrix of terms not marked as stop-words and with frequency higher than the Term Frequency Threshold;

Σ, U, V SVD (A); {Product of SVD decomposition of A}

k ← 0; {Start with zero clusters}
n ← rank(A);

Repeat
k ← k + 1;
$q \leftarrow (\sum_{i=1}^{k} \Sigma_{ii}) / (\sum_{i=1}^{n} \Sigma_{ii})$;
Until q < Candidate Label Threshold;

P ← phrase matrix for P_f;
for all columns of $U_k^T P$ **do**
find the largest component m_i in the column;
add the corresponding phrase to the Cluster Label Candidates set;
labelScore ← m_i;
end for

Calculate cosine similarities between all pairs of candidate labels;

identify groups of labels that exceed the Label Similarity Threshold;

for all groups of similar labels do
select one label with the highest score;
end for

STEP 4 : Cluster Content Discovery
for all L ∈ Cluster Label Candidates **do**
create cluster C described with L;
add to C all documents whose similarity to C exceeds the Snippet Assignment Threshold;
end for
put all unassigned documents in the "Others" group;

STEP 5 : Final Cluster Formation

ตารางที่ 2.7 (ต่อ)

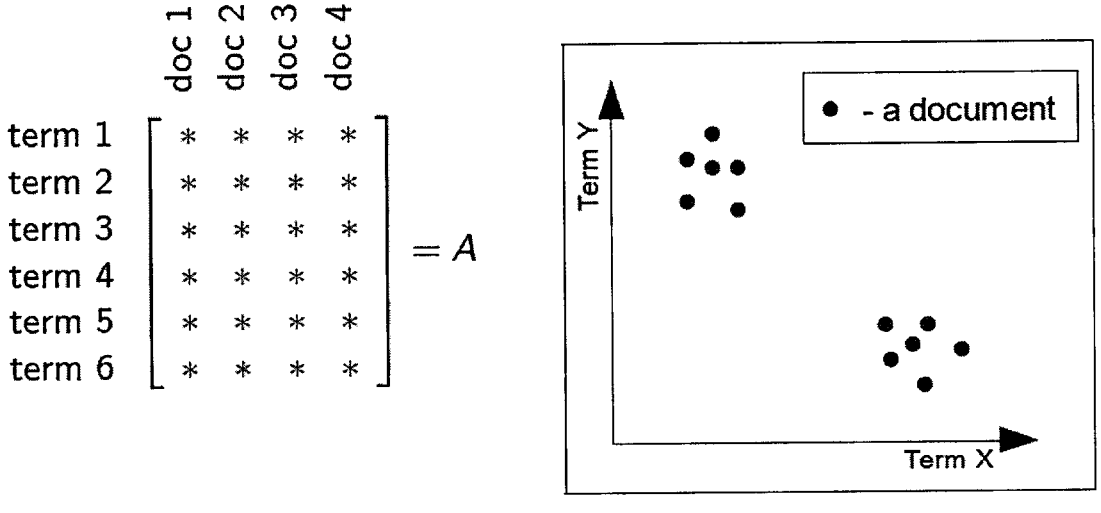
```
    for all clusters do
        clusterScore ← labelScore × ||C||;
end for
```

2.5.1 เตรียมเอกสาร (Preprocessing) เริ่มต้นจากเมื่อได้ผลการสืบค้นจากระบบสืบค้นบนอินเตอร์เน็ต (Snippets) จะต้องผ่านกระบวนการกรองคำที่จะใช้จัดกลุ่ม ดังนี้

1. ตัดคำที่ไม่ต้องการออกจากข้อความ เช่น Tag HTML เว้นวรรค ตัวเลข หรือเครื่องหมายต่างๆ เป็นต้น

2. กรณีที่เป็นภาษาอังกฤษ จะตัดคำจำพวกที่เป็น Stop Word เช่น a, and, the เป็นต้น และลดรูปคำให้อยู่ในรูปแบบรากศัพท์

2.5.2 จำแนกวลีที่ปรากฏในเอกสาร (Frequent Phrase Extraction) ในขั้นตอนนี้จะเป็นขั้นตอนการจำแนกวลีที่ปรากฏในเอกสาร โดย LINGO จะใช้โครงสร้างข้อมูลแบบซัฟฟิกอาเรย์ (Suffix Array) จากนั้นสร้าง Term-Document Matrix เพื่อเตรียมเข้าสู่กระบวนการพิสูจน์ป้ายชื่อของกลุ่ม (Cluster Label Induction) ดัง**รูปที่ 2. 16**



รูปที่ 2. 16 ตัวอย่างของ Decompose term document matrix

วลีที่ถูกจำแนกจะสามารถกลายเป็นป้ายชื่อของกลุ่ม (Label) จะต้องมีคุณสมบัติดังนี้

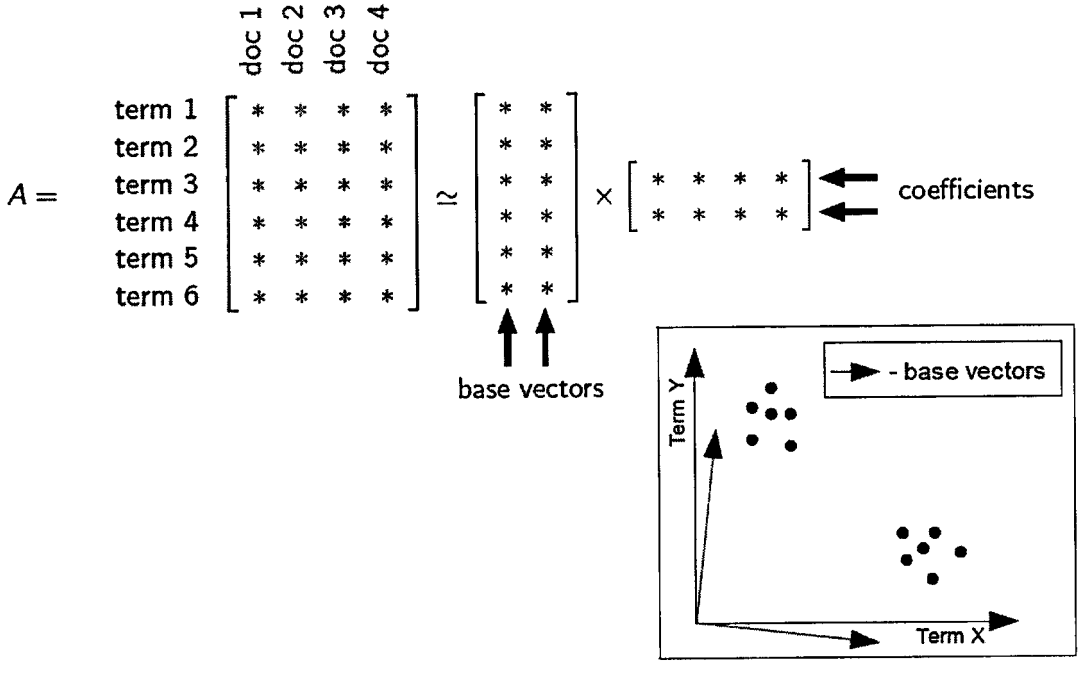
1. ต้องเป็นวลีที่ปรากฏในเอกสารทั้งหมดที่มีความถี่ต่ำ

2. เป็นคำที่สมบูรณ์แล้ว (ผ่านกระบวนการ Preprocessing มาแล้ว และวลีนั้นจะต้องมีความกระชับและมีความหมาย เช่น จะต้องเลือกวลีที่มีความหมายคล้ายๆ เช่น "Senator Hillary Rodham Clinton" และ "Hillary Rodham" เป็นต้น)

3. ต้องไม่มี Stop Word ติดมากับวลีทั้งนำหน้าและต่อท้าย ยกเว้น Stop Word ที่อยู่ตรงกลางวลีหรือประโยค

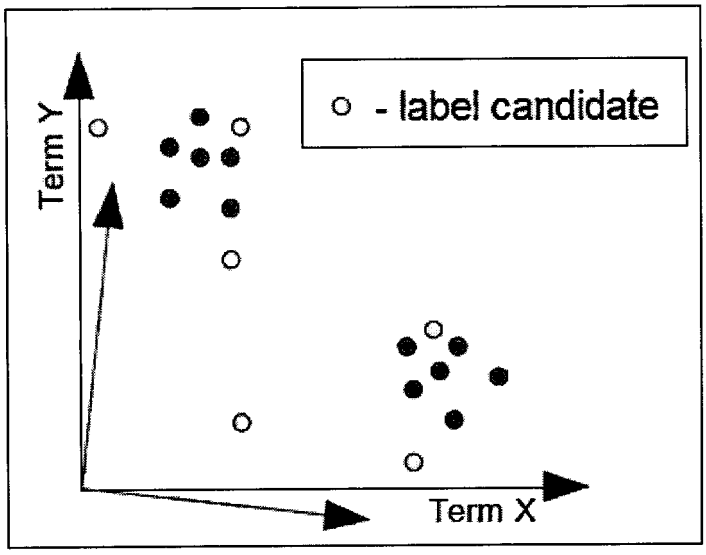
2.5.3 พิสูจน์ป้ายชื่อของกลุ่ม (Cluster Label Induction) เป็นขั้นตอนเพื่อใช้หาป้ายชื่อของกลุ่ม (Label) และพิสูจน์ว่ามีคุณสมบัติที่จะเลือกมาเป็นป้ายชื่อของกลุ่มหรือไม่ โดยมี 3 ขั้นตอนคือ

1. สร้าง Decompose term document matrix จากเอกสารที่มีอยู่ทั้งหมดออกมา และเข้ากระบวนการของ SVD ดังรูปที่ 2. 17 เมื่อผ่านกระบวนการ SVD แล้วจำได้แมทริกซ์ออกมา



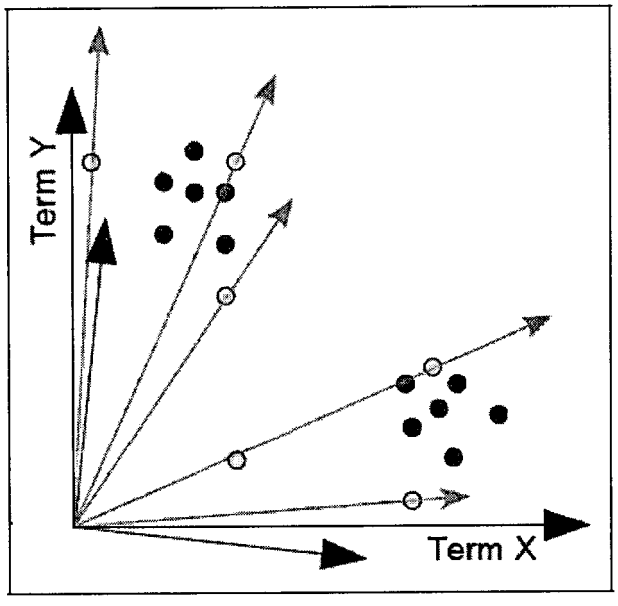
รูปที่ 2. 17 ภาพแสดงกระบวนการทำงานของ SVD

2. แยกกลุ่มของเอกสารที่มีความสัมพันธ์กันให้อยู่ใกล้กัน และคัดเลือกป้ายชื่อของกลุ่ม (Label) ออกมา จากนั้นให้คะแนนกับป้ายชื่อของกลุ่ม โดยจะให้คะแนนป้ายชื่อที่มีวลีที่ยาวที่สุดมีคะแนนสูงสุดและก็ให้ไล่เลี่ยลงมาจนถึงวลีที่มีชื่อสั้นที่สุด ดังรูปที่ 2. 18



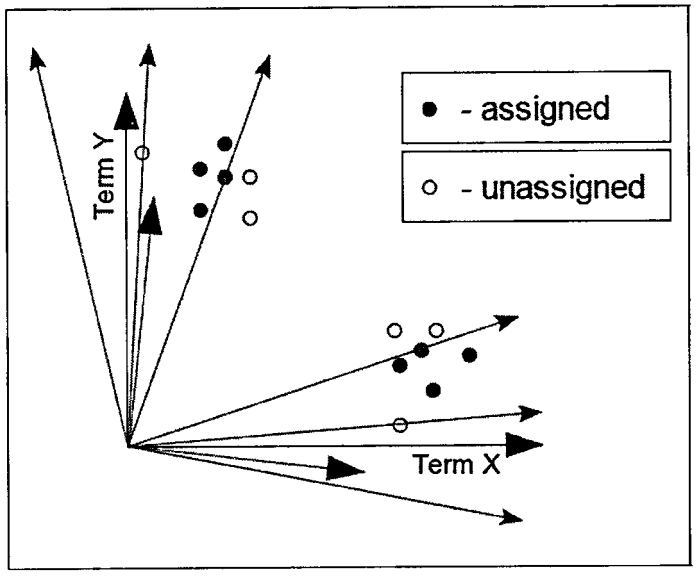
รูปที่ 2. 18 ภาพแสดงแยกกลุ่มของเอกสารที่มีความสัมพันธ์กันให้อยู่ใกล้กัน และคัดเลือกป้ายชื่อของกลุ่ม

3. จับคู่กันระหว่างวลีและป้ายชื่อของกลุ่มที่ได้ โดยการคำนวณค่าความคล้ายคลึงกันระหว่างป้ายชื่อของกลุ่มด้วยกันเองโดยใช้วิธี Cosine similarity เพื่อที่จะคัดเลือกป้ายชื่อที่มีค่าสูงที่สุดมาเป็นป้ายชื่อหลักอีกครั้งหนึ่ง โดยวัดจากคะแนนของป้ายชื่อของกลุ่มแต่ละอันว่าอันใดมีคะแนนมากกว่าก็จะได้เป็นป้ายชื่อหลักของกลุ่ม ดังรูปที่ 2. 19



รูปที่ 2. 19 ภาพแสดงการคำนวณค่าความคล้ายคลึงกันระหว่างป้ายชื่อของกลุ่มด้วยกันเองโดยใช้วิธี Cosine similarity

1.5.4 ค้นหาเอกสารที่เกี่ยวข้องกับป้ายชื่อของกลุ่ม (Cluster Content Discovery) ขั้นตอนนี้เป็นการค้นหาเอกสารที่เกี่ยวข้องกับป้ายชื่อของกลุ่มหลัก โดยเมื่อได้ป้ายชื่อของกลุ่มหลักแล้ว จากนั้นจะสร้างป้ายชื่อของกลุ่มชั่วคราว (C) เพื่อมาอธิบายป้ายชื่อของกลุ่มของกลุ่มหลัก และนำไปเปรียบเทียบกับเอกสาร (Snippets) ที่มีอยู่ว่ามีความคล้ายคลึงกันเพียงใด โดยจะมีการกำหนดค่าความคล้ายคลึงนั้นไว้ ซึ่งหากเมื่อเปรียบเทียบกันแล้วมีค่ามากกว่าค่าที่กำหนดไว้ หมายความว่าเอกสารนั้นอยู่ภายใต้ป้ายชื่อนั้น (Assigned) หากมีค่าน้อยกว่าจะเป็นเอกสารที่อยู่นอกเหนือจากกลุ่มที่เปรียบเทียบ (Unassigned) ดังรูปที่ 2. 20



