การแยกเวบเพจภาษาไทยให้เป็นหมวดหมู่แบบอัตโนมัติ

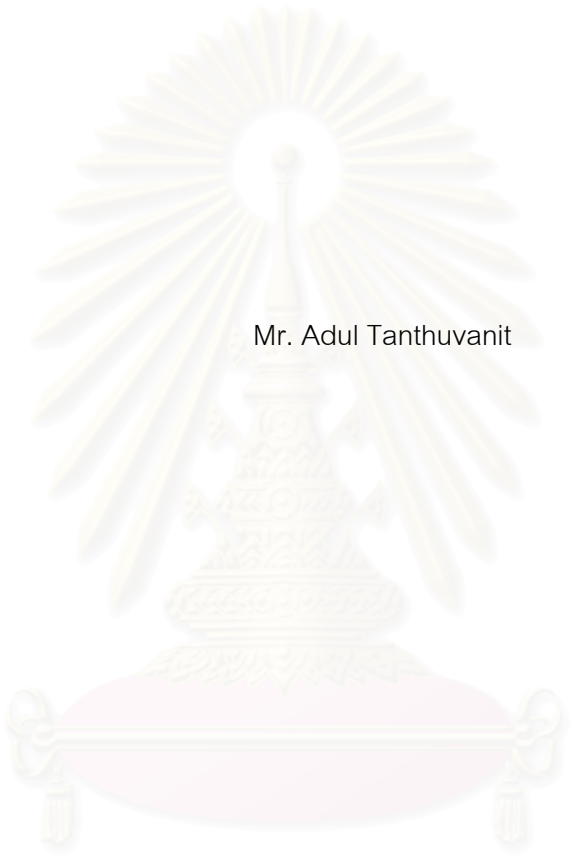
นาย อดุลย์ ตันธุวนิตย์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2545

ISBN 974-17-1228-6

AUTOMATIC THAI WEB PAGE CATEGORIZATION

Mr. Adul Tanthuvanit



A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science in Computer Science

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2002

ISBN 974-17-1228-6

หัวข้อวิทยานิพนธ์	การแยกเวบเพจภาษาไทยให้เป็นหมวดหมู่แบบอัตโนมัติ
โดย	นายอดุลย์ ตันธุวนิตย์
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

…….. คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

…….. ประธานกรรมการ
(รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์)

…….. อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล)

…….. กรรมการ
(อาจารย์ ดร.ยรรยง เต็งอำนวย)

…….. กรรมการ
(อาจารย์ นครทิพย์ พร้อมพูล)

อดุลย์ ตันธุวนิตย์ : การแยกเวบเพจภาษาไทยให้เป็นหมวดหมู่แบบอัตโนมัติ (AUTOMATIC THAI WEB PAGE CATEGORIZATION) อ. ที่ปรึกษา : ผศ. ดร.บุญเสริม กิจศิริกุล, 49 หน้า. ISBN 974-17-1228-6.

ในปัจจุบันนี้เอกสารหรือเวบเพจบนอินเตอร์เน็ตเพิ่มขึ้นอย่างรวดเร็ว ทำให้การค้นหาเอกสารที่ต้องการทำได้ยากมาก แต่ถ้ามีการจัดหมวดหมู่ให้กับเวบเพจก่อนแล้ว จะทำให้การค้นและเข้าถึงข้อมูลที่ต้องการทำได้ง่ายขึ้น

วิทยานิพนธ์นี้ศึกษาวิธีการแยกหมวดหมู่ให้กับเวบเพจภาษาไทยแบบอัตโนมัติเพื่อนำไปใช้ร่วมกับการค้นหาข้อมูลเวบเพจภาษาไทย โดยจะแบ่งขอบเขตของการศึกษาออกเป็น 3 ส่วน คือ (1) ศึกษาถึงความสำคัญของคำในแท็กเอชทีเอ็มแอลที่มีต่อความการแยกหมวดหมู่ให้เอกสาร (2) การลดจำนวนของคำเพื่อเพิ่มประสิทธิภาพในการแยกหมวดหมู่ให้เอกสาร และ (3) วิธีการแยกหมวดหมู่

ผลการทดลองแสดงให้เห็นว่า (1) ถ้าเพิ่มความสำคัญให้กับคำที่อยู่ในแท็กเอชทีเอ็มแอลให้มากกว่าคำในเอกสาร การแยกหมวดหมู่ให้เวบเพจภาษาไทย จะมีความแม่นยำมากขึ้น (2) การลดจำนวนคำจะเพิ่มความถูกต้องเล็กน้อย และช่วยลดเวลาในการประมวลผล (3) เอสวีเอ็ม (SVM – Support Vector Machines) มีประสิทธิภาพดีกว่าตัวแยกแยะเบย์อย่างง่าย

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่ออาจารย์ที่ปรึกษา .....

ปีการศึกษา 2545 .....

4270386021 : MAJOR COMPUTER SCIENCE

KEY WORD: CATEGORIZATION / THAI LANGUAGE / AUTOMATIC

ADUL TANTHUVANIT : AUTOMATIC THAI WEB PAGE CATEGORIZATION.

THESIS ADVISOR : BOONSERM KIJSIRIKUL, Ph.D, 49 pp.

ISBN 974-17-1228-6.

Nowadays the number of documents or Web pages in the Internet is increasing rapidly, and this makes searching of required documents is very difficult. If the Web pages are organized into categories, the user can more easily search and access the Web pages.

This thesis studies a method of automatic Thai Web page categorization for applying to Thai search engines. The study is divided into three parts, i.e. (1) the study of significance of data in HTML tags in document categorization, (2) the method of reducing the number of words for efficient document categorization, and (3) the method of document categorization.

The experimental results show that (1) if words in HTML tags are given higher significance than the other words in the documents, the categorization of Thai Web pages will be more accurate, (2) the reduction of the number of words gives slightly more accuracy and speeds up the processing time, and (3) an SVM performs better than Naïve Bayes.

Department Computer Engineering Student's signature .....

Fields of study Computer Science Advisor's signature .....