รูปที่ 2. 20 ภาพแสดงการค้นหาเอกสารที่เกี่ยวข้องกับป้ายชื่อของกลุ่ม

1.5.5 จัดลำดับการแสดงผลกลุ่มข้อมูล (Final Cluster Formation) ขั้นตอนสุดท้ายคือการจัดลำดับในการแสดงผลกลุ่มของข้อมูล โดยวัดจากคะแนนของกลุ่ม ว่ากลุ่มใดมีคะแนนมากที่สุดจะได้แสดงรายการเป็นอันดันแรกและไล่เลี่ยกันลงมาตามคะแนนที่ได้ โดยคะแนนของกลุ่มนั้นได้จาก

```
clusterScore ← labelScore × ||C||;
```

clusterScore	คือ คะแนนกลุ่ม
labelScore	คือ คะแนนป้ายชื่อของกลุ่ม
‖C‖	คือ จำนวนเอกสารที่อยู่ในกลุ่มของ C

บทที่ 3
การออกแบบและพัฒนาระบบ

ข้อด้อยของระบบสืบค้นข้อมูลบนอินเตอร์เน็ตในปัจจุบันคือ การที่ได้มาซึ่งผลการสืบค้นเป็นจำนวนมาก ทำให้เกิดปัญหาการเข้าถึงเนื้อหาที่ตรงตามความต้องการของผู้ใช้งาน เนื่องจากผลการสืบค้นที่ได้มานั้นมีจำนวนมาก อาจทำให้ข้อมูลที่ผู้ใช้งานต้องการใช้นั้นอาจต้องใช้เวลาในการหานานขึ้น หรือข้อมูลที่ได้มาอาจจะไม่ใช้ข้อมูลที่ตรงกับความต้องการทั้งหมด

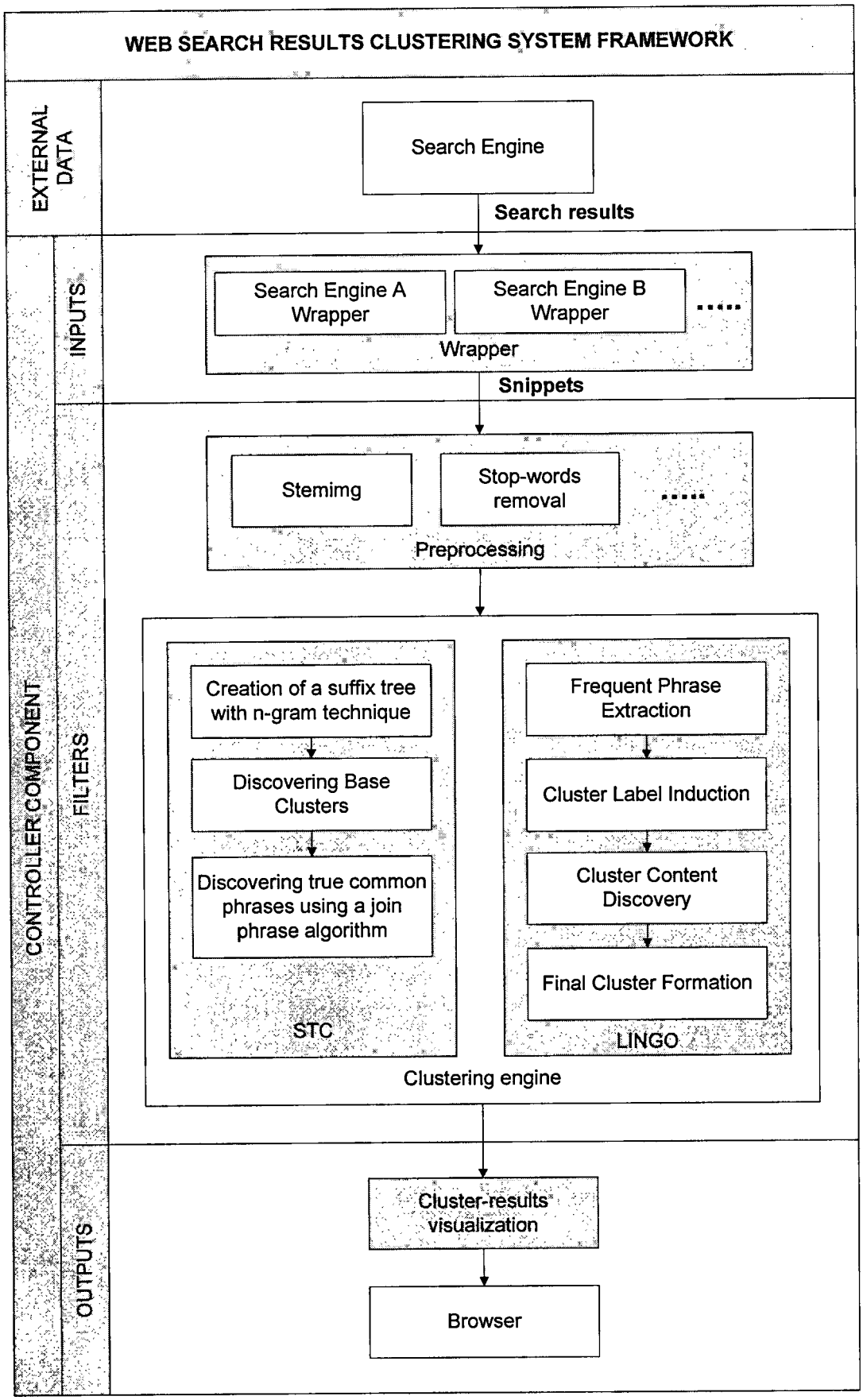
จากปัญหาข้างต้นนี้ทำให้เกิดแนวคิดที่ต้องการจัดกลุ่มข้อมูลผลการสืบค้นให้อยู่ในรูปแบบกลุ่มผลการสืบค้นตามเนื้อหาที่ผลลัพธ์นั้นๆ เกี่ยวข้อง เพื่อช่วยให้ผู้ใช้ระบบสามารถเข้าไปดูผลสืบค้นข้อมูลได้ตามหมวดหมู่ที่ตนสนใจ ซึ่งจะช่วยให้การแสดงผลการสืบค้นมีประสิทธิภาพมากยิ่งขึ้น ทำให้ช่วยประหยัดเวลาของผู้สืบค้นข้อมูล ซึ่งในเอกสารเล่มนี้จะใช้อัลกอริทึม STC และ LINGO มาประยุกต์ใช้ในระบบ โดยผู้ใช้สามารถเลือกได้ว่าจะจัดกลุ่มโดยใช้อัลกอริทึมใด

การออกแบบระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ตนี้ ใช้หลักการของ Component Based Programming โดยการแยกกระบวนการทำงานออกมาแต่ละส่วนเป็นอิสระต่อกันรวมถึงรองรับกับการขยายระบบ เช่น เพิ่มอัลกอริทึมที่ใช้จัดกลุ่มข้อมูลได้ ติดต่อหรือให้บริการกับโปรแกรมหรือระบบงานภายนอกได้ในอนาคต เป็นต้น โดยการทำงานในแต่ละส่วนจะติดต่อกันข้อมูลรูปแบบ XML

ในบทที่ 3 นี้จะกล่าวถึงการออกแบบระบบการจัดกลุ่มผลการสืบค้นบนอินเตอร์เน็ตและรายละเอียดในการพัฒนาระบบ ได้แก่ การออกแบบเฟรมเวิร์ค (Framework) การทำงานภายในระบบ รูปแบบการเชื่อมต่อข้อมูล และการออกแบบการแสดงผลลัพธ์ของระบบ รวมถึงเครื่องมือและภาษาที่ใช้พัฒนาระบบ

3.1 เฟรมเวิร์ค (Framework) การทำงานของระบบ

ระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ตนี้ ใช้หลักการของ Component Based Programming โดยการแยกกระบวนการทำงานออกมาแต่ละส่วนเป็นอิสระต่อกันรวมถึงรองรับกับการขยายระบบ เช่น เพิ่มอัลกอริทึมที่ใช้จัดกลุ่มข้อมูลได้ ติดต่อหรือให้บริการกับโปรแกรมหรือระบบงานภายนอกได้ในอนาคต เป็นต้น ระบบจะมีส่วนการที่ทำงานที่ชื่อ “Controller Component” เพื่อใช้ควบคุมการทำงานของทั้งระบบ โดยมีการออกแบบเฟรมเวิร์ค (Framework) การทำงานของระบบ ดังรูปที่ 3. 1



รูปที่ 3. 1 รูปแสดงเฟรมเวิร์ค (Framework) การทำงานของระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต

จากรูปที่ 3. 1 แสดงเฟรมเวิร์ค (Framework) การทำงานของระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต รายละเอียดขั้นตอนการทำงานของระบบมีดังนี้

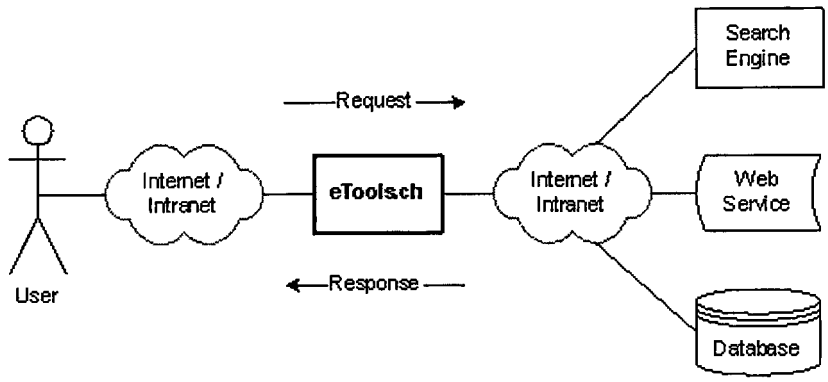
3.1.1 ดึงข้อมูลจากระบบภายนอก (EXTERNAL DATA) เนื่องจากระบบจัดกลุ่มผลการสืบค้นข้อมูลฯ จำเป็นต้องใช้ผลการสืบค้นข้อมูลบนอินเตอร์เน็ต (Search Result) เพื่อใช้ในการจัดกลุ่ม ดังนั้นการออกแบบระบบจึงได้เลือกใช้การเชื่อมต่อข้อมูลจากระบบภายนอกเพื่อต้องการข้อมูลที่หลายหลาย จึงใช้การเชื่อมต่อจากระบบสืบค้นข้อมูลบนอินเตอร์เน็ต (Search Engine) หลายราย เช่น Google, Yahoo, MSN ซึ่งระบบสืบค้นข้อมูลบนอินเตอร์เน็ตเหล่านี้จะมีบริการ (Service) ให้ผู้พัฒนาระบบ (Developer) สามารถเข้ามาใช้บริการในรูปแบบ Application Programming Interface (API) ได้ดังรูปที่ 3. 2

Google (API) Yahoo (API) MSN (API)

รูปที่ 3. 2 ตัวอย่างระบบสืบค้นข้อมูลบนอินเตอร์เน็ตที่ให้บริการ (Service) ในรูปแบบ API

จากรูปที่ 3. 1 หากต้องการเชื่อมต่อกับระบบสืบค้นข้อมูลบนอินเตอร์เน็ตรายใด ก็จำเป็นสร้างส่วนการทำงานย่อยของระบบ (Wrapper) เพื่อที่ใช้เชื่อมต่อกับระบบสืบค้นข้อมูลบนอินเตอร์เน็ตแต่ละราย เนื่องจากรูปแบบของข้อมูลที่ระบบสืบค้นข้อมูลบนอินเตอร์เน็ตแต่ละรายให้บริการนั้นไม่เหมือนกัน ดังนั้นจึงจำเป็นจะต้องสร้าง Wrapper เพื่อช่วยแปลงข้อมูลให้อยู่ในรูปแบบเดียวกันทั้งหมด เพื่อนำไปเข้าสู่กระบวนการถัดไป

โดยข้อมูลที่ใช้ในที่นี้ระบบจัดกลุ่มผลการสืบค้นข้อมูลฯ จะดึงข้อมูลจากภายนอกใช้โดยใช้วิธี Application Programming Interface (API) ซึ่งในระบบจะเลือกใช้บริการจากเว็บ www.etool.ch ซึ่งการให้บริการของเว็บนี้มีลักษณะการให้บริการ (Service) ส่งผลการสืบค้นจากแหล่งข้อมูลต่างๆ เช่น Search Engine, Web Service หรือจากฐานข้อมูลจากที่ต่างๆ ส่งกลับมายังผู้เรียกใช้บริการ การทำงานของเว็บ www.etool.ch มีลักษณะดังรูปที่ 3. 3



รูปที่ 3. 3 ภาพแสดงการทำงานของเว็บ www.etool.ch

จากรูปที่ 3. 3 แสดงการทำงานของของเว็บ www.etool.ch โดย User หมายถึงระบบจัดกลุ่มผลการสืบค้นข้อมูลฯ ซึ่งผลการลืบค้นที่ได้มาจาก www.etool.ch จะได้จากการดึงข้อมูลจากแหล่งต่างๆ ได้แก่ Google, Ask, Yahoo, Altavista, Seekport, Live, Search, Cuil, Wikia และ Lycos ซึ่งผลการสืบค้นที่นำมาจัดกลุ่มนั้นจะถูกแปลงให้อยู่ในรูปแบบของเอกสาร XML โดยมีรูปแบบดังตารางที่ 3. 1 และตารางที่ 3. 2

ตารางที่ 3. 1 Tag ในเอกสาร XML ที่ระบบสร้างขึ้น

Tag	ความหมาย
<searchresult>	Element ที่เก็บผลลัพธ์การสืบค้นทั้งหมดในแต่ละครั้งที่สืบค้นได้
<document>	Element ที่เก็บข้อมูลผลลัพธ์แต่ละรายการ โดยมี attribute "id" เพื่อไว้ใช้ในการอ้างอิง
<title>	Element ที่เก็บ Title ของแต่ละผลลัพธ์
<url>	Element ที่เก็บ URL ของแต่ละผลลัพธ์
<snippet>	Element ที่เก็บคำอธิบายโดยสรุปของแต่ละผลลัพธ์

ตารางที่ 3. 2 รูปแบบเอกสาร XML ที่ระบบสร้างขึ้น

```
<searchresult>
  <query></query>
  <document id="">
     <title></title>
     <url></url>
     <snippet></snippet>
  </document>

  ...

</searchresult>
```

ตารางที่ 3. 3 ตัวอย่างเอกสาร XML ที่ได้รับสร้าง

```
<searchresult>
  <query>Globe</query>
  <document id="0">
     <title>default</title>
     <url>http://www.globe.com.ph/</url>
     <snippet>
          Provides mobile communications (GSM) including
          GenTXT, handyphones, wireline services, an
          broadband Internet services.
     </snippet>
  </document>

  <document id="1">
     <title>Skate Shoes by Globe | Time For Change</title>
     <url>http://www.globeshoes.com/</url>
    <snippet>
          Skaters, surfers, and showboarders
          designing in their own style.
    </snippet>
  </document>

  ...

</searchresult>
```

3.1.2 กระบวนการกลั่นกลองข้อมูล (FILTERS) คือ การนำข้อมูลเข้ากระบวนการต่างๆ ในการจัดกลุ่มข้อมูล โดยมีการบวนารทั้งหมดดังนี้

1. เตรียมข้อมูล (Preprocessing) การเตรียมเอกสารก่อนเข้ากระบวนการจัดกลุ่มในแต่ละอัลกอริทึม จะต้องผ่านการเตรียมข้อมูลก่อนเพื่อให้ข้อมูลที่จะนำไปจัดกลุ่มนั้นมีคำที่เหมาะกับการที่จะหาป้ายชื่อของกลุ่ม (Label) และเพื่อเพิ่มประสิทธิภาพในการทำงานให้มากขึ้นด้วย โดยมีขั้นตอนการทำงานดังนี้

Stemming คือ การทำ Stemming คือการแปลงหรือตัดส่วนของคำให้กลายเป็นรากศัพท์ของคำเหล่านั้น เช่น คำว่า “process” “processing” หรือ “processed” โดยคำทั้งหมดนี้มาจากรากศัพท์เดียวกันคือ “process” ข้อดีของการทำ Stemming คือคำสำคัญในการค้นคืน “processing” ถูกตัดส่วนไปเป็น “process” ก่อน คำสำคัญดัชนีนี้จะถูกใช้ในการค้นหาและค้นคืนเอกสารซึ่งใช้เป็นคำสำคัญของ “processing” และ “processed” ด้วย ส่วนข้อเสียของการ Stemming คือบางครั้งคำที่ค้นคืนออกมาจะเป็นคำที่ถูกตัดส่วนแล้วเป็นรากศัพท์คำเดียวกันแต่มีความเกี่ยวกับคำค้นคืนไม่

เหมือนกัน เช่น คำค้นคืน "process management" อาจคืนเอกสาร "food processors" ออกมาด้วย โดยระบบจะใช้อัลกอริทึม Porter Stemming มาใช้ในขั้นตอนนี้

Stop-words removal คือ การลดคำที่มีความถี่มากเกินไปในการปรากฏของเอกสารเช่น a หรือ the คำเหล่านี้จะถูกกำจัดทิ้งไปเพราะคำเหล่านี้จะปรากฏในทุกๆเอกสาร การนำ Stop-words มาใช้จะช่วยให้ความต้องการในการจัดเก็บดัชนีของคำน้อยลงและคำที่มีความถี่สูงๆ เมื่อถูกกำจัดทิ้งไปจะทำให้การค้นคืนเอกสารมีความรวดเร็วขึ้น แต่ข้อเสียของการใช้ Stop-words คือในการค้นหาบางครั้ง ผู้ใช้ก็ต้องการคำที่มาจาก Stop-words ด้วย โดยระบบจะใช้วิธีการเก็บรวบรวม Stop-words ทั้งหมดไว้ในไฟล์เพื่อใช้เปรียบเทียบกับเอกสารที่จะนำไปจัดกลุ่ม หากพบว่ามีคำในเอกสารเหมือนกับ Stop-words ที่เก็บไว้ในไฟล์ก็จะถูกทำเครื่องหมายกำกับไว้ว่าคำนั้นเป็น Stop-words โดยสามารถอธิบายกระบวนการทำงานได้จากรหัสเทียม (Pseudo-Code) ในตารางที่ 3. 4

ตารางที่ 3. 4 รหัสเทียม (Pseudo-Code) ของ Stop-words removal

```
for each document
{
      for each available stoplist
      {
            count the occurrences of terms from the stoplist
            in the document;
      }

      choose the stoplist that recorded the highest number
      of occurrences;

      if (the highest number of occurrences > 1)
      {

            decide that the document's language is the
            stoplist's language;
      }
      else
      {
            decide that the document's language is
            unidentified;
      }
}
```

2. เครื่องมือในการจัดกลุ่ม (Clustering Engine) ในขั้นตอนนี้จะเข้าสู่เครื่องมือในการจัดกลุ่ม ซึ่งก็คืออัลกอริทึมที่ระบบจัดกลุ่มผลการสืบค้นฯ ได้นำมาประยุกต์ใช้ โดยมี 2 อัลกอริทึม คือ

ซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO) โดยมีขั้นตอนการทำงานในแต่ละอัลกอริทึมดังนี้

STC – การจัดกลุ่มด้วยอัลกอริทึม STC ออกแบบกระบวนการทำงานออกเป็น 3 ส่วนหลัก ได้แก่

1. สร้าง Suffix Tree สามารถอธิบายกระบวนการทำงานได้จากรหัสเทียม (Pseudo-Code) ในตารางที่ 3. 5

ตารางที่ 3. 5 รหัสเทียม (Pseudo-Code) ของการสร้าง Suffix Tree

```
split text into sentences consisting of words;
for each document
{
      for each sentence
      {
            if (sentence length > 0)
            {
                  insert sentence and all its substrings into
                  generalized suffix tree and update internal
                  nodes  with the index to current document
                  while rearranging the tree;
            }
      }
}
```

2. ค้นหา Base Cluster สามารถอธิบายกระบวนการทำงานได้จากรหัสเทียม (Pseudo-Code) ในตารางที่ 3. 6

ตารางที่ 3. 6 รหัสเทียม (Pseudo-Code) ของการค้นหา Base Cluster

```
for each node in the tree
{
      if (number of documents in node's subtree > 2)
      {
            if (candidateBaseClusterScore >
                Minimal_Base_Cluster_Score)
            {
                  add a base cluster to the list of base
                  clusters;
            }
      }
}
```

3. รวมป้ายชื่อของกลุ่มที่มีความคล้ายคลึงกัน(Merge Cluster) และเลือกป้ายชื่อหลักเพื่อใช้แทนกลุ่มของป้ายชื่อ สามารถอธิบายกระบวนการทำงานได้จากรหัสเทียม (Pseudo-Code) ในตารางที่ 3. 7

ตารางที่ 3. 7 รหัสเทียม (Pseudo-Code) ของการรวมป้ายชื่อของกลุ่ม

```
    build a graph where nodes are base clusters and there
is a link between node A and B if and only if the number of
common documents indexed by A and B is greater than the
Merge_Threshold;
    clusters are coherent subgraphs of that graph;
```

LINGO – การจัดกลุ่มด้วยอัลกอริทึม LINGO มีการออกแบบการทำงานมาได้ 4 ส่วน ได้แก่

1. จำแนกวลีที่ปรากฏในเอกสาร (Frequent Phrase Extraction) สามารถอธิบายกระบวนการทำงานได้จากรหัสเทียม (Pseudo-Code) ในตารางที่ 3. 8

ตารางที่ 3. 8 รหัสเทียม (Pseudo-Code) ของการจำแนกวลีที่ปรากฏในเอกสาร (Frequent Phrase Extraction)

```
STEP : 1 Conversion of the representation
     for each document
     {
          convert the document from the character-based to
          the word-based representation;
     }

STEP : 2 Document concatenation
     concatenate all documents;
     create an inverted version of the concatenated
     documents;

STEP : 3 Complete phrase discovery
     discover right-complete phrases;
     discover left-complete phrases;
     sort the left-complete phrases alphabetically;
     combine the left- and right-complete phrases into a
     set of complete phrases;

STEP : 4 Final selection
          for further processing choose the terms and phrases
     whose frequency exceed the Term Frequency Threshold;
```

2. พิสูจน์ป้ายชื่อของกลุ่ม (Cluster Label Induction) สามารถอธิบายกระบวนการทำงานได้จากรหัสเทียม (Pseudo-Code) ในตารางที่ 3. 9

ตารางที่ 3. 9 รหัสเทียม (Pseudo-Code) ของ พิสูจน์ป้ายชื่อของกลุ่ม (Cluster Label Induction)

```
STEP 1 : Term-document matrix building
        Build the term-document matrix A for the input
    snippet collection. as index terms use the non-stop
    words that exceed the predefined term frequency
    threshold. use the tf-idf weighting scheme;

STEP 2 : Abstract concept discovery
        perform the Singular Value Decomposition of the
    term-document matrix to obtain U, S and V matrices;
    based on the value of the q parameter and using the S
    matrix - calculate the desired number k of abstract
    concepts;
    use the first k columns of the U matrix to form the Uk
    matrix;

STEP 3 : Phrase matching
          using the tf-idf term weighting create the phrase
   matrix P;

   for each column of the Uk matrix
   {
       multiplicate the column by the P matrix;
       find the largest value in the resulting vector to
       determine the best matching phrase;
   }

STEP 4 : Candidate label pruning
          calculate similarities between all pairs of
    candidate labels;
          form groups of labels that exceed a predefined
    similarity threshold;

    for each group of similar labels
    {
       select one label with the highest score;
    }
```

3. ค้นหาเอกสารที่เกี่ยวข้องกับป้ายชื่อของกลุ่ม ขั้นตอนนี้เป็นการค้นหาเอกสารที่เกี่ยวข้องกับป้ายชื่อของกลุ่มหลัก โดยเมื่อได้ป้ายชื่อของกลุ่มหลักแล้ว จากนั้นจะสร้างป้ายชื่อของ

กลุ่มชั่วคราว เพื่อมาอธิบายป้ายชื่อของกลุ่มของกลุ่มหลัก และนำไปเปรียบเทียบกับเอกสาร (Snippets) ที่มีอยู่ว่ามีความคล้ายคลึงกันเพียงใด โดยจะมีการกำหนดค่าความคล้ายคลึงนั้นไว้ ซึ่งหากเมื่อเปรียบเทียบกันแล้วมีค่ามากกว่าค่าที่กำหนดไว้ หมายความว่าเอกสารนั้นอยู่ภายใต้ป้ายชื่อนั้น และหากมีค่าน้อยกว่าจะเป็นเอกสารที่อยู่นอกเหนือจากกลุ่มที่เปรียบเทียบ

4. ค้นหาเอกสารที่เกี่ยวข้องกับป้ายชื่อของกลุ่ม (Cluster Content Discovery) นำป้ายชื่อของกลุ่มชั่วคราวไปเปรียบเทียบกับเอกสาร (Snippets) ที่มีอยู่ว่ามีความคล้ายคลึงกันเพียงใด โดยจะมีการกำหนดค่าความคล้ายคลึงนั้นไว้ ซึ่งหากเมื่อเปรียบเทียบกันแล้วมีค่ามากกว่าค่าที่กำหนดไว้ หมายความว่าเอกสารนั้นอยู่ภายใต้ป้ายชื่อนั้น และหากมีค่าน้อยกว่าจะเป็นเอกสารที่อยู่นอกเหนือจากกลุ่มที่เปรียบเทียบ

ขั้นตอนสุดท้ายคือการจัดลำดับในการแสดงผลกลุ่มของข้อมูล โดยวัดจากคะแนนของกลุ่มว่ากลุ่มใดมีคะแนนมากที่สุดจะได้แสดงรายการเป็นอันดันแรกและไล่เลี่ยกันลงมาตามคะแนนที่ได้

เมื่อผ่านกระบวนการจัดกลุ่มข้อมูลเรียบร้อยแล้ว ระบบจะสร้างเอกสาร XML ขึ้นมาใหม่เพื่อไว้เก็บข้อมูลผลการสืบค้นและข้อมูลเกี่ยวกับผลลัพธ์ที่ได้จัดกลุ่มไว้ เพื่อที่จะนำเข้าสู่กระบวนการแสดงผลต่อไป โดยรูปแบบเอกสาร XML ที่ระบบสร้างขึ้นมีรูปแบบดังตารางที่ 3. 10 และ ตารางที่ 3. 11

ตารางที่ 3. 10 Tag ของเอกสาร XML ที่ระบบสร้างขึ้น

Tag	ความหมาย
<searchresult>	Element ที่เก็บผลลัพธ์การสืบค้นทั้งหมดในแต่ล่ะครั้งที่สืบค้นได้
<document>	Element ที่เก็บข้อมูลผลลัพธ์แต่ละรายการ โดยมี attribute “id” เพื่อไว้ใช้ในการอ้างอิง
<title>	Element ที่เก็บ Title ของแต่ละผลลัพธ์
<url>	Element ที่เก็บ URL ของแต่ละผลลัพธ์
<snippet>	Element ที่เก็บคำอธิบายโดยสรุปของแต่ละผลลัพธ์
<group>	Element ที่เก็บข้อมูลหมวดหมู่ โดยมี attribute “id” เพื่อรหัสกลุ่มเพื่อใช้ในการอ้างอิง และ มี attribute “size” เพื่อเก็บข้อมูลจำนวนเอกสารที่อยู่ในกลุ่มนี้
<title>	Element ที่เก็บข้อมูลชื่อหมวดหมู่
<phrase>	Element ที่เก็บข้อมูลชื่อหมวดหมู่ที่จัดกลุ่มได้

ตารางที่ 3.10 (ต่อ)

Tag	ความหมาย
<document>	Element ที่เก็บข้อมูลรหัสเอกสารที่อยู่ภายใต้หมวดหมู่นั้นๆ โดยมี attribute "refid" เพื่อไว้ใช้ในการอ้างอิงไปยังผลการสืบค้นที่อยู่ภายใต้กลุ่ม

ตารางที่ 3. 11 รูปแบบของเอกสาร XML ที่ระบบสร้างขึ้น

```
<searchresult>
  <query>...</query>
  <document id="...">
     <title>...</title>
     <url>...</url>
     <snippet>...</snippet>
  </document>

  ...

<group id="..." size="...">
   <title>
      <phrase>...</phrase>
    </title>
</group>

<group id="..." size="...">
   <title>
      <phrase>...</phrase>
   </title>
   <document refid="..."/>
   <document refid="..."/>

   ...

</group>

...

</searchresult>
```