Academic year 2002 .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างสูงจาก ผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ได้ให้ความรู้ คำแนะนำ และข้อคิดเห็นในการทำวิจัย ข้าพเจ้ารู้สึกซาบซึ้งในการสนับสนุนเป็นอย่างมาก และขอขอบคุณคณะกรรมการวิทยานิพนธ์ รองศาสตราจารย์ ดร.วันชัย ริ้วไพบูลย์ อาจารย์ ดร.ยรรยง เต็งอำนวย และอาจารย์นครทิพย์ พร้อมพูล ที่กรุณาให้คำแนะนำ ตรวจและแก้ไขวิทยานิพนธ์ฉบับนี้

ขอขอบคุณ อาจารย์นวลวรรณ สุนทรภิษัช ที่ให้คำแนะนำ และชี้แนวทางในการทำการวิจัย และการศึกษาค่าต่างๆ ขอบคุณเพื่อนสมาชิกห้องปฏิบัติการอัจฉริยภาพเครื่องกลและการค้นพบความรู้ (MIND LAB) รวมทั้งเพื่อนจากภาควิชาวิศวกรรมคอมพิวเตอร์ทุกคนที่นอกจากจะให้คำแนะนำดีๆแล้ว ยังให้กำลังในอยู่เรื่อยมา ขอขอบคุณเพื่อนร่วมงานของข้าพเจ้าที่คอยช่วยนำงานของข้าพเจ้าไปทำแทนเพื่อให้ข้าพเจ้ามีเวลาในการทำงานวิจัยมากขึ้น

สุดท้ายนี้ ข้าพเจ้าขอกราบขอบพระคุณบิดา มารดา และญาติพี่น้องที่คอยสนับสนุน และให้ความช่วยเหลือ จนสามารถผลิตผลงานวิจัยที่ข้าพเจ้าเชื่อมั่นว่าจะเป็นประโยชน์ต่อสังคม

สารบัญ

หน้า

กิตติกรรมประกาศ ..... **ฉ**

สารบัญ ..... **ช**

สารบัญตาราง ..... **ฌ**

สารบัญภาพ ..... **ญ**

บทที่ 1 บทนำ **1**

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 2

1.3 ขอบเขตของการวิจัย 2

1.4 ขั้นตอนการดำเนินการ 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

1.6 ลำดับขั้นตอนในการเสนอผลวิทยานิพนธ์ 3

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง **4**

2.1 ทฤษฎีที่เกี่ยวข้อง 4

2.1.1 ตัวแยกแยะเบย์อย่างง่าย (Naïve Bayes Classifier) 4

2.1.2 เอชทีเอ็มเอล (HTML - Hyper Text Markup Language) 6

2.1.3 การลดมิติ (Dimensionality Reduction) 8

2.1.4 ซัพพอร์ตเวกเตอร์แมชชีน (Supprt Vector Machines - SVM) 11

2.2 งานวิจัยที่เกี่ยวข้อง 13

2.2.1 การตัดคำภาษาไทย (CTTEX) 13

2.2.2 การศึกษาเปรียบเทียบวิธีในการแยกหมวดหมู่ให้กับคำภาษาจีน 15

บทที่ 3 การเตรียมข้อมูล **17**

3.1 การเก็บข้อมูล 17

3.2 การแยกส่วนประกอบของเวบเพจ 19

3.3 การตัดคำภาษาไทย 21

3.4 การเตรียมรายการคำทั่วไป (Stopword) 23

3.4.1 คำทั่วไปภาษาไทย 23
3.4.2 คำทั่วไปภาษาอังกฤษ 23
บทที่ 4 การทดลองและผลการทดลอง 24
4.1 วิธีการทดลอง 24
4.1.1 การทดลองเพื่อหาความสำคัญของการเตรียมข้อมูลแบบต่างๆ 26
4.1.2 การทดลองผลจากการลดมิติของคำ 27
4.1.3 การทดลองประสิทธิภาพของวิธีการแยกหมวดหมู่ 27
4.1.4 การทดลองเพื่อหาความสำคัญของข้อมูลในแท็กเอชทีเอ็มแอล 27
4.1.5 การทดลองว่า Information Gain ช่วยลดเวลาในการใช้เอสวีเอ็ม 28
4.2 ผลการทดลอง 28
4.2.1 การทดลองเพื่อหาความสำคัญของการเตรียมข้อมูลแบบต่างๆ 28
4.2.2 การทดลองผลจากการลดมิติของคำ 29
4.2.4 การทดลองเพื่อหาความสำคัญของข้อมูลในแท็กเอชทีเอ็มแอล 31
4.2.7 เวลาที่ใช้ในการทำงานของเอสวีเอ็มเมื่อมีการลดมิติแล้ว 32
4.3 สรุปผลการทดลอง 33
บทที่ 5 สรุปผลการวิจัย และข้อเสนอแนะ 35
5.1 สรุปผลการวิจัย 35
5.2 ข้อเสนอแนะ 36
รายการอ้างอิง 37
ประวัติผู้เขียนวิทยานิพนธ์ 39

สารบัญตาราง

หน้า

ตารางที่ 1 หมวดหมู่ย่อยในข้อมูลแต่ละชุด 17
ตารางที่ 2 ลักษณะของชุดข้อมูลวิทยาศาสตร์และเทคโนโลยี (ภาษาไทย) 24
ตารางที่ 3 ลักษณะของชุดข้อมูลกีฬาและนันทนาการ (ภาษาไทย) 25
ตารางที่ 4 ลักษณะของชุดข้อมูลงานอดิเรก (ภาษาอังกฤษ) 25
ตารางที่ 5 เปรียบเทียบความถูกต้องของการเตรียมข้อมูลแบบต่างๆ 28
ตารางที่ 6 ความถูกต้องของตัวแยกแยะเบย์อย่างง่ายและเอสวีเอ็ม 31
ตารางที่ 7 ความถูกต้องเมื่อเพิ่มความถี่คำในแท็กเอชทีเอ็มแอล 31
ตารางที่ 8 ผลการจับเวลาการทำงานของเอสวีเอ็มเมื่อมีการลดมิติแล้ว 33

สารบัญภาพ

หน้า

รูปที่ 1 ตัวอย่างการแยกหมวดหมู่ด้วยตัวแยกแยะเบย์อย่างง่าย 5
รูปที่ 2 ตัวอย่างเวบเพจ 7
รูปที่ 3 ตัวอย่างคำที่อยู่ในแท็กเอชทีเอ็มแอล 7
รูปที่ 4 ตัวอย่างข้อมูลที่จะนำมาลดมิติ 8
รูปที่ 5 ระนาบหลายมิติที่ใช้แยกดีที่สุด จะให้ระยะห่างระหว่างกลุ่มทั้งสองกลุ่มเป็น 2/||w|| 11
รูปที่ 6 การใช้เอสวีเอ็มในการแยกหมวดหมู่ให้เอกสาร 12
รูปที่ 7 ลักษณะ Tree ของ Trie 13
รูปที่ 8 ลักษณะของ Double-Array Trie 15
รูปที่ 9 สรุปขั้นตอนการทำงานของโปรแกรมท่องเวบไซต์ 18
รูปที่ 10 ตัวอย่างโค้ดเอชทีเอ็มแอลของเวบเพจ 19
รูปที่ 11 เวบเพจเมื่อนำไปแสดงผลในเวบบราวเซอร์ 20
รูปที่ 12 กราฟความถูกต้องเมื่อมีการลดมิติในการเตรียมข้อมูลแบบ HTML+2 29
รูปที่ 13 กราฟความถูกต้องเมื่อมีการลดมิติในการเตรียมข้อมูลแบบ No Stopword 30