ตารางที่ 3. 12 ตัวอย่างเอกสาร XML ที่สร้าง

```
<searchresult>
  <query>Globe</query>
  <document id="0">
    <title>default</title>
    <url>http://www.globe.com.ph/</url>
    <snippet>
      Provides mobile communications (GSM) including
      GenTXT, handyphones, wireline services, an
      broadband Internet services.
    </snippet>
  </document>

  <document id="1">
    <title>Skate Shoes by Globe | Time For Change</title>
    <url>http://www.globeshoes.com/</url>
    <snippet>
      Skaters, surfers, and showboarders
      designing in their own style.
    </snippet>
  </document>

    ...

<searchresult>

<group id="0" size="60">
   <title>
      <phrase>com</phrase>
   </title>
</group>

<group id="1" size="2">
   <title>
      <phrase>amazon.com</phrase>
   </title>
   <document refid="43"/>
   <document refid="77"/>
</group>

<group id="2" size="2">
   <title>
      <phrase>boston.com</phrase>
   </title>

   <document refid="4"/>
   <document refid="7"/>
```

ตารางที่ 3.12 (ต่อ)

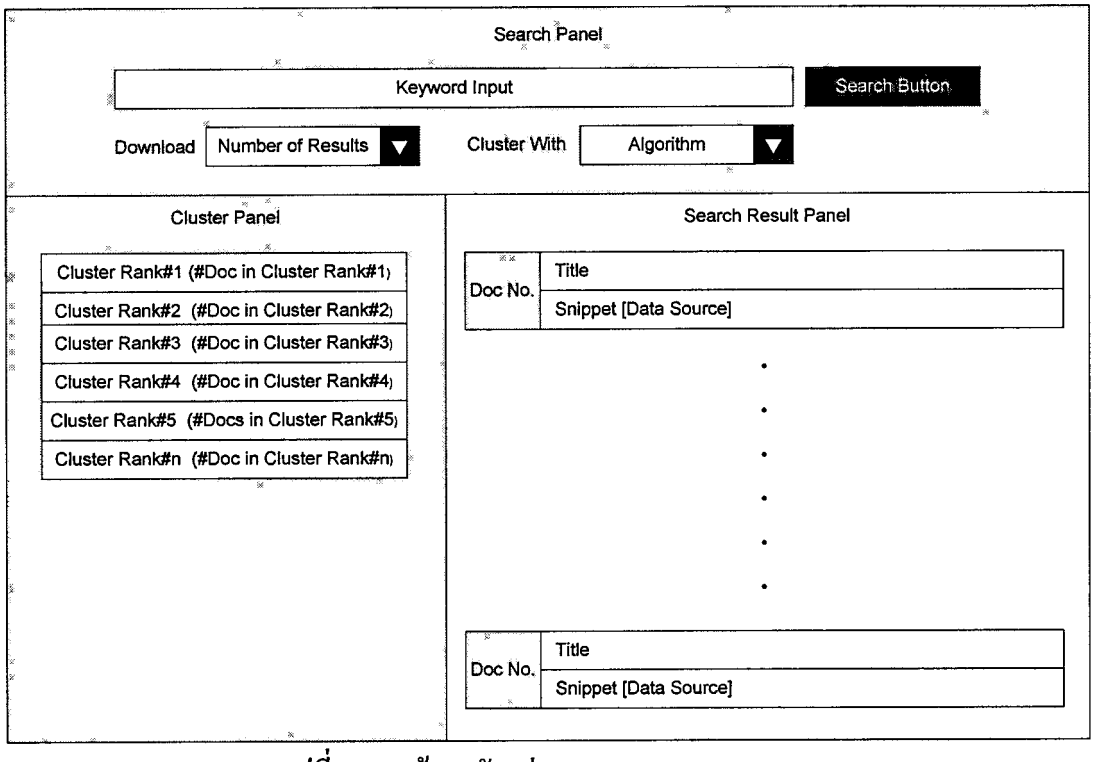
```
</group>

...

<group id="7" size="48">
   <title>
      <phrase>Other Sites</phrase>
   </title>
   <document refid="1"/>
   <document refid="2"/>

   ...

</group>

<group id="8" size="12">
   <title>
     <phrase>org</phrase>
   </title>
</group>

<group id="9" size="2">
   <title>
        <phrase>en.wikipedia.org</phrase>
      </title>
      <document refid="9"/>
      <document refid="14"/>

      ...

    </group>

...

</searchresult>
```

3. การแสดงผล (OUTPUT) นำเอกสาร XML ที่สร้างมาประมวลผลเพื่อแสดงผลลัพธ์โดยจากอ่านค่าต่างๆ และแปลงออกมาอยู่ในรูปแบบที่ผู้ใช้เข้าใจได้ง่าย เช่นตัวอย่างข้อมูลในเอกสารในตารางที่ 3. 13

ตารางที่ 3. 13 แสดงตัวอย่างข้อมูลในเอกสารบางส่วน

```
<group id="1" size="2">
  <title>
    <phrase>amazon.com</phrase>
  </title>
  <document refid="43"/>
  <document refid="77"/>
</group>
```

จากตารางที่ 3. 13 พบว่า Tag <group> มี attribute "id" โดยมีค่าเท่ากับ 1 หมายความว่ากลุ่มข้อมูลนี้ มีการจัดอันดับได้เป็นกลุ่มที่ 1 และ attribute "size" มีค่าเท่ากับ 2 หมายความว่ากลุ่มเอกสารนี้มีจำนวนเอกสารที่เกี่ยวข้องอยู่ 2 รายการ และมีชื่อกลุ่มว่า "Other Sites" ที่แสดงใน Tag <phrase> และใน Tag <document> ที่ attribute "refid" มีค่า 43 และ 77 ตามลำดับหมายถึงหมายเลขของเอกสาร (Snippet) ที่อยู่ภายในกลุ่มนี้ สามารถนำมาแสดงผลได้ดังรูปที่ 3. 4



รูปที่ 3. 4 หน้าจอตัวอย่างการแสดงผลของระบบ

จากรูปที่ 3. 4 คือรูปแบบการแสดงผลของระบบ มีการทำงาน 3 ส่วนหลักได้แก่

1. **Search Panel** – ส่วนของการสืบค้นข้อมูล โดยมีส่วนที่ให้ผู้ใช้งานกรอกคำที่ต้องการสืบค้น (Keyword Input) กำหนดจำนวนผลการสืบค้นที่นำมาจัดกลุ่ม (Download) และเลือกที่จะจัดกลุ่มด้วยอัลกอริทึมใด (Cluster With)

2. **Cluster Panel** – ส่วนของการแสดงผลป้ายชื่อของกลุ่ม (Label) ที่ระบบจัดกลุ่มได้ โดยเรียงลำดับป้ายชื่อของกุล่มตามคะแนนที่ได้ โดยจะเรียงจากจำนวนมากสุดไปหาน้อยสุด

3. **Search Result Panel** – ส่วนที่แสดงผลของเอกสารที่อยู่ภายใต้ป้ายชื่อที่เลือกจากหน้าจอส่วนที่แสดงป้ายชื่อของกลุ่มทั้งหมด (Cluster Panel)

3.1.4 ส่วนที่ใช้ควบคุมการทำงานของระบบ (CONTROLLER COMPONENT) การทำงานในส่วนนี้ มีหน้าที่คุบควมการทำงานของระบ โดยจะติดต่อสื่อสารกันในรูปแบบเอกดสาร XML ซึ่งระบบมีการควบคุมการทำงานหลัก ได้แก่

1. **เชื่อมต่อกับแหล่งข้อมูลต่างๆ** – ส่วนของการควบคุมการเชื่อมต่อข้อมูล โดยเก็บข้อมูลการเชื่อมต่อกับระบบภายนอกจะเก็บข้อมูลการตั้งค่าไว้ในรูปแบบเอกสาร XML โดยมีรูปแบบดังตารางที่ 3. 14 และ ตารางที่ 3. 15

ตารางที่ 3. 14 Tag ของเอกสาร XML สำหรับควบคุมการเชื่อมต่อกับแหล่งข้อมูลภายนอก

Tag	ความหมาย
<sources>	Element ที่เก็บข้อมูลแหล่งข้อมูลจากภายนอกทั้งหมด
<source>	Element ที่เก็บข้อมูลแหล่งข้อมูลจากภายนอกแต่ละที่ โดยมี attribute “component-class” เพื่อเก็บข้อมูลคลาส (Class) ที่ใช้ในการเชื่อมต่อข้อมูลกับแหล่งข้อมูลนั้น และมี attribute “id” เพื่อเก็บไอดีของแหล่งข้อมูลนั้น
<label>	Element ที่เก็บข้อมูลป้ายชื่อของแหล่งข้อมูลเพื่อใช้แสดงผล
<title>	Element ที่เก็บข้อมูลชื่อของแหล่งข้อมูล
<description>	Element ที่เก็บข้อมูลรายละเอียดของแหล่งข้อมูล

ตารางที่ 3. 15 รูปแบบเอกสาร XML สำหรับควบคุมการเชื่อมต่อกับแหล่งข้อมูลภายนอก

```
<sources>
    <source component-class="..." id="...">
      <title>...</title>
      <description>...</description>
    </source>

...

<sources>
```

ตารางที่ 3. 16 ตัวอย่างเอกสาร XML สำหรับควบคุมการเชื่อมต่อกับแหล่งข้อมูลภายนอก

```
<sources>
    <source component-class="org.
     WebSearchClustering.source.WebDocumentSource" id="web">
       <title>Search the Web with etools.ch</title>
       <description>Searches the web using the etools.ch
       meta search engine</description>
    </source>
<sources>
```

2. **อัลกอริทึมที่ใช้จัดกลุ่ม** – ส่วนของการควบคุมอัลกอริทึมการจัดกลุ่ม โดยจะเก็บข้อมูลอัลกอริทึมที่ใช้จัดกลุ่มในรูปแบบเอกสาร XML โดยมีรูปแบบดังตารางที่ 3. 17 และ ตารางที่ 3. 18

ตารางที่ 3. 17 Tag ของเอกสาร XML สำหรับควบคุมวิธีการจัดกลุ่ม

Tag	ความหมาย
<algorithms>	Element ที่เก็บข้อมูลแหล่งข้อมูลอัลกอริทึมที่ใช้จัดกลุ่มทั้งหมด
<algorithm>	Element ที่เก็บข้อมูลอัลกอริทึมที่ใช้จัดกลุ่มแต่ละประเภท โดยมี attribute "component-class" เพื่อเก็บข้อมูลคลาส (Class) ที่ใช้ในการจัดกลุ่มในอัลกอริทึมนั้น และมี attribute "id" เพื่อเก็บไอดีของอัลกอริทึมนั้น
<label>	Element ที่เก็บข้อมูลป้ายชื่อของอัลกอริทึมเพื่อใช้แสดงผล
<title>	Element ที่เก็บข้อมูลชื่อของอัลกอริทึม

ตารางที่ 3. 18 รูปแบบเอกสาร XML สำหรับควบคุมวิธีการจัดกลุ่ม

```
<algorithms>
    <algorithm component-class="..." id="...">
      <label>...</label>
      <title>...</title>
    </algorithm>
</algorithms>
```

ตารางที่ 3. 19 ตัวอย่างเอกสาร XML สำหรับควบคุมวิธีการจัดกลุ่ม

```
<algorithms>
    <algorithm component-
    class="org.WebSearchClustering.clustering.STCAlgorithm
```

ตารางที่ 3.19 (ต่อ)

```
id="stc">
      <label>STC</label>
      <title>Suffix Tree Clustering</title>
    </algorithm>

    <algorithm      component-class="org.WebSearchClustering
    .clustering.LingoAlgorithm" id="lingo">
      <label>Lingo</label>
      <title>Lingo Clustering</title>
    </algorithm>
</algorithms>
```

การแสดงผลลัพธ์การจัดกลุ่ม – ส่วนของการควบคุมการแสดงผลลัพธ์การจัดกลุ่ม โดยจะเก็บข้อมูลจำนวนผลการสืบค้นที่นำมาจัดกลุ่มและประเภทในการแสดงผลลัพธ์การจัดกลุ่มในรูปแบบเอกสาร XML โดยมีรูปแบบดังตารางที่ 3. 20 และ ตารางที่ 3. 21

ตารางที่ 3. 20 Tag ของเอกสาร XML สำหรับควบคุมการแสดงผลลัพธ์การจัดกลุ่ม

Tag	ความหมาย
<sizes>	Element ที่เก็บข้อมูลจำนวนผลการสืบค้นที่นำมาจัดกลุ่มทั้งหมด
<size>	Element ที่เก็บข้อมูลจำนวนผลการสืบค้นที่นำมาจัดกลุ่ม โดยมี attribute "default" เพื่อเก็บข้อมูลว่าให้รายการนั้นแสดงเป็นค่าเริ่มต้น
<views>	Element ที่เก็บข้อมูลประเภทในการแสดงผลลัพธ์การจัดกลุ่มทั้งหมด
<view>	Element ที่เก็บข้อมูลประเภทในการแสดงผลลัพธ์การจัดกลุ่ม โดยมี attribute "default" เพื่อเก็บข้อมูลว่าให้รายการนั้นแสดงเป็นค่าเริ่มต้น

ตารางที่ 3. 21 รูปแบบเอกสาร XML สำหรับควบคุมการแสดงผลลัพธ์การจัดกลุ่ม

```
  <sizes>
    <size size="..." default="true" />

    ...

  </sizes>
<views>
    <view id="..." default="true">
      <label>...</label>
```

ตารางที่ 3.21 (ต่อ)

```
    </view>

    ...

  </views>
```

ตารางที่ 3. 22 ตัวอย่างเอกสาร XML สำหรับควบคุมการแสดงผลลัพธ์การจัดกลุ่ม

```
<sizes>
   <size size="10"  />
   <size size="30"  />
   <size size="50"  />
   <size size="100" default="true" />
   <size size="150" />
   <size size="200" />
   <size size="250" />
</sizes>

<views>
   <view id="tree" default="true">
     <label>Tree</label>
   </view>
</views>
```

3.2 เครื่องมือที่ใช้ในการพัฒนาระบบ

ระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต มีการทำงานในลักษณะโปรแกรมประยุกต์บนเว็บ (Web Application) โดยมีเครื่องมือที่ใช้พัฒนาระบบงานดังนี้

- ภาษาที่ใช้ในการพัฒนา : Java (J2SE 5.0, J2EE 1.5), Java Server Pages (JSP), AJAX (JSON)
- เครื่องมือที่ใช้ในการเขียนโปรแกรม : Eclipse SDK 3.3
- เว็บเซิร์ฟเวอร์ : Apache Tomcat 5.5
- ระบบปฏิบัติการ : Window XP
- เว็บบราส์เซอร์ : Internet Explorer, Mozilla Firefox 2.2, Google Chrome, Opera

บทที่ 4

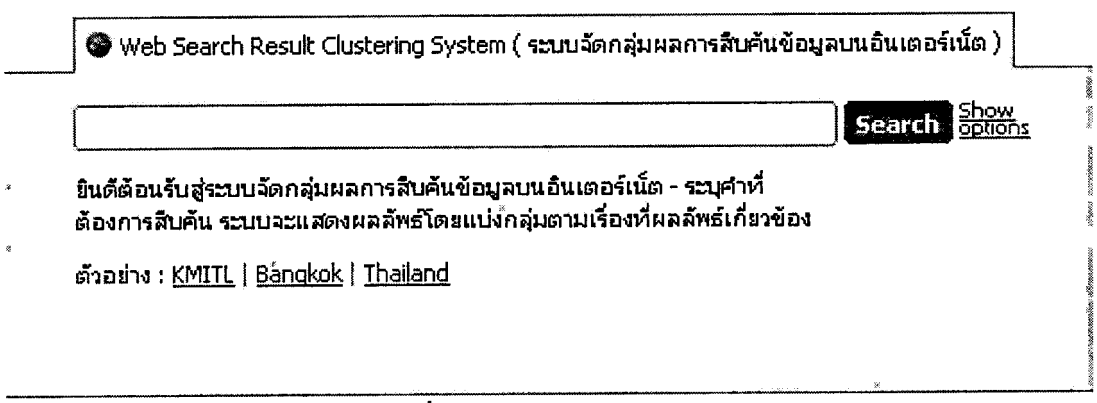
การทำงานของระบบ

ในบทนี้จะกล่าวถึงหน้าจอการทำงานทั้งหมดของระบบและการแสดงผลลัพธ์ที่ได้จากการจัดกลุ่มข้อมูลรวมถึงวิธีการใช้งานของระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต

4.1 หน้าจอการทำงานของระบบ

ระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต มีรูปแบบการทำงานบนพื้นฐานของโปรแกรมประยุกต์บนเว็บ (Web Application) สามารถทำงานบนเว็บบราวส์เซอร์ (Web Browser) ที่รองรับการทำงานของจาวาสคริปต์ (JavaScript) และการทำงานในรูปแบบเอแจ็กซ์ (Asynchronous JavaScript And XML หรือ AJAX) เช่น Internet Explorer เวอร์ชัน 5.5 ขึ้นไป , Mozilla Firefox 2.2, Google Chrome เป็นต้น โดยระบบมีหน้าจอการทำงานดังนี้

4.1.1 หน้าจอหลักของระบบ



รูปที่ 4. 1 หน้าจอหลักของระบบ

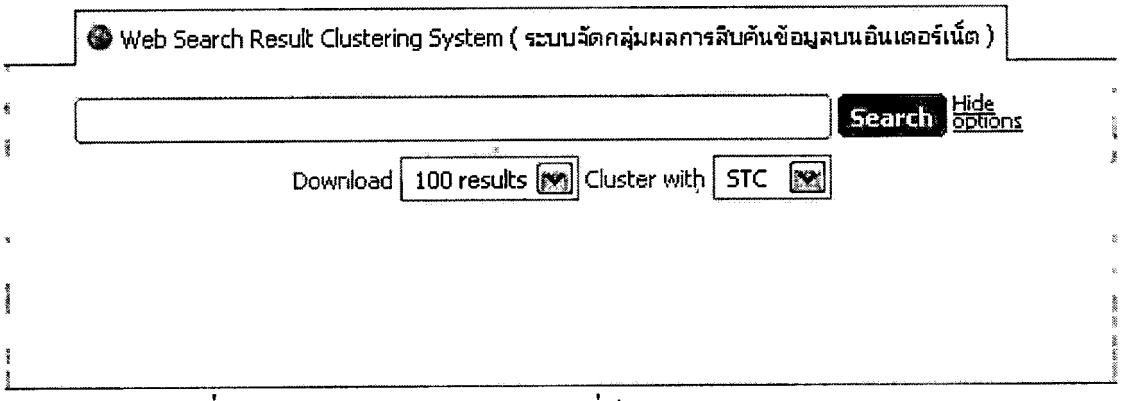
หน้าจอหลักของระบบ รูปที่ 4. 1เมื่อผู้ใช้ต้องการค้นหาข้อมูล จะต้องระบุ Keyword ที่ต้องการสืบค้น ระบบสามารถปรับแต่งค่าในการสืบค้นข้อมูล โดยคลิกที่ Show options ระบบจะแสดงค่าในการปรับแต่งของระบบ ดังรูปที่ 4. 2 ได้แก่

- จำนวนผลลัพธ์ที่ต้องการนำมาจัดกลุ่ม (Download) ซึ่งมีค่าตั้งแต่ 10 30 50 100 150 200 และ 250 รายการ
- ประเภทของการจัดกลุ่ม (Cluster With) โดยมีการจัดกลุ่มได้ 2 แบบคือ

  STC คือ การจัดกลุ่มโดยใช้อัลกอริทึม Suffix Tree Clustering (STC)

Lingo คือ การจัดกลุ่มโดยใช้อัลกอริทึม Lingo

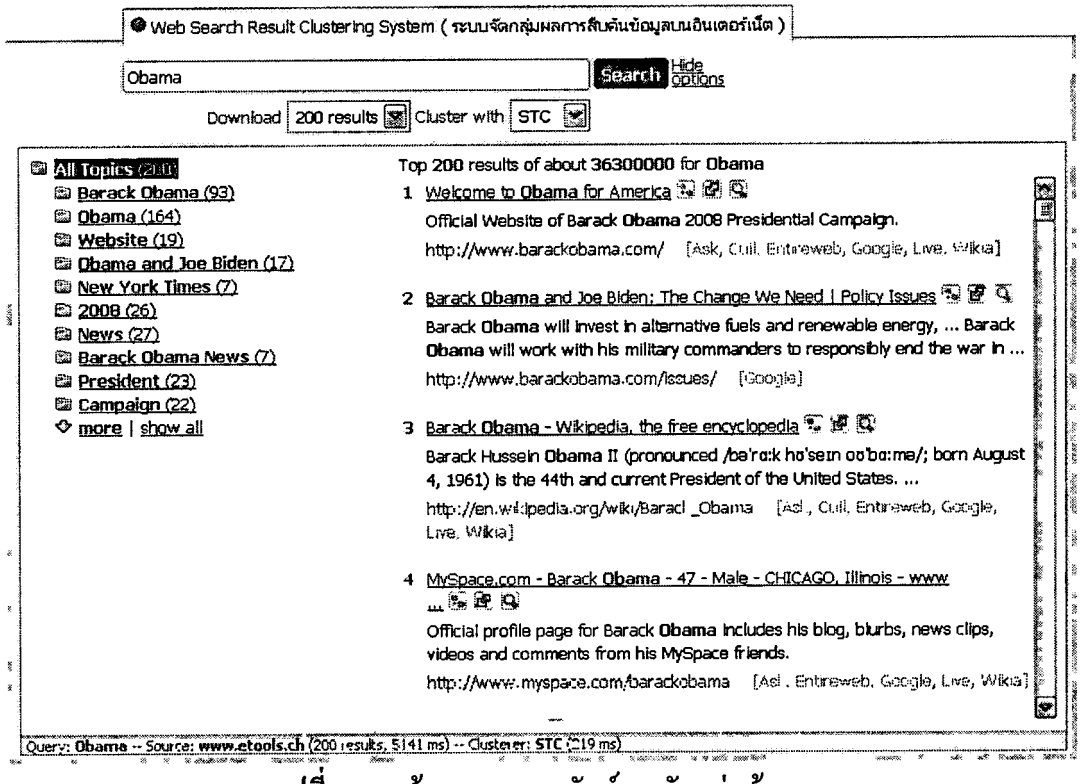
เมื่อต้องการซ่อนในส่วนของการปรับแต่งค่าในการสืบค้นและการจัดกลุ่มข้อมูล สามารถคลิกที่ Hide options ระบบจะซ่อนส่วนการปรับแต่งให้กลับมาอยู่ในรูปแบบเดิม (รูปที่ 4. 1)



รูปที่ 4. 2 หน้าจอหลักของระบบกรณีที่ต้องการปรับค่าต่างๆ ของระบบ

4.1.2 แสดงผลลัพธ์การจัดกลุ่ม

เมื่อระบุ Keyword ที่ต้องการสืบค้นและคลิกที่ปุ่ม Search ระบบจะแสดงหน้าจอแสดงรายการผลลัพธ์การสืบค้นและข้อมูลที่จัดกลุ่มได้ ดังรูปที่ 4. 3 ซึ่งในหน้าจอจะแสดงข้อมูล ดังนี้



รูปที่ 4. 3 หน้าจอแสดงผลลัพธ์การจัดกลุ่มข้อมูล

จากรูปที่ 4. 3 สามารถอธิบายรายละเอียดในส่วนต่างๆ ของหน้าจอแสดงผลลัพธ์การสืบค้นและผลการจัดกลุ่มได้ดังนี้

Top Result คือ จำนวนผลลัพธ์ที่นำมาแสดงและผลลัพธ์ทั้งหมดที่สืบค้นได้

Query คือ Keyword ที่ระบุในการสืบค้น

Source คือ แหล่งข้อมูลที่ได้ผลลัพธ์การสืบค้นมา โดยจะแสดงจำนวนผลลัพธ์และเวลาที่ในการสืบค้น

Cluster คือ จัดกลุ่มข้อมูลด้วยอัลกอริทึมใดและแสดงเวลาที่ใช้ในการจัดกลุ่มด้วย

ในส่วนหน้าจอที่แสดงผลการจัดกลุ่ม (Cluster Panel) จะแสดงหมวดหมู่ที่ระบบจัดกลุ่มได้ดังรูปที่ 4. 4 โดยจะแสดงชื่อของหมวดหมู่และจำนวนผลลัพธ์ที่อยู่ภายใต้หมวดหมู่นี้

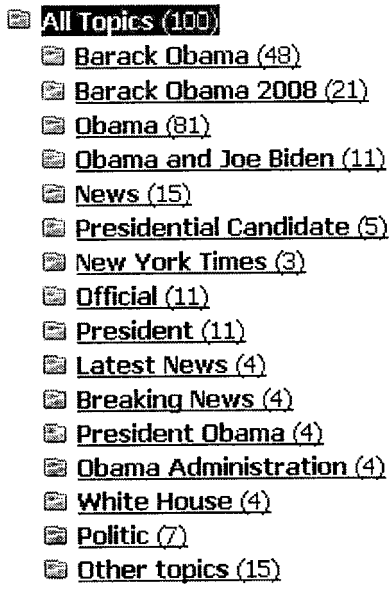
รูปที่ 4. 4 ส่วนของหน้าจอที่แสดงหมวดหมู่ที่ระบบจัดกลุ่มได้ (Cluster Panel)

จากรูปที่ 4. 4 สามารถอธิบายรายละเอียดในส่วนต่างๆ ในส่วนหน้าจอที่แสดงผลการจัดกลุ่ม (Cluster Panel) ได้ดังนี้

All Topics คือ จำนวนผลลัพธ์ลัพธ์ทึ่งหมดที่นำมาจัดกลุ่ม

ชื่อกลุ่ม คือ ชื่อกลุ่มที่ระบบจัดกลุ่มได้ และแสดงจำนวนเอกสารที่อยู่ภายในกลุ่มนี้ด้วย

More คือ แสดงชื่อกลุ่มเพิ่มเติม โดยระบบจะแสดงชุดชื่อกลุ่มที่ได้คะแนนรองลงมา ดังรูปที่ 4. 5

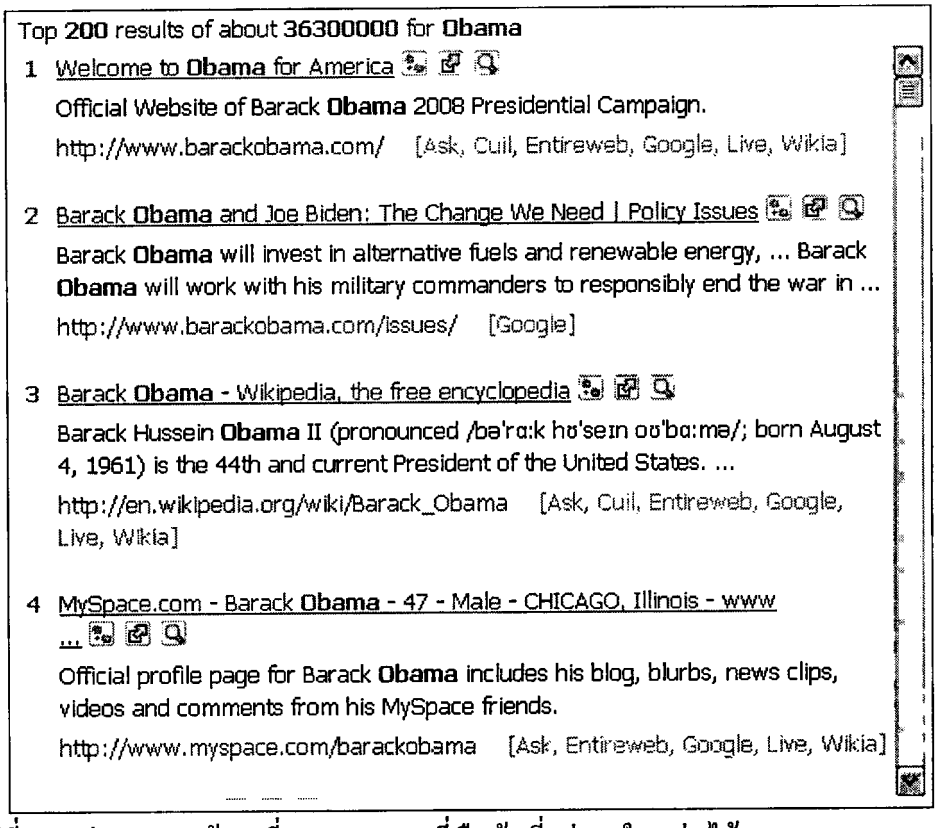


รูปที่ 4. 5 ส่วนของหน้าจอที่แสดงหมวดหมู่ที่ระบบจัดกลุ่มได้ (Cluster Panel) โดยการเลือก more

Show all คือ แสดงชื่อกลุ่มทั้งหมดที่ระบบจัดกลุ่มได้ ดังรูปที่ 4. 6 เมื่อผลลัพธ์แสดงขึ้น จะมีข้อความว่า "Always Show all Clusters" ให้เลือกในกรณีต้องการให้แสดงผลลัพธ์ในการจัดกลุ่มทั้งหมดเป็นค่าเริ่มต้น

รูปที่ 4. 6 ส่วนของหน้าจอที่แสดงหมวดหมู่ที่ระบบจัดกลุ่มได้ (Cluster Panel) โดยการเลือก Show all