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันนี้ โครงข่ายใยแมงมุม หรือที่เรียกกันทั่วไปว่า เวิลด์ไวด์เวบ (World Wide Web) ได้มีอัตราการเติบโตที่สูงขึ้นมาก ซึ่งภายในเวิลด์ไวด์เวบนี้มีข้อมูลอยู่จำนวนมหาศาล ดังนั้นสิ่งที่สำคัญมากสำหรับเวิลด์ไวด์เวบอย่างหนึ่งคือ กระบวนการค้นหาข้อมูลที่มีประสิทธิภาพ เดิมทีเครื่องมือค้นหาข้อมูลของเวิลด์ไวด์เวบนั้นอยู่ในรูปแบบของการค้นหาตามคีย์เวิร์ด โดยที่ไม่มีการจัดข้อมูลเป็นหมวดหมู่ก่อน

หลังจากยาฮู (Yahoo!) [2] ได้เปิดให้บริการในช่วงต้นปี พ.ศ. 2538 ยาฮูได้นำเสนอวิธีการค้นหาข้อมูลแบบใหม่ เพื่อให้ข้อมูลตรงกับสิ่งที่ต้องการมากขึ้น คือการค้นหาตามหมวดหมู่ โดยในขณะนั้นยาฮู ใช้วิธีการให้คนมาใส่ข้อมูลเวบที่เพิ่มใหม่อยู่เรื่อยๆ เพื่อลดความผิดพลาดอันเนื่องมาจากการใส่หมวดหมู่ผิด วิธีการของยาฮู นี้ให้ความสะดวกกับผู้ที่ใช้บริการเป็นอย่างมาก เพราะว่าทำให้สามารถค้นหาผลลัพธ์ที่ต้องการได้รวดเร็วยิ่งขึ้น

นักวิจัยหลายท่านได้มีการวิจัยเกี่ยวกับการจัดหมวดหมู่ให้เอกสารแบบอัตโนมัติหลายประเภท เช่น หัวข้อข่าว เอกสารคนไข้ และ เวบเพจ ขณะที่อินฟอซีค (Infoseek) เองก็ได้มีการใช้นิวรอลเน็ตเวิร์กมาใช้ในการจัดเวบเพจนับล้านให้เป็นหมวดหมู่ โดยใช้เทคโนโลยีที่ชื่อว่า ซีซีอี (CCE - Content Classification Engine) [4]

ย้อนไปในช่วงปี พ.ศ. 2523 นั้น การแยกหมวดหมู่แบบอัตโนมัติทำได้โดยการสร้างระบบฐานความรู้หรือระบบผู้เชี่ยวชาญที่มีความสามารถในการแบ่งแยกหมวดหมู่ได้ ซึ่งจำเป็นจะต้องใช้ผู้เชี่ยวชาญมาสร้างกฎต่างๆ ไว้ก่อน เมื่อมีการเปลี่ยนแปลงใดๆ ในโครงสร้างของหมวดหมู่ ก็จะต้องให้ผู้เชี่ยวชาญมาสร้างกฎใหม่อีกรอบ

หลังจากนั้นประมาณ 10 ปีได้มีวิธีการใหม่ในการแบ่งแยกเอกสารแบบอัตโนมัติ ซึ่งเป็นวิธีการที่เป็นที่นิยมมาจนถึงปัจจุบัน จากเดิมที่จะต้องมาสร้างกฎในการแยกหมวดหมู่ของเอกสาร วิธีการนี้จะใช้วิธีการเรียนรู้ลักษณะเฉพาะของเอกสารตัวอย่างที่ได้มีการจัดหมวดหมู่ให้ไว้ก่อนแล้ว เพื่อที่จะหาว่าเอกสารที่ใช้ทดสอบมีความใกล้เคียงกับเอกสารตัวอย่างในหมวดหมู่ใด เอกสารทดสอบก็ควรที่จะอยู่ในหมวดหมู่นั้นด้วยเช่นกัน

ในประเทศไทยการเติบโตของเวิลด์ไวด์เวบนั้นสูงมาก จึงมีความต้องการเครื่องมือในการค้นหาที่มีประสิทธิภาพสูง การแยกหมวดหมู่ของเวบเพจแบบอัตโนมัติจึงเป็นวิธีการที่จะช่วยให้ผู้ที่เข้ามาหาข้อมูลสามารถพบข้อมูลที่ต้องการได้อย่างรวดเร็ว หากแต่เพียงว่าข้อจำกัดของ

ภาษาไทยที่ไม่มีการแบ่งคำอย่างชัดเจนดังเช่นภาษาอังกฤษ ดังนั้นจึงต้องมีการทดลองหาวิธีการแยกเวบเพจแบบอัตโนมัติ ที่เหมาะสมกับภาษาไทยด้วย

โดยปัญหาที่งานวิจัยนี้สนใจคือ การแยกหมวดหมู่ให้กับเวบเพจ มีการจัดการกับเอกสารที่ทำให้มีความถูกต้องมากกว่าการแยกหมวดหมู่ให้เอกสารธรรมดาหรือไม่ การที่เวบเพจมีสองภาษาทำให้จำนวนคำมีมากขึ้นจะทำให้ความถูกต้องต่ำกว่าเวบเพจที่มีแต่ภาษาอังกฤษอย่างเดียวหรือไม่ ถ้าความถูกต้องต่ำกว่าจะมีวิธีการใดๆที่จะทำให้จำนวนคำลดลงและทำให้ความถูกต้องสูงขึ้นได้หรือไม่ และวิธีการแยกหมวดหมู่ที่เหมาะสมกับเวบเพจภาษาไทยคือวิธีการใด

1.2 วัตถุประสงค์ของการวิจัย

เพื่อพัฒนาวิธีการและขั้นตอน ในการคัดแยกเวบเพจภาษาไทยออกเป็นหมวดหมู่ได้อย่างอัตโนมัติ

1.3 ขอบเขตของการวิจัย

1. ใช้ได้กับเวบเพจภาษาไทยที่มีการเข้ารหัสคำตามมาตรฐาน ISO-620 [6]
2. มีการใช้มอดูลการตัดคำที่ได้มีการพัฒนาไว้ก่อนแล้ว
3. เวบเพจที่ใช้ในการวิจัยเป็นเวบเพจภาษาไทยทั้งหมด
4. เวบเพจที่ใช้ในการเปรียบเทียบผลจะมีเวบเพจภาษาอังกฤษด้วย