ในส่วนหน้าจอแสดงรายการที่สืบค้นที่อยู่ภายในกลุ่มได้ (Panel Result Panel) ดังรูปที่ 4. 7 เป็นหน้าจอที่แสดงรายการที่สืบค้นที่อยู่ภายในกลุ่มได้



รูปที่ 4. 7 ส่วนของหน้าจอที่แสดงรายการที่สืบค้นที่อยู่ภายในกลุ่มได้ (Panel Result Panel)

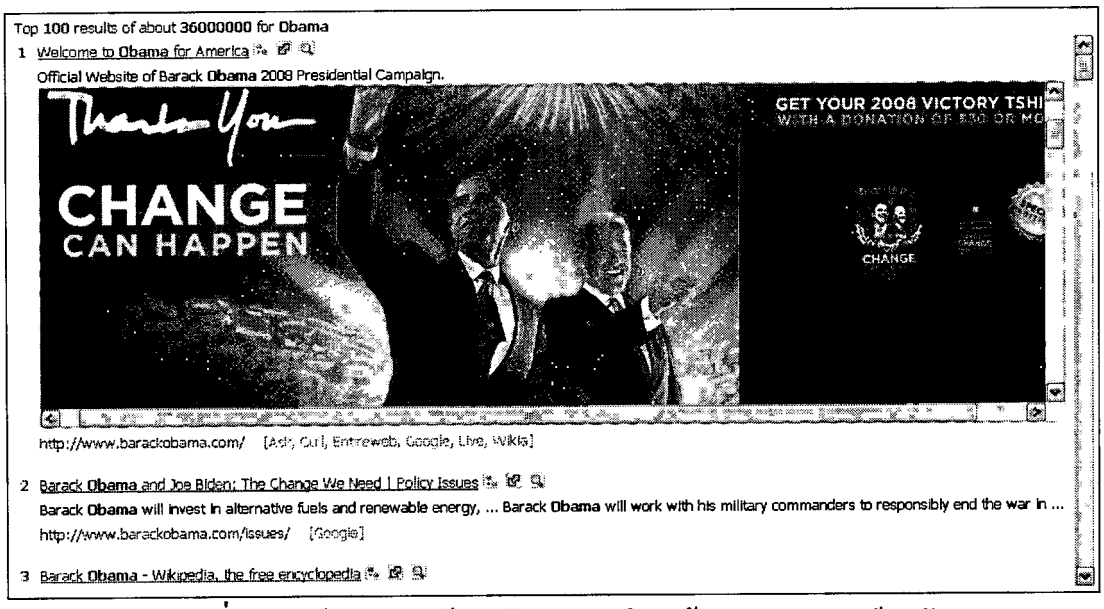
จากรูปที่ 4. 7 สามารถอธิบายรายละเอียดในส่วนต่างๆ ในส่วนของหน้าจอที่แสดงรายการที่สืบค้นที่อยู่ภายในกลุ่มได้ (Panel Result Panel) ได้ดังนี้

1. **ผลลัพธ์การสืบค้น (Search Result)** คือ ข้อมูลผลการสืบค้นแต่ละรายการ โดยแสดงข้อมูล ชื่อของรายการ (Title) คำอธิบายรายการ (Snippets) ที่อยู่ของเอกสาร (URL) แหล่งที่ส่งผลลัพธ์มา (Data Source)

2. **แสดงชื่อกลุ่มผลลัพธ์เกี่ยวข้อง** คือ การคลิกที่ชื่อของรายการ (Title) หรือไอคอน ระบบจะแสดงไฮไลท์ชื่อกลุ่มที่เกี่ยวข้องกับผลลัพธ์ที่คลิกในในส่วนหน้าจอที่แสดงผลการจัดกลุ่ม (Cluster Panel)

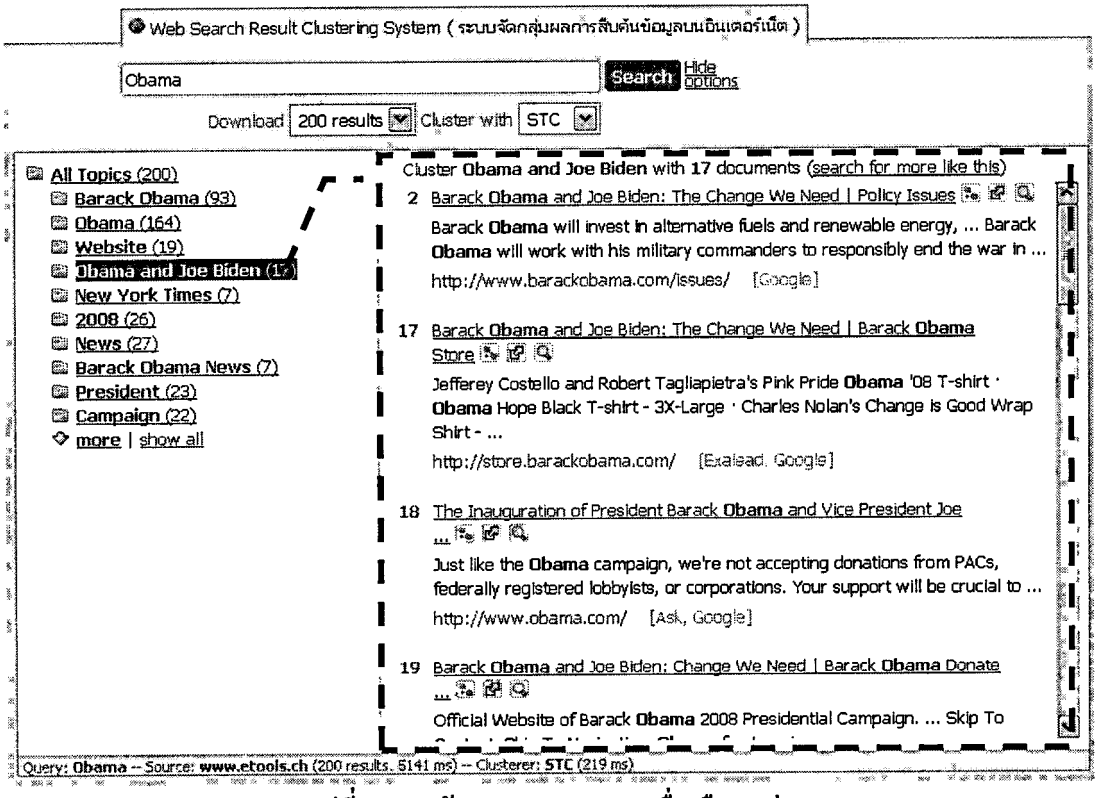
3. **แสดงราคายในหน้าต่างใหม่ (Open in new window)** คือ การเปิดเข้าไปดูข้อมูลในเอกสารนั้น โดยการเปิดหน้าต่างการทำงาน (Window) ใหม่ขึ้นมา โดยการคลิกที่ไอคอน

4. **แสดงราคายในหน้าต่างเดียวกัน (Show Preview)** คือ การเปิดเข้าไปดูข้อมูลในเอกสารนั้น โดยการแสดงข้อมูลในหน้าจอการทำงานเดิม โดยการคลิกที่ไอคอน ระบบจะแสดงเว็บเพจ (Web Page) ดังรูปที่ 4. 8



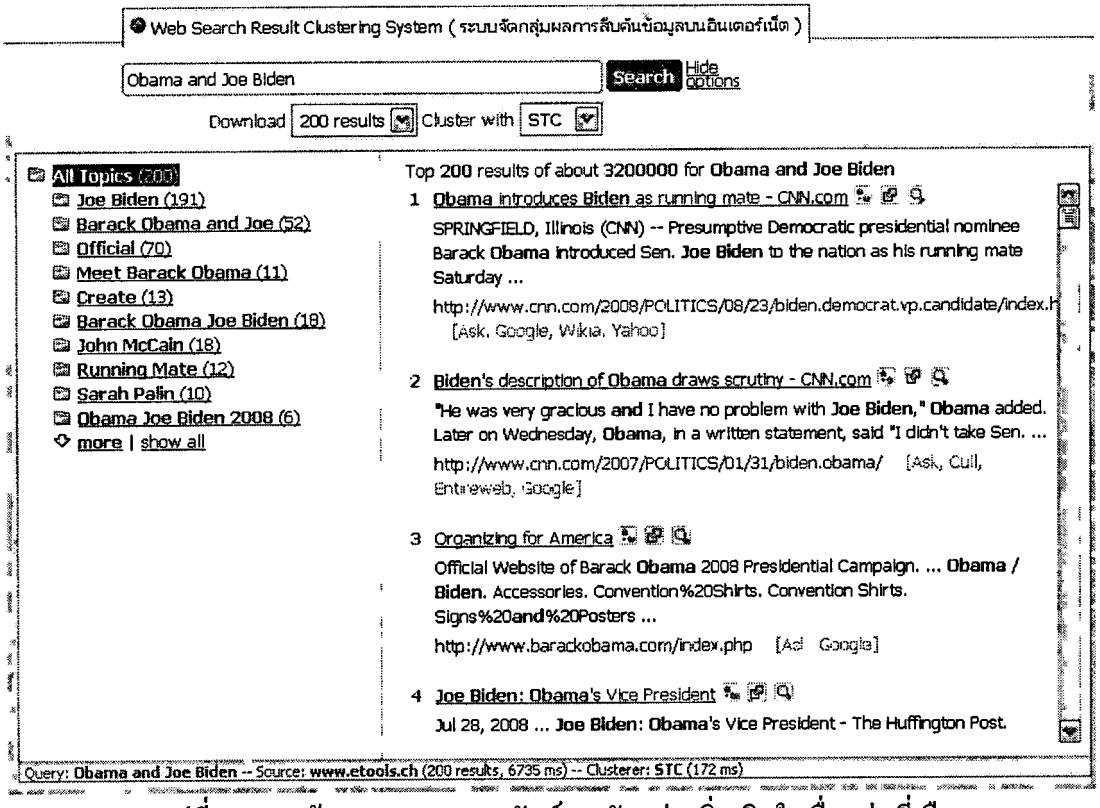
รูปที่ 4. 8 หน้าจอแสดงข้อมูลในเอกสารในหน้าจอการทำงานเดียวกัน

เมื่อเลือกหมวดหมู่ (Topics) หน้าจอด้านขวามือจะแสดงรายการผลลัพธ์ที่อยู่ภายในกลุ่มนั้นมาแสดง ดังรูปที่ 4. 9



รูปที่ 4. 9 หน้าจอแสดงการรายเมื่อเลือกกลุ่ม

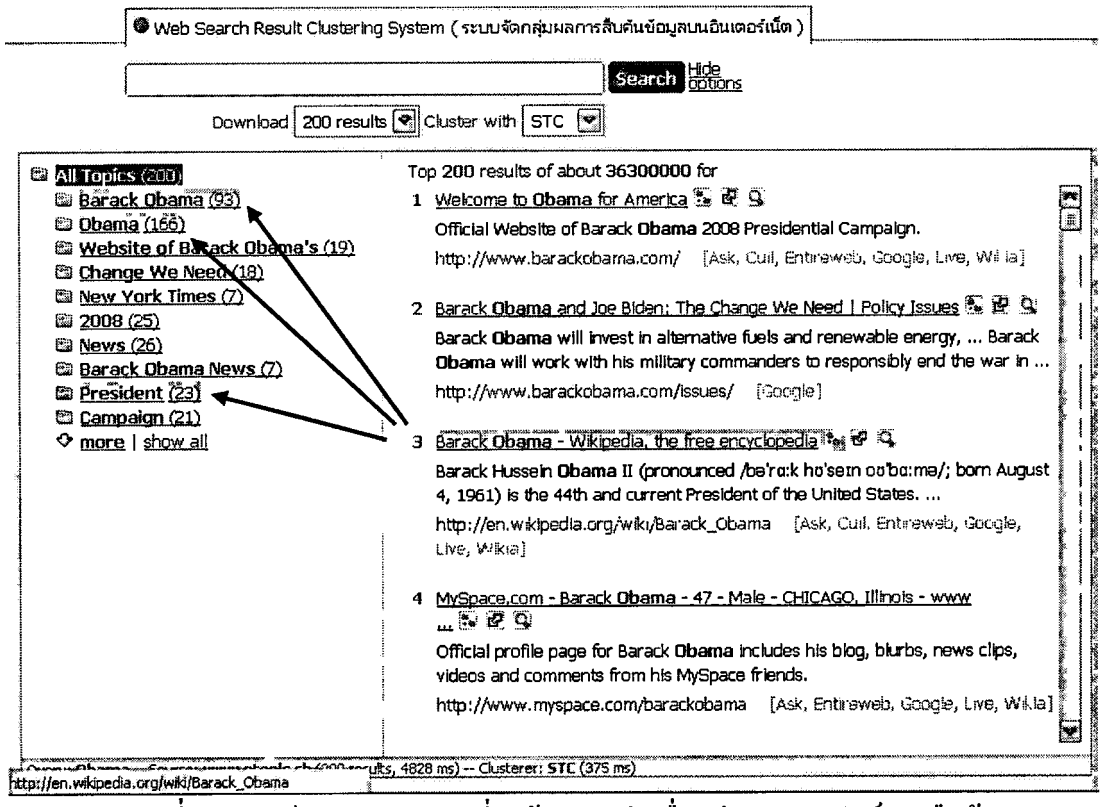
จากรูปที่ 4. 9 ระบบจะแสดงเอกสารที่อยู่ภายในกลุ่มที่เลือก และระบบยังสามารถเข้าไปค้นหาข้อมูลในชื่อกลุ่มนั้นเพิ่มเติมอีก โดยการคลิกที่ “search for more like this” ระบบจะทำการสืบค้นด้วยชื่อกลุ่มที่เลือกอีกครั้ง ดังรูปที่ 4. 10



รูปที่ 4. 10 หน้าจอแสดงการผลลัพธ์การจัดกลุ่มเพิ่มเติมในชื่อกลุ่มที่เลือก

เมื่อคลิกที่รายการผลลัพธ์ในการสืบค้นในส่วนที่แสดงผลลัพธ์ในการสืบค้น (Search Results Panel) ระบบจะแสดงไฮไลท์ (Highlight) ตรงชื่อกลุ่มที่รายการผลลัพธ์นั้นเกี่ยวข้อง ดังรูปที่ 4. 11

จากรูปที่ 4. 11 จะเห็นว่ารายการผลลัพธ์การสืบค้น 1 รายการสามารถเกี่วยข้องกับกลุ่มข้อมูลที่จัดได้มากกว่า 1 กลุ่ม ซึ่งขึ้นอยู่กับการเลือกใช้อัลกอริทึมใดในการจัดกลุ่มด้วย