1.4 ขั้นตอนการดำเนินการ

1. ศึกษาทฤษฎีเกี่ยวกับวิธีการแยกหมวดหมู่แบบต่างๆ
2. ศึกษาถึงรูปแบบและวิธีการตัดคำแบบต่างๆ
3. ศึกษาโครงสร้างของภาษาเอชทีเอ็มเอล
4. เก็บข้อมูลเวบตัวอย่าง
5. สร้างข้อมูลเวบตัวอย่างที่ได้รับการแยกเป็นหมวดหมู่แล้ว สำหรับใช้ในการเรียนรู้
6. นำข้อมูลมาทดสอบกับโปรแกรม เก็บผลการทดลอง
7. วิเคราะห์ผลการทดลองที่ได้
8. สรุปผลและเรียบเรียงวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เพิ่มความสามารถของระบบค้นหา (Search Engine) ภาษาไทยให้สามารถค้นหาข้อมูลได้ตรงตามความต้องการของผู้ค้นมากขึ้น
2. สร้างไดเร็กทอรี่เวบ (Web Directory) ได้อย่างสะดวกและรวดเร็ว

3. นำงานวิจัยไปใช้กับภาษาอื่นที่มีลักษณะเดียวกับภาษาไทยได้
4. เพิ่มประสิทธิภาพในการตรวจสอบเวบไซต์ที่อยู่ในไดเร็กทอรี่เวบ ที่มีการเปลี่ยนเนื้อหาให้ไม่ตรงกับหมวดหมู่ที่อยู่เดิม

1.6 ลำดับขั้นตอนในการเสนอผลวิทยานิพนธ์

เนื้อหาของวิทยานิพนธ์ฉบับนี้ถูกแบ่งออกเป็น 5 บทดังนี้ บทที่ 1 เป็นบทนำ บทถัดไปบทที่ 2 จะกล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้อง เช่น การตัดคำภาษาไทย การลดมิติ เอสวีเอ็ม และตัวแยกแยะเบย์อย่างง่าย บทที่ 3 จะกล่าวถึง การเตรียมข้อมูลเพื่อใช้ในการทดลอง บทที่ 4 จะเป็นผลการทดลอง และ บทที่ 5 จะเป็นสรุปผลการวิจัยและข้อเสนอแนะ

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ตัวแยกแยะเบย์อย่างง่าย (Naïve Bayes Classifier)

ในแง่ของสถิติแล้ว ถ้ามีเอกสาร d_j ต้องการหาความน่าจะเป็นที่จะพบเอกสาร d_j ในหมวดหมู่ c_i จะสามารถหาได้จาก $P(c_i|d_j)$ ตามสมการด้านล่าง

$$P(c_i \mid d_j) = \frac{P(c_i)P(d_j \mid c_i)}{P(d_j)} \quad (1)$$

สมการนี้เป็นทฤษฎีของเบย์ (Bayes) โดยค่า $P(c_i)$ คือความน่าจะเป็นที่จะพบหมวดหมู่ c_i $P(d_j)$ คือความน่าจะเป็นที่จะพบเอกสาร d_j และ $P(d_j|c_i)$ คือความน่าจะเป็นที่จะพบหมวดหมู่ c_i เมื่อพบเอกสาร d_j ในความเป็นจริงการจะหาค่า $P(d_j)$ และค่า $P(d_j|c_i)$ จะต้องใช้เอกสารจำนวนมาก เพื่อเป็นตัวแทนของเอกสารทั้งหมด

เพื่อให้สามารถหาค่าของ $P(d_j|c_i)$ และสามารถละค่าของ $P(d_j)$ ได้ จึงต้องมีการตั้งสมมติฐานขึ้นมา 2 ข้อ [15] คือ (1) เอกสารทุกเอกสารเป็นตัวแปรแบบสุ่ม (Random Variables) และ เอกสารทุกเอกสารเป็นอิสระต่อกัน (Statistically Independent) (2) ตำแหน่งของคำในเอกสารไม่มีผลต่อความสำคัญของคำ กล่าวคือ ไม่ว่าคำนั้นจะอยู่ที่ต้นเอกสาร หรือท้ายเอกสารก็มีความสำคัญเท่ากัน

จากสมมติฐานข้อแรกทำให้ค่า $P(d_j)$ ของแต่ละเอกสารสามารถละได้เนื่องจากเมื่อเอกสารทุกเอกสารมีโอกาสพบเท่ากัน การที่จะมีค่า $P(d_j)$ หรือไม่มีในสมการของ $P(d_j|c_i)$ ก็ไม่มีผลต่อการเปรียบเทียบค่า $P(d_j|c_i)$

จากสมมติฐานข้อที่สองทำให้สามารถประมาณค่า $P(d_j|c_i)$ เป็นผลคูณของความน่าจะเป็นที่คำในเอกสารนั้นจะทำให้เกิดหมวดหมู่ c_i แทนได้ โดยให้ d_j เป็นเวกเตอร์ของคำมีค่าเป็น $(w_{1j},.....,w_{rj})$ เมื่อ r คือจำนวนคำทั้งหมดในเอกสารนั้น และเขียนค่าของ $P(d_j|c_i)$ ได้ใหม่เป็น

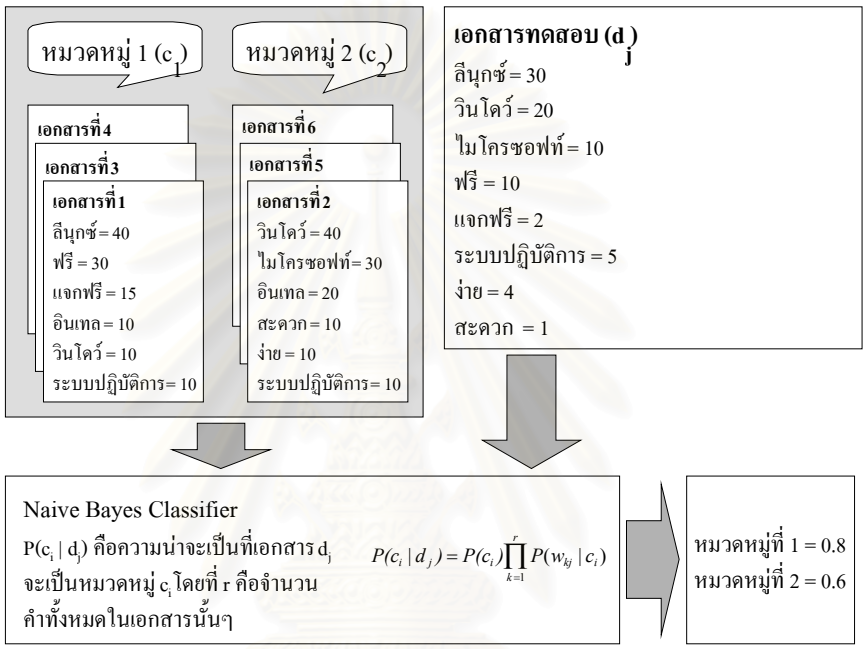
$$P(d_j \mid c_i) = \prod_{k=1}^{r} P(w_{kj} \mid c_i) \quad (2)$$

การแยกหมวดหมู่ให้เอกสารที่ใช้ข้อสมมติฐานนี้ จะเรียกว่า ตัวแยกแยะเบย์อย่างง่าย (Naïve Bayes Classifiers) สมการสำหรับหาค่าของ $P(c_i|d_j)$ จึงสามารถเขียนได้ใหม่เป็น

$$P(c_i \mid d_j) = P(c_i)\prod_{k=1}^{r} P(w_{kj} \mid c_i) \quad (3)$$

จากสมการดังกล่าว หากทราบค่า $P(w_{kj}|c_i)$ ของคำแต่ละคำในหมวดหมู่แต่ละหมวดหมู่ และทราบค่า $P(c_i)$ ของแต่ละหมวดหมู่ จะทำให้สามารถทราบได้ว่าเอกสารที่นำมาทดสอบมีความน่าจะเป็นที่จะอยู่ในหมวดหมู่ใดมากกว่า

ค่า $P(c_i)$ นี้สามารถคำนวณจากจำนวนเอกสารในหมวดหมู่นี้ต่อจำนวนเอกสารทั้งหมดได้ ส่วนค่าของ $P(w_{kj}|c_i)$ สามารถหาได้จาก จำนวนของคำ w_{kj} ปรากฏในเอกสารที่อยู่ในหมวดหมู่ c_i ต่อจำนวนคำ w_{kj} ทั้งหมด



รูปที่ 1 ตัวอย่างการแยกหมวดหมู่ด้วยตัวแยกแยะเบย์อย่างง่าย

จากรูปที่ 1 กรอบสีเทาคือเอกสารตัวอย่าง มี 2 หมวดหมู่ เมื่อนำเอกสารตัวอย่างเข้ามาเรียนรู้ด้วยตัวแยกแยะเบย์อย่างง่าย ตัวแยกแยะเบย์อย่างง่ายจะอ่านคำทุกคำในเอกสารขึ้นมา แล้วหาค่า $P(w_{kj}|c_i)$ ของแต่ละคำในแต่ละหมวดหมู่ และค่า $P(c_i)$ เก็บเอาไว้ ในฐานความรู้ (Knowledge Base)

หลังจากผ่านกระบวนการเรียนรู้แล้ว เมื่อนำเอกสารทดสอบมาหาหมวดหมู่ด้วยตัวแยกแยะเบย์อย่างง่าย ก็เพียงแต่นำค่า $P(w_{kj}|c_i)$ และค่า $P(c_i)$ ที่เก็บเอาในฐานความรู้ มาแทนค่าให้สอดคล้องกับคำที่อยู่ในเอกสารทดสอบ จนได้ค่า $P(c_i|d_j)$ ของทุกหมวดหมู่ แล้วจึงเลือกหมวดหมู่ที่มีค่า $P(c_i|d_j)$ สูงที่สุดเป็นหมวดหมู่ที่ถูกต้อง

2.1.2 เอชทีเอ็มเอล (HTML - Hyper Text Markup Language) [19,13]

เอชทีเอ็มเอล เป็นภาษาที่ยืมโครงสร้างของเอสจีเอ็มแอลมาพัฒนาต่อเพื่อควบคุมการแสดงผลของเอกสารบนหน้าจอและกำหนดโครงสร้างอย่างง่ายให้กับเอกสาร ลักษณะของเอกสารเอชทีเอ็มเอลนั้น เป็นการเพิ่มแท็ก (Tag) ซึ่งมีลักษณะ “<tagname>.....</tagname>” หรือ “<tagname attr=value>” เข้าไปในเอกสาร เพื่อเป็นการอธิบายหรือจัดรูปแบบการแสดงผลของคำหรือประโยคที่อยู่ระหว่างแท็กนั้นได้ เช่น “เวบเพจ” นั้นหมายถึง คำว่า “เวบเพจ” จะเป็นตัวหนา

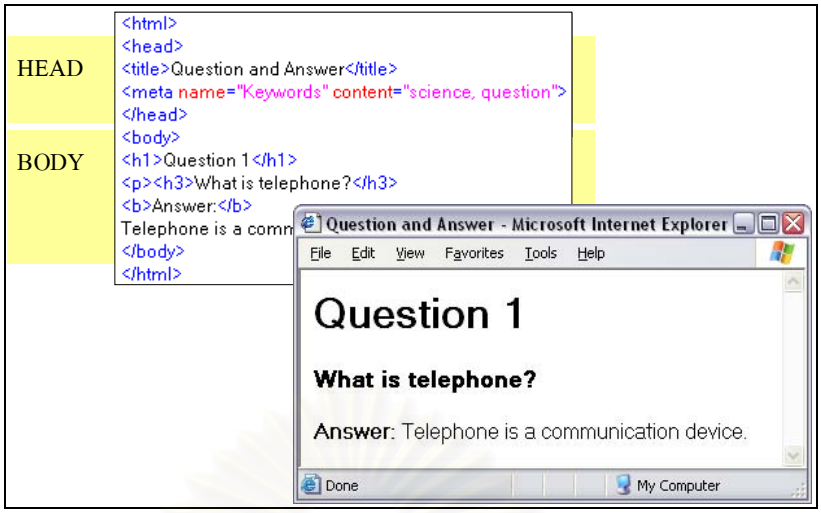
ภาษาเอชทีเอ็มเอลมีโครงสร้างที่ง่ายไม่ซับซ้อน สามารถสร้างจากโปรแกรมแก้ไขเอกสารธรรมาดา หรือจะเป็นโปรแกรมแก้ไขเอกสารเอชทีเอ็มแอลโดยเฉพาะก็ได้ (เช่น Macromedia Dreamweaver) ปัจจุบันนี้เอชทีเอ็มเอลมีถึงรุ่นที่ 4 แล้ว โดยอยู่ในการควบคุมของ W3C เพื่อให้เอชทีเอ็มแอลมีมาตรฐานเดียวกันทั่วโลก สำหรับเอกสารภาษาเอชทีเอ็มเอลนั้น จำเป็นจะต้องเปิดดูภายใต้โปรแกรมที่เรียกว่า เวบบราวเซอร์ (Web Browser)