รูปที่ 4. 11 หน้าจอแสดงความเกี่ยวข้องระหว่างชื่อกลุ่มและผลลัพธ์การสืบค้น

บทที่ 5
สรุปผลและข้อเสนอแนะ

ในบทนี้กล่างถึงการสรุปผลและข้อเสนอแนะ โดยนำเสนอผลการเปรียบเทียบในด้านผลลัพธ์ที่ได้จากการจัดกลุ่มข้อมูลของทั้งสองอัลกอริทึมและเวลาที่ใช้ประมวลผลในการจัดกลุ่มข้อมูล

5.1 สรุปผลโครงการพัฒนาระบบงาน

ระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต ได้นำเสนออัลกอริทึมการจัดกลุ่มข้อมูล 2 อัลกิริทึมคือซัฟฟิกทรีคลัสเตอร์ริง (Suffix Tree Clustering หรือ STC) และ ลินโก (LINGO) โดยทั้งสองอัลกอริทึมนั้นมีหลักการในการจัดกลุ่มข้อมูลที่แตกต่างกันทำให้ผลลัพธ์ที่ได้อาจจะมีข้อแตกต่างกันรวมถึงเวลาในการประมวลด้วย ดังนั้นในส่วนนี้จะขอนำเสนอการเปรียบเทียบผลโดยมีประเด็นดังนี้

5.1.1 ผลลัพธ์ที่ได้จากการจัดกลุ่ม

ในส่วนนี้เปรียบเทียบชื่อป้ายที่ได้ ว่าแต่ละอัลกอริทึมมีลักษณะอย่างไร โดยพิจารณาถึงความหมายป้ายชื่อของกลุ่มที่ได้จากการตัดสินของผู้ใช้ระบบ (Human Judgment) และจำนวนป้ายชื่อกลุ่มที่ได้ โดยกำหนดตัวอย่างคำสืบค้น (Keyword) ที่ใช้สืบค้นได้แก่ Obama, Apache, Microsoft และ Clustering จากเอกสาร (Search Results) 100 และ 200 เอกสารมีผลลัพธ์ที่ได้ดังนี้

ตารางที่ 5.1 ตารางเปรียบเทียบป้ายชื่อกลุ่มที่ได้จากคำสืบค้น "Obama" โดยกำหนดจำนวนเอกสาร 100 เอกสาร

STC	LINGO
All Topics (100) Barack Obama (48) Barack Obama 2008 (21) Obama (81) Obama and Joe Biden (11) News (15) Presidential Candidate (5) New York Times (3) Official (11) President (11) Latest News (4) Breaking News (4) President Obama (4) Obama Administration (4) White House (4) Politic (7) Other topics (15)	All Topics (100) President Barack Obama (10) Change We Need (6) Book (4) News Videos (4) White House (4) Latest News (3) McCain (3) New York Times (3) Vote (3) Facebook (2) Issue Positions (2) Post (2) Reports (2) Stimulus (2) Story (2) Swissinfo (2) T-Shirts (2) Wikipedia (2) Other topics (51)

ตารางที่ 5.2 ตารางเปรียบเทียบป้ายชื่อกลุ่มที่ได้จากคำสืบค้น "Obama" โดยกำหนดจำนวนเอกสาร 200 เอกสาร

STC	LINGO
All Topics (200) Barack Obama (92) Obama (165) Website of Barack Obama's (19) Change We Need (18) New York Times (7) 2008 (25) News (26) Barack Obama News (7) President (24) Campaign (21) Barack Hussein Obama (6) United States (7) Official (15) New (15) President Obama (8) Other topics (29)	All Topics (200) President Barack Obama (26) Barack Obama News (21) U.S. (12) Obama Biden (11) President of the United States (10) Profile (8) Hope (6) New York Times (6) News Videos (6) Vote Obama (6) Washington (6) GAME (5) Race (5) Wikipedia (5) Administration (4) Americans (4) Articles (4) Characteristics (4) Patients (4) Says (4) White House (4) Book (3) John McCain (3) PAY (3) Barack Obama Logo (2) Bush (2) Election Center 2008 (2) Facebook (2) Intrahepatic Cholangiocarcinoma (2) Israel (2) Issue Positions (2) Photo Gallery (2) Post (2) Posters (2) Proteins (2) Radical Muslim (2) Republicans (2) Sarah Palin T-Shirts (2) Stem Cell Transplantation (2) Swissinfo (2) Victory Speech (2) Other topics (74)

จากตารางที่ 5.1 และตารางที่ 5.2 พบว่าป้ายชื่อที่ได้จากทั้งสองอัลกอริทึมนั้น อัลกอริทึม LINGO ได้ป้ายชื่ออันดับแรกว่า "President Barack Obama" ซึ่งมีความหมายดีกว่าอัลกอริทึม STC ที่ได้ป้ายชื่อกลุ่ม "Barack Obama" และมีป้ายชื่อกลุ่มลำดับท้ายๆ ที่มีชื่อว่า "President" ซึ่งยังไม่สามารถสื่อความหมายของป้ายชื่อกลุ่มได้ดีเท่าอัลกอริทึม LINGO

ในส่วนของจำนวนป้ายชื่อกลุ่มที่ได้และเอกสารที่อยู่ภายใต้ป้ายชื่อกลุ่มนั้นพบว่าเมื่อกำหนดจำนวนเอกสารที่นำมาจัดกลุ่มโดยมีจำนวน 100 และ 200 เอกสาร อัลกอริทึม STC ได้จำนวนป้ายชื่อกลุ่มเท่ากันแต่จำนวนเอกสารที่อยู่ภายใต้ป้ายชื่อกลุ่ม

ตารางที่ 5.3 ตารางเปรียบเทียบป้ายชื่อกลุ่มที่ได้จากคำสืบค้น “Apache” โดยกำหนดจำนวนเอกสาร 100 เอกสาร

STC	LINGO
All Topics (100) Apache Web Server (14) Open Source (14) Apache Software Foundation (9) Server (19) Software (17) Case (8) Open Source Software (3) Installed (11) Developed (11) Project (10) Information (9) Module (9) Linux (7) System (7) PHP (7) Other topics (30)	All Topics (100) Apache Web Server (12) Open Source (12) Apache Software Foundation (10) Module for the Apache (8) Apache Projects (7) Install Apache (6) Build (5) Development (5) Download (5) PHP and MySQL (5) Mac OS X (4) Test (3) Apache Compile HOWTO (2) Apache County (2) Apache News (2) Apache Point Observatory (2) Apache Tomcat (2) Apache Week (2) Attack Helicopter (2) Gallery Creates an Thumbnail Index (2) Intensive (2) New Mexico (2) Prediction of Outcome in Acute PANCREATITIS (2) Security (2) Other topics (33)

ตารางที่ 5.4 ตารางเปรียบเทียบป้ายชื่อกลุ่มที่ได้จากคำสืบค้น “Apache” โดยกำหนดจำนวนเอกสาร 200 เอกสาร

STC	LINGO
All Topics (192) Apache Web Server (28) Open Source (21) Server (36) Apache Software Foundation (8) White Mountain Apache (7) Intensive care Unit (7) Project (17) Installed (17) Software (16) Easy to Install (4) Developed (15) Module (14) Sites (13) Information (13) Apache II Score (8) Other topics (67)	All Topics (192) Apache Server (42) Apache Web Server (25) Apache Project (16) Apache Module (15) Open Source (15) Apache 2 (14) Apache Software Foundation (12) Apache Webserver (9) Apache Tomcat (7) Apache History (6) Apache II Score (6) Apache License (6) Apache News (6) Apache Security (6) Intensive (6) PHP and MySQL (6) Apache Week (5) Blog (5) Release of Apache (5) PHP and Perl (4) Wikipedia (4) API Application (3) Mod_perl (3) Native Americans (3) Red Hat Linux (3) Technology (3) Test Page for Apache Installation (3) Western Apache (3) Apache 1.3 (2) Apache Cocoon (2) Apache Design Solutions (2) Apache Junction (2) Gallery Creates an Thumbnail Index (2) Microsoft Windows (2) Open Source Software Development (2)

ตารางที่ 5.4 (ต่อ)

STC	LINGO
	Physiology and Chronic Health Evaluation (2) San Carlos (2) Verify the Integrity (2) White Mountain Apache Tribe (2) Other topics (51)

จากตารางที่ 5.3 - ตารางที่ 5.4 พบว่าหากใช้คำสืบค้น "Apache" ป้ายชื่อกลุ่มที่จัดกลุ่มได้ในอันดับต้นๆ นั้นมีความคล้ายคลึงกัน เช่น Apache Server, Open Source, Apache Software Foundation เป็นต้น แต่ในช่วงลำดับกลางของจำนวนป้ายชื่อกลุ่มที่ได้อัลกอริทึม LINGO จะได้ป้ายชื่อกลุ่มที่ที่หลากหลายและสื่อความหมายได้กว่าอัลกอริทึม STC

ตารางที่ 5. 5 ตารางเปรียบเทียบป้ายชื่อกลุ่มที่ได้จากคำสืบค้น "Microsoft" โดยกำหนดจำนวนเอกสาร 100 เอกสาร

STC	LINGO
All Topics (100) Microsoft (89) Microsoft Office (9) Microsoft Windows (10) Windows (17) Microsoft Download Center (3) Microsoft Windows Update (4) Updates (12) Download (10) Software (9) Latest (9) Product (9) Support (8) Microsoft Research (4) Use (7) Web (7) Other topics (7)	All Topics (100) Microsoft Corporation (12) Microsoft Office (9) Windows Update (5) Microsoft Silverlight (4) Download Center (3) Excel (3) Microsoft Solutions (3) Program (3) Live Search (2) Microsoft Developer Network (2) Microsoft Game Studios (2) Microsoft Help and Support (2) Microsoft Research Cambridge (2) November 2008 (2) Real World (2) Software Company (2) Startup (2) Terms of Use Trademarks Privacy Statement (2) Train Simulator (2) Other topics (45)

ตารางที่ 5.6 ตารางเปรียบเทียบป้ายชื่อกลุ่มที่ได้จากคำสืบค้น "Microsoft" โดยกำหนดจำนวนเอกสาร 200 เอกสาร

STC	LINGO
All Topics (200) Microsoft Office (20) Microsoft Train Simulator (10) Windows (33) United States (8) Updates (20) Microsoft Corporation (12) Microsoft Windows Update (6) Microsoft Windows (17) Software (16) Dozens of Languages (3) Download (15) Web (15) Microsoft Research (8) Use (14) Sites (13) Other topics (83)	All Topics (198) Microsoft Windows (35) Microsoft Office (21) Microsoft Corporation (20) Microsoft Update (11) Microsoft Corp (7) Resources (7) Microsoft Company (6) Microsoft Excel (6) Microsoft Solutions (5) Microsoft Train (5) Network (5) Microsoft CRM (4) Microsoft Silverlight (4) V. Microsoft (4) Download Center (3) Help and Support (3) Microsoft Games (3) Microsoft Geek Blogger (3) Microsoft Partner Program (3) Software Developer (3) Blogs and Learn (2) High Quality (2) Hosting (2) Internet Explorer (2) Microsoft Corporation MSFT (2) Microsoft Research Cambridge (2) Microsoft Weblog (2) Microsoft and Citrix (2) News and Analysis (2) Novell (2) November 2008 (2) Operating System (2) Real World (2) Retail Management (2) SQL Server (2) Says (2) Small Business (2) Tutorial (2) Visual Studio (2) Vulnerabilities (2) Other topics (64)

จากตารางที่ 5. 5 และ ตารางที่ 5.6 เมื่อใช้คำสืบค้น "Microsoft" พบว่าป้ายชื่อกลุ่มที่ได้จากอัลกอริทึม LINGO มีความหลากหลายมากกว่าอัลกอริทึม STC เช่น "Microsoft Solution", "Microsoft Partner Program" เป็นต้น ทำให้ผู้ใช้สามารถพบป้ายชื่อกลุ่มใหม่ๆ ซึ่งอาจส่งผลให้พบข้อมูลที่ต้องการสืบค้นได้สะดวกและรวดเร็วยิ่งขึ้น

ตารางที่ 5. 7 ตารางเปรียบเทียบป้ายชื่อกลุ่มที่ได้จากคำสืบค้น "Clustering" โดยกำหนดจำนวนเอกสาร 100 เอกสาร

STC	LINGO
All Topics (100) Cluster (90) Cluster Analysis (9) Servers MyriNet Clustering Connectivity (3) Windows Server (6) Open Source (6) Document Clustering (6) Server (13) Automated Tag Clustering (3) Used (11) Search (11) Engine (7) Cluster Search (4) Groups (9) Methods (8) Algorithms (8) Other topics (3)	All Topics (100) Data Clustering (9) Cluster Analysis (8) Search Engine (7) Clustering is Used (6) Linux (6) Windows Server (6) Clustering Methods (5) Spectral Clustering (5) Advanced (4) Metasearch Engine (4) Automated Tag Clustering (3) Encyclopedia (3) Enterprise Search (3) Fuzzy Clustering (3) Hosting (3) Ideas (3) Load Balancing (3) Open Source Clustering Software (3) Dedicated Servers (2) Detection (2) Extreme Integration (2) Failover Clustering (2) Hardware (2) Single Computer (2) Text (2) Other topics (35)

ตารางที่ 5. 8 ตารางเปรียบเทียบป้ายชื่อกลุ่มที่ได้จากคำสืบค้น "Clustering" โดยกำหนดจำนวนเอกสาร 200 เอกสาร

STC	LINGO
All Topics (189) Cluster (166) Cluster Analysis (16) Document Clustering (14) Document Clustering Engine (4) Clustering Meta Search Tool (4) Used (24) Hierarchical Clustering (9) Data (21) Clustering Search Engine (5) Search (17) Computer (17) Clustering Engine (7) Windows Server 2008 (3) German Corpora Using Natural (4) Methods (16) Other topics (14)	All Topics (200) Data Clustering (17) Clustering Methods (12) Cluster Analysis (11) Document Clustering (11) Hierarchical Clustering (10) Clustering Solutions (9) Groups (8) Linux Clustering (8) Fuzzy Clustering (7) Metasearch Engine (7) Search Results (7) Windows Server (7) Spectral Clustering (6) Cluster Computing (5) Clustering Models (5) Encyclopedia (5) Mixture Modeling (5) Clustering Problems (4) K-means (4)

ตารางที่ 5.8 (ต่อ)

STC	LINGO
	Load Balancing (4) Open Source Software (4) Research (4) Automated Tag Clustering (3) Clustering Routines (3) Clusty (3) Dedicated Servers (3) Detection (3) Distributional Clustering (3) Expression (3) Information Retrieval (3) Model-based Clustering (3) MySQL Clustering (3) Objects (3) Variable (3) Windows Server 2008 (3) iBoogie (3) Beowulf Cluster (2) Conceptual Clustering (2) Galaxy (2) Networking and Storage (2) Part 1 (2) Semantic Clustering (2) Terracotta (2) Words (2) Other topics (61)