โครงสร้างของเอกสารเอชทีเอ็มเอลนั้นจะมี 2 ส่วนใหญ่ๆ คือส่วนที่เป็นเฮด (Head) และ ส่วนที่เป็นบอดี้ (Body) โดยส่วนที่เป็นเฮดจะเก็บข้อมูลทั่วไปของเอกสารนั้นๆ เช่น ชื่อของเอกสาร (Title) การจัดรูปแบบเอกสาร (Style Sheet) ภาษาสคริปต์สำหรับทำงานในเอกสาร (Script) และ ข้อมูลจำเพาะเกี่ยวกับเอกสารนั้นๆ (Meta Tag) สำหรับในส่วนของบอดี้ จะเป็นเนื้อหา ซึ่งจะแสดงบนพื้นที่แสดงผลของ เวบบราวเซอร์ทั้งหมด โดยแท็กที่คนนิยมใช้กันเช่น ทำตัวหนา (B) ใส่รูป (Img) ทำตาราง (Table) ขึ้นบรรทัดใหม่ (Br) การทำตัวเชื่อมโยง (A) และ ทำย่อหน้า (P, Dd, Ol, Ul, Li) เป็นต้น

ตัวอย่างเช่น ถ้ามีเอกสารเขียนว่า

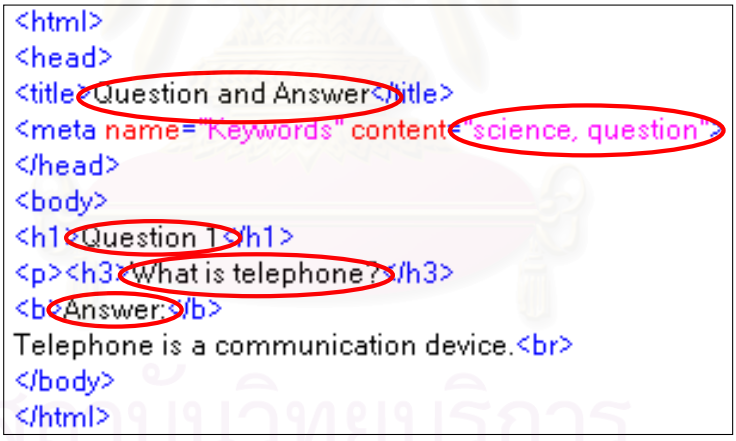
```
Question 1
What is Telephone?
Answer: Telephone is a communication device.
```

ถ้าต้องการนำเอกสารดังกล่าวมาสร้างเป็นเวบเพจ ก็เพียงเพิ่มแท็กเอชทีเอ็มแอลเข้าไป แล้วนำไปแสดงผลในเวบบราวเซอร์ ดังรูปที่ 2



รูปที่ 2 ตัวอย่างเวบเพจ

พิจารณาจากรูปที่ 2 จะเห็นว่าในโค้ดเอชทีเอ็มแอลแบ่งออกเป็น 2 ส่วนคือ เฮด และบอดี้ โดยข้อมูลที่แสดงผลบนเวบบราวเซอร์จะเป็นข้อมูลในส่วนบอดี้ จากโค้ดนี้จะแสดงให้เห็นว่ามีข้อมูลบางส่วนที่เมื่อเพิ่มแท็กเอชทีเอ็มแอลเข้าไปแล้ว นอกจากจะควบคุมการแสดงผลแล้ว ข้อมูลดังกล่าวยังมีความสำคัญมากขึ้นอีกด้วย ข้อมูลดังกล่าวแสดงในรูปที่ 3



รูปที่ 3 ตัวอย่างคำที่อยู่ในแท็กเอชทีเอ็มแอล

ในงานวิจัยนี้จึงมีการดึงข้อมูลจากแท็กเอชทีเอ็มแอลบางแท็กออกมาเพื่อที่จะใช้เพิ่มความถูกต้องในการแยกหมวดหมู่ให้เวบเพจ สำหรับแท็กที่จะดึงข้อมูลออกมามีดังต่อไปนี้

META แท็กนี้จะมีรูปแบบเป็นแท็กที่ไม่ต้องการตัวปิด (ไม่ต้องใช้ </META>) โดยข้อมูลที่สำคัญจะอยู่ในแท็กนี้ โครงสร้างของแท็กนี้คือ <META NAME=A CONTENT=B> โดย A จะเป็นชื่อของข้อมูลเช่น KEYWORD, DESCRIPTION, ROBOTS เป็นต้น ส่วน CONTENT จะเป็นข้อมูลที่สอดคล้องกับชื่อ

TITLE ชื่อของเวบเพจ โดยจะแสดงที่หัวของหน้าต่างเวบบราวเซอร์ และแสดงในชื่อรายการโปรด (Favourites) แท็กนี้จะอยู่ในรูปแบบ <TITLE>ชื่อเวบเพจ</TITLE>

B, I ทำให้ข้อมูลที่อยู่ในแท็กดังกล่าวนี้เป็นตัวหนา (B) และเป็นตัวเอียง (I) วิธีการใช้คือ <B>ตัวหนา</B>, <I>ตัวเอียง</I>

H1, H2, H3, H4, H5, H6 จะเป็นการสร้างหัวข้อขนาดต่างๆ H1 จะให้หัวข้อตัวอักษรขนาดใหญ่ที่สุด และ H6 จะตัวเล็กที่สุด วิธีการใช้งานคือ <H1>หัวข้อหลัก</H1>

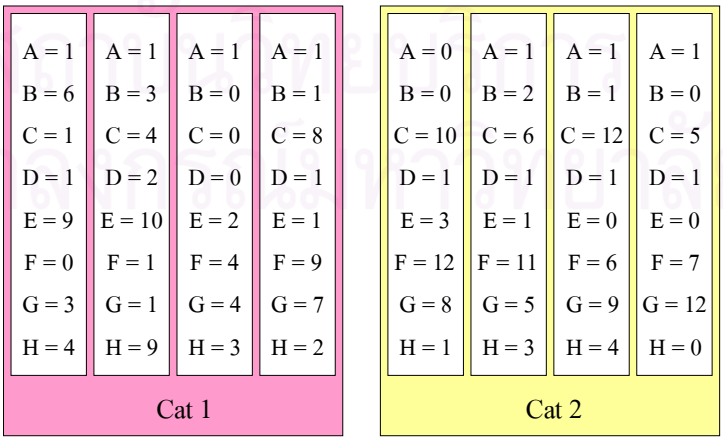
IMG จะเป็นการแทรกรูปภาพเข้าไปในเอกสาร โดยในแท็กนี้จะมีข้อมูลอยู่ที่ชื่อภาพอยู่ด้วย รูปแบบการใช้งานของ IMG คือ <IMG SRC=รูปภาพ ALT=ชื่อภาพ> แท็กนี้ไม่จำเป็นต้องปิดแท็ก

A ย่อมาจาก Anchor ซึ่งจะเป็นการเชื่อมโยงไปยังเวบเพจอื่นๆ หรือส่วนอื่นในเวบเพจ ข้อมูลที่อยู่ระหว่าง <A>…</A> จะเป็นข้อมูล หรือรูป ที่อธิบายถึงเวบเพจที่เชื่อมโยงไปถึง รูปแบบการใช้งานคือ <A HREF=ชื่อเวบเพจ>ข้อความอธิบายเวบเพจที่เชื่อมโยงไป</A>

ข้อมูลที่อยู่ในแท็กข้างต้นล้วนแล้วแต่มีความสำคัญมากกว่าข้อความปกติที่อยู่บนเวบเพจ อีกทั้งข้อความที่อยู่ในแท็กบางแท็กยังไม่แสดงผลอีกด้วย

2.1.3 การลดมิติ (Dimensionality Reduction) [20]

การลดมิติเป็นการลดจำนวนของคำทั้งหมด ด้วยจุดมุ่งหมายที่จะเก็บเฉพาะคำที่เป็นประโยชน์ต่อการแยกหมวดหมู่ในเวบเพจมากที่สุดเอาไว้เท่านั้น ซึ่งจะช่วยเพิ่มความเร็วและเพิ่มความถูกต้องให้กับกระบวนการแยกหมวดหมู่ นอกจากนี้ยังนำไปใช้เพื่อลดจำนวนมิติให้กับวิธีการแยกหมวดหมู่ที่ไม่สามารถรองรับมิติมากเกินไปได้อีกด้วย



รูปที่ 4 ตัวอย่างข้อมูลที่จะนำมาลดมิติ

จากรูปที่ 4 กรอบใหญ่คือหมวดหมู่ มีสองหมวดหมู่คือ Cat 1 และ Cat 2 ในแต่ละหมวดหมู่จะมีเอกสาร 4 เอกสาร และมีคำอยู่ทั้งหมด 8 คำ เมื่อพิจารณาเป็นคำแล้ว คำที่ให้โอกาสที่เอกสารจะเป็นหมวดหมู่ Cat 1 มากคือ B, E และ H ขณะที่คำที่ให้โอกาสเอกสารจะเป็นหมวดหมู่ Cat 2 มากคือ C, F และ G ส่วนคำ A และ D จะปรากฏในหมวดหมู่ Cat 1 และ Cat 2 ในความน่าจะเป็นที่ใกล้เคียงกัน จุดมุ่งหมายของการลดมิติของคำจึงหาวิธีการคำนวณค่าความสำคัญของคำ เพื่อที่จะสามารถตัดคำที่เป็นคำทั่วไป ไม่มีความสำคัญต่อการแยกหมวดหมู่อย่าง A และ D ออกไปได้

วิธีการคำนวณหาค่าความสำคัญของคำที่นิยมกันมีดังต่อไปนี้

Document Frequency Thresholding (DF) วิธีการนี้จะเป็นการนับจำนวนของคำที่พบในเอกสาร โดยคำที่มีจำนวนความถี่น้อยกว่าค่าขีดแบ่ง (Threshold) จะไม่ถูกนำมาใช้ โดยจะมีสมมติฐานที่ว่า คำที่พบน้อย จะไม่มีข้อมูลที่จำเป็นสำหรับการจัดหมวดหมู่ ดังนั้นการที่จะลดคำที่พบน้อยย่อมเป็นการลดมิติและเพิ่มความถูกต้องในการจัดหมวดหมู่ของเอกสาร วิธีการนี้จะเป็นวิธีการที่ง่ายที่สุดในการลดจำนวนคำ นอกจากนี้ยังสามารถนำไปใช้ได้ง่ายกับข้อมูลขนาดใหญ่ โดยเวลาจะเพิ่มขึ้นแบบเชิงเส้นตามจำนวนเอกสารที่นำมาทดสอบ อย่างไรก็ดีวิธีการนี้ยังมีข้อเสียอยู่ตรงที่คำบางคำที่มีความถี่มากอาจจะเป็นคำทั่วไป และไม่ได้ช่วยในการแยกหมวดหมู่เลยก็ได้ เช่น มีข้อมูลชุดกีฬา ประกอบด้วยหมวดหมู่ฟุตบอล บาสเกตบอล และวอลเลย์บอล คำว่ากีฬา มีความถี่สูง พบได้ในเอกสารเกือบทุกเอกสาร ถ้าใช้วิธีการ DF คำว่ากีฬาจะไม่ถูกตัดทิ้ง ทั้งที่ความจริงแล้วคำว่ากีฬา เป็นคำที่ไม่ได้ช่วยให้การแยกหมวดหมู่ดีขึ้น

Information Gain (IG) เป็นวิธีการที่จะวัดค่าความดีของคำ นิยมใช้ในการสร้างต้นไม้ตัดสินใจ (Decision Tree) โดยจะคำนวณค่าจำนวนบิตของสารสนเทศ (Bit of Information) ที่คำนั้นมีต่อการแยกหมวดหมู่ เมื่อใช้คำดังกล่าว และเมื่อคำดังกล่าวหายไป หากคำดังกล่าวจะอยู่หรือหายไปก็ไม่มีผลต่อการแยกหมวดหมู่ คำดังกล่าวจะมีจำนวนบิตของข้อมูลต่ำ ขณะที่ถ้าคำดังกล่าวอยู่หรือหายไปมีผลต่อการแยกหมวดหมู่มาก คำดังกล่าวจะให้ค่าบิตของสารสนเทศสูง

Information Gain จะใช้ทฤษฎีสารสนเทศ (Information Theory) ซึ่งกล่าวไว้ว่า “สารสนเทศที่ได้รับจากสารจะขึ้นอยู่กับความน่าจะเป็นที่จะเกิดสารและสามารถวัดในรูปของบิตได้จากลบของลอการิทึมฐาน 2 ของความน่าจะเป็นนั้น” โดยจะหาค่าสารสนเทศทั้งหมดก่อน แล้วจึงมาลบออกด้วยสารสนเทศที่ได้จากการที่ในหมวดหมู่ c จะมีคำ t และสารสนเทศที่ได้จากการที่ในหมวดหมู่ c จะไม่มีคำ t สมการของ Information Gain ของคำ t จึงเขียนได้ดังนี้

$$
\begin{aligned}
IG(t) = & -\sum_{i=1}^{m} P(c_i)\log P(c_i) \\
& + P(t)\sum_{i=1}^{m} P(c_i \mid t)\log P(c_i \mid t) \\
& + P(\bar{t})\sum_{i=1}^{m} P(c_i \mid \bar{t})\log P(c_i \mid \bar{t})
\end{aligned} \tag{4}
$$