จากตารางที่ 5. 7 - ตารางที่ 5. 8 การจัดกลุ่มข้อมูลด้วยคำสืบค้น “Clustering” พบว่าผลลัพธ์ที่ได้มีลักษณะคล้ายคลึงกับการจัดกลุ่มโดยใช้คำสืบค้น “Microsoft” คือป้ายชื่อกลุ่มที่ได้จากอัลกอลิทึม LINGO มีความหลากหลายมากกว่าอัลกอริทึม STC เช่น “Hierarchical Clustering”, ”Clustering Solution”, ”Fuzzy Clustering”, ”Distributional Clustering” เป็นต้น ซึ่งป้ายชื่อกลุ่มเหล่านี้อาจจะทำให้ผู้ใช้งานสามารถพบข้อมูลที่ตนต้องการได้อย่างสะดวกและรวดเร็ว

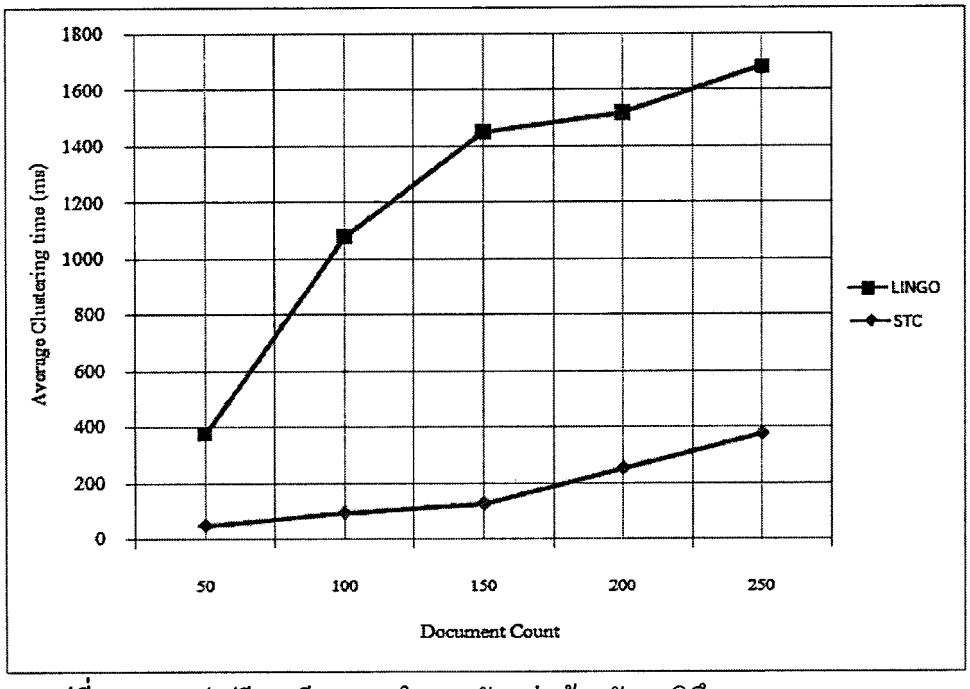
สิ่งหนึ่งที่สามารถบอกถึงความถูกต้องของป้ายชื่อกลุ่มได้คือจำนวนเอกสารที่เกี่ยวข้องกับป้ายชื่อกลุ่มนั้น ว่ามีจำนวนที่เกี่ยวข้องมากน้อยเพียงใดเมื่อเปรียบเทียบกับอีกอัลกอริทึมด้วยจำนวนเอกสารที่นำมาจัดกลุ่มเท่ากันจากตารางที่ 5.1 ถึงตารางที่ 5. 8 จะพบว่าเมื่อกำหนดจำนวนรายการผลสืบค้น 100 รายการเพื่อนำมาจัดกลุ่ม จำนวนป้ายชื่อกลุ่มที่ได้นั้นมีจำนวนใกล้เคียงกันทั้งสองอัลกอริทึม แต่เมื่อเพิ่มจำนวนเอกสาร (ผลการสืบค้นข้อมูลบนอินเตอร์เน็ต) ที่จะนำมาจัดกลุ่มเป็น 200 รายการ จำนวนป้ายชื่อกลุ่มที่ได้นั้นมีความแตกต่างกันมากเนื่องจากลักษณะการจัดกลุ่มของ STC จะใช้การเปรียบเทียบวลีที่มีใช้ร่วมกันระหว่างผลลัพธ์จากการสืบค้นที่ได้ ซึ่งรายการที่ได้มานั้นก็ยังอาจจะมีวลีที่คล้ายกันทำให้จำนวนกลุ่มที่ได้อาจใกล้เคียงกับการกำหนดจำนวนเอกสาร 100 รายการได้ แต่ LINGO นั้นมีหลักการในการจัดกลุ่มโดยเริ่มจากการหาวลีหรือคำที่ใช้ป้ายชื่อกลุ่มที่มีความหมายก่อน จากนั้นจึงนำวลีหรือคำที่ได้คัดเลือกมาเป็นป้ายชื่อกลุ่มมา

ค้นหาเอกสารที่เกี่ยวข้องโดยใช้หลักการ Singular Value Decomposition (SVD) และเมื่อมีจำนวนเอกสารที่นำมาจัดกลุ่มเพิ่มมากขึ้น จึงทำให้เกิดการได้ป้ายชื่อกลุ่มใหม่ๆ เพิ่มขึ้นตามไปด้วย

ด้วยเหตุนี้เมื่อมีผลลัพธ์ในการจัดกลุ่มเพิ่มขึ้น แต่จำนวนกลุ่มยังมีจำนวนไม่ต่างจากเดิมมากนั้น ส่งผลจำนวนเอกสารที่เกี่ยวข้องกับป้ายชื่อกลุ่มนั้นเพิ่มขึ้นเรื่อยๆ ทำให้ป้ายชื่อกลุ่มมีจำนวนเอกสารที่เกี่ยวขอ้องมากขึ้น ส่งผลให้ป้ายชื่อกลุ่มนั้นไม่สามารถจะสื่อความหมายของกลุ่มได้ครอบคลุมทุกเอกสาร

5.1.2 เวลาที่ใช้ในการจัดกลุ่ม

การเปรียบเทียบเวลาที่ใช้ในการจัดกลุ่มของทั้งสองอัลกอริทึม โดยจะเปรียบเทียบเพียงเวลาการจัดกลุ่มเท่านั้น ไม่รวมถึงเวลาที่แหล่งข้อมูลภายนอกส่งผลการสืบค้นกลับมายังระบบ โดยจะเริ่มนับเวลา ณ ขณะที่ระบบรับผลการสืบค้นมาทั้งหมดแล้ว รวมถึงสภาพแวดล้อมในระบบที่เหมือนกัน (Environment) ดังนี้ Intel Core Duo T5600 1.83GHz, 1.5GB MB RAM, Windows XP, Java Virtual Machine: Sun JDK 1.6.0 ในการทดสอบจะใช้ตัวอย่างคำสืบค้นจำนวน 100 คำ โดยจะกำหนดจำนวนเอกสาร (ผลการสืบค้นข้อมูลบนอินเตอร์เน็ต) เพื่อนำมาจัดกลุ่มได้แก่ 50 100 150 และ 200 และวัดเวลาเฉลี่ยในการจัดกลุ่มข้อมูลทั้งสองอัลกอริทึม โดยสามารถแสดงในรูปแบบกราฟดังรูปที่ 5. 1



รูปที่ 5. 1 กราฟเปรียบเทียบเวลาในการจัดกลุ่มด้วยอัลกอริทึม STC และ LINGO

จากรูปที่ 5. 1 แกนแนวตั้งของกราฟแสดงถึงเวลาในการจัดกลุ่มข้อมูล และแกนแนวนอนแสดงจำนวนเอกสารที่ใช้จัดกลุ่มข้อมูล โดยข้อมูลการกราฟแสดงให้เห็นว่าอัลกอริทึม STC ใช้เวลาในการจัดกลุ่มน้อยกว่าอัลกอริทึม LINGO อย่างเห็นได้ชัด เนื่องจากกระบวนการทำงานของ

STC ที่ไม่ซับซ้อนมากนักเมื่อเปรียบเทียบกับ LINGO ที่ใช้กระบวนการที่ซับซ้อนมากกว่า นั่นคือ กระบวนการหาป้ายชื่อกลุ่มที่ LINGO ให้ความสำคัญมากที่สุด โดย STC ใช้การหาวลี (Phase) ที่กลุ่มเอกสารใช้ร่วมกันมากที่สุดมาเป็นป้ายชื่อกลุ่ม ส่วน LINGO ใช้วิธีการ Singular Value Decomposition (SVD) ที่มีขั้นตอนการประประมวลผลที่ซับซ้อน เช่น การสร้าง Decompose term document matrix หรือ การหาป้ายชื่อของกลุ่มด้วยวิธี cosine similarity เป็นต้น

จากการเปรียบเทียบผลการทำงานจะเห็นได้ว่าทั้งสองอัลกอริทึมมีทั้งข้อดีและข้อด้อยต่างกันได้แก่ ข้อดีของ STC คือใช้เวลาเฉลี่ยในการจัดกลุ่มเร็วกว่า LINGO แต่ในทางกลับกัน LINGO สามารถสร้างป้ายชื่อกลุ่มที่สื่อความหมายให้กับผู้ใช้งานได้ดีกว่า STC ซึ่งเป็นสิ่งที่ LINGO ให้ความสำคัญมากที่สุด แต่สิ่งที่ควรพิจารณาอีกเรื่องคือ จำนวนเอกสารที่ใช้จัดกลุ่มหากมีจำนวนไม่มากพอ ผลลัพธ์การจัดกลุ่มเอกสารที่ได้อาจจะไม่พบข้อแตกต่างกันอย่างชัดเจนได้

5.2 ผลการดำเนินการพัฒนาระบบ

ระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต ช่วยแก้ปัญหาการแสดงผลข้อมูลผลการสืบค้นในอินเตอร์เน็ตซึ่งมีจำนวนมาก ทำให้เกิดความยุ่งยากกับผู้ค้นหา ระบบนี้จึงช่วยให้การสืบค้นข้อมูลจัดอยู่ในรูปแบบการสืบค้นตามเนื้อหาที่ผลนั้นๆ เกี่ยวข้องและลำดับของผลการสืบค้นจะช่วยให้การแสดงผลมีประสิทธิภาพมากยิ่งขึ้น และยังเพิ่มโอกาสให้ผลการสืบค้นในลำดับท้ายๆ ถูกนำมาใช้เพิ่มมากขึ้น

5.3 ข้อจำกัดและข้อเสนอแนะ

จากการศึกษาและพัฒนาระบบจัดกลุ่มผลการสืบค้นข้อมูลบนอินเตอร์เน็ต พบว่ายังคงมีข้อจำกัดอยู่ แต่สามารถเพิ่มคุณสมบัติบางประการ เพื่อลดข้อจำกัดและพัฒนาให้ระบบมีประสิทธิภาพเพิ่มสูงขึ้นได้ โดยสามารถสรุปประเด็นต่างๆ ได้ดังนี้

1. การจัดกลุ่มคำในกรณีเอกสารที่เนื้อหาเป็นภาษาไทย เนื่องจากคำแต่ละคำในภาษาไทยนั้น แต่ละประโยคมักมีลักษณะติดกัน จึงยากแก่การที่จะแยกคำแต่ละคำออกมาได้อย่างถูกต้องและสมบูรณ์ จึงทำให้ผลลัพธ์ของการจัดกลุ่มอาจมีความคลาดเคลื่อนได้ แต่สามารถนำวิธีการจำแนกคำภาษาไทยที่มีประสิทธิภาพเข้ามาร่วมประยุกต์ใช้ได้ เพื่อเพิ่มประสิทธิภาพในการจัดกลุ่มเอกสารให้สามารถรองรับภาษาไทยได้อย่างถูกต้อง เช่น ความหมายของป้ายชื่อกลุ่มเป็นต้น

2. การเพิ่มประสิทธิภาพในการจัดกลุ่มของอัลกอลิทึม LINGO เนื่องจากยังมีข้อด้อยในเรื่องความเร็วในการประมวลผล ในส่วนการทำงานสำคัญคือการทำ Singular Value Decomposition (SVD)

3. ในอนาคตสามารถปรับปรุงและพัฒนาระบบให้รองรับระบบจากภายนอกสามารถเรียกใช้บริการการจัดกลุ่มข้อมูลได้

บรรณานุกรม

พรฤดี เนติโสภากุลและวิริศ ลีลาภัทร. 2548. **สถาปัตยกรรมเว็บเสิร์ชเอนจิ้นที่สามารถปรับตามความต้องการของผู้ค้นหา. วารสารวิชาการเนคเทค** 6, 17 (พ.ย. 2548 - มิ.ย. 2550 2550). 19 -26

R. Baeza-Yates and B. Ribeiro-Neto. 1999. **Modern Information Retrieval.** Addison Wesley

Stanislaw Osilski. 2003. **An Algorithm for Clustering of Web Search Results.** Master thesis, Department of Computing Science, Poznal University of Technology

Stanislaw Osilski and Dawis Wiess. 2004. **Conceptual Clustering Using Lingo Algorithm: Evaluation on Open Directory Project Data.** Advances in Soft Computing, Intelligent Information Processing and Web Mining, Proceedings of the International IIS: IIPWM´04 Conference, Zakopane, Poland, 2004. 369-378

Stanislaw Osilski and Dawis Wiess. 2005. **A Concept-Driven Algorithm for Clustering Search Results.** IEEE Intelligent Systems, May/June, 3 (vol. 20), 2005. 48-54.

Zamir, O and Etzioni O. 1997. **Fast and intuitive clustering of web documents.** In Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining

Zamir, O and Etzioni O. 1998. **Web Document Clustering: A Feasibility Documentstration.** Department of Computer Science and Engineering

Zeng, H. et.al. 2004. **Learning to Cluster Web Search Result.** [Online]. Available: http://research.microsoft.com/user/hjzeng/p230-zeng.pdf

ประวัติผู้เขียน

ชื่อผู้จัดทำโครงงาน	นายพิศาล สรสิทธิ์
วันเดือนปีเกิด	15 พฤศจิกายน 2526
สถานที่เกิด	พิษณุโลก
ประวัติการศึกษา	
มัธยมศึกษา	โรงเรียนบางบ่อวิทยาคม
อุดมศึกษา	ปวส. คอมพิวเตอร์ธุรกิจ มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร วิทยาเขตพณิชยการพระนคร
	วทบ. วิทยาการคอมพิวเตอร์ มหาวิทยาลัยมหิดล