โดยที่ $\bar{t}$ คือคำทุกคำที่ไม่ใช่ t ผลที่ได้จากสมการข้างต้นจึงเป็นจำนวนบิตของสารสนเทศที่เพิ่มขึ้นเมื่อมีการใส่คำ t เข้าไปในข้อมูล และเมื่อนำคำ t ออกไปจากกลุ่มของข้อมูล ถ้าคำ t เป็นคำที่ไม่มีความสำคัญจะทำให้ค่าสารสนเทศที่เพิ่มขึ้นน้อยมาก ขณะที่ถ้าคำดังกล่าวเป็นคำที่มีความสำคัญต่อการแยกหมวดหมู่มากจะทำให้สารสนเทศที่เพิ่มขึ้นมีค่ามาก

การนำ Information Gain เป็นใช้กับการลดมิติโดยการเลือกเฉพาะคำที่มีค่า Information Gain สูงไว้ และตัดคำที่มี Information Gain ต่ำทิ้งไป จะทำให้การลดมิติเหลือแต่คำที่มีประโยชน์ต่อการแยกหมวดหมู่เอาไว้

χ^2 statistic (CHI) การที่คำ t และเกิดขึ้นในหมวดหมู่ c เท่านั้น ในทางสถิติแล้วหมายความว่า t และ c ไม่เป็นอิสระต่อกัน ค่าความไม่เป็นอิสระต่อกันของ t และ c นั้นสามารถหาได้จากการกระจายแบบ χ^2 ที่มีระดับขั้นความเสรี (Degree of freedom) เป็น 1 โดยการใช้ ตารางความน่าจะเป็นสองทางของ t และ c เมื่อ A คือจำนวนที่ t และ c เกิดขึ้นพร้อมกัน B คือจำนวนครั้งที่ t ไม่มีใน c C คือจำนวนครั้งที่ c ไม่มีใน t D คือจำนวนครั้งที่ไม่เกิดทั้ง t และ c และ N คือจำนวนเอกสารทั้งหมด ค่า χ^2 ของ t และ c สามารถหาได้จาก

$$
\chi^2(t,c) = \frac{N \times (AD - CB)^2}{(A+C)\times(B+D)\times(A+B)\times(C+D)} \tag{5}
$$

โดยที่ χ^2 จะมีค่าเป็น 0 ถ้าหากว่า t และ c เป็นอิสระต่อกัน หรือ t สามารถเกิดขึ้นในหมวดหมู่ใดก็ได้ไม่เฉพาะเจาะจงกับหมวดหมู่ c เท่านั้น หลังจากได้คำนวณครบทุกหมวดหมู่ของทุกคำแล้ว จะสามารถประมาณค่า χ^2 ของคำ t ได้สองวิธีคือ

$$
\begin{aligned}
\chi^2_{avg}(t) &= \sum_{i=1}^{m} P(c_i)\chi^2(t,c_i) \\
\chi^2_{\max}(t) &= \max_{i=1}^{m}\{\chi^2(t,c_i)\}
\end{aligned} \tag{6}
$$

χ^2 จะสามารถนำมาใช้ในการลดมิติได้เพราะว่า คำที่เป็นอิสระต่อทุกหมวดหมู่ย่อมมีค่า χ^2 น้อย ขณะที่ถ้าคำไม่เป็นอิสระคือจะต้องเกิดที่หมวดหมู่ใด หมวดหมู่หนึ่งเท่านั้นจะทำให้ค่า χ^2 มีค่ามาก ในการลดมิติจึงเลือกเฉพาะคำที่มีค่า χ^2 สูงเก็บไว้เท่านั้นก็พอ

จากงานวิจัยของ Yang, Y., Pedersen, J. [20] ได้สรุปไว้ว่าค่า IG และค่า CHI ให้ผลในการลดมิติดีที่สุด แต่เนื่องจากค่า CHI นั้นเหมาะกับข้อมูลที่มีความถี่ของคำสูงเพราะว่า

ต้องประมาณค่าให้มีการแจกแจงปรกติ (Normal Distribution) ซึ่งในงานวิจัยนี้มีเวบเพจสำหรับทดลองเพียง 2,000 เวบเพจซึ่งน้อยเกินไป จึงใช้ค่า IG เข้ามาลดค่าเพียงวิธีการเดียว

2.1.4 ซัพพอร์ตเวกเตอร์แมชชีน (Supprt Vector Machines - SVM) [5]

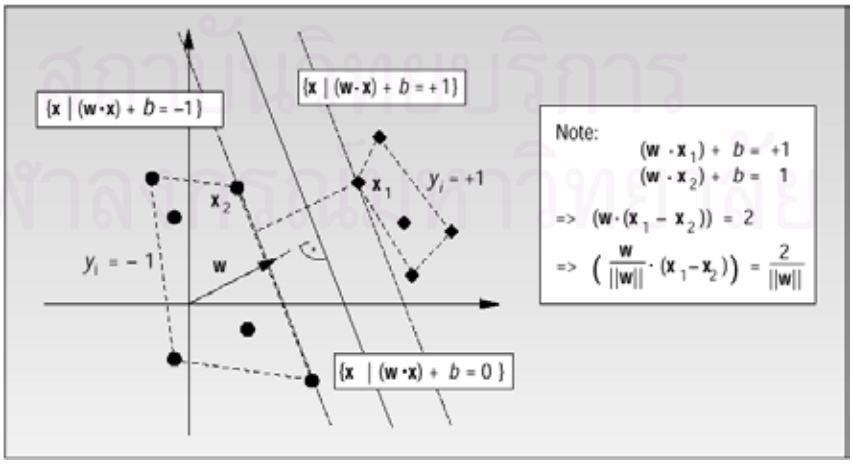
เอสวีเอ็มเป็นวิธีการเรียนรู้ทางสถิติที่เสนอโดย Vapnik [18] โดยอาศัยทฤษฎี การลดความเสี่ยงเชิงโครงสร้างให้ต่ำสุด (Structural Risk Minimization) เอสวีเอ็มจะหาระนาบที่จะแบ่งข้อมูล 2 ประเภทออกจากกัน โดยจะหาจากจุดซัพพอร์ตเวกเตอร์ ซึ่งจะเป็นการเลือกเฉพาะข้อมูลที่มีความสำคัญเท่านั้น

ให้ S เป็นจุดจำนวน N จุดที่แยกออกจากกันแบบเชิงเส้น (Linearly Separable) $S = \{x_i \in R^n | i=1,2,\ldots,N\}$ จุด x_i จะอยู่ในหมวดหมู่ใดหมวดหมู่หนึ่งจากทั้งหมด 2 หมวดหมู่เขียนอยู่ในรูป $y_i \in \{-1, +1\}$ เส้นที่เรียกว่าระนาบหลายมิติที่ใช้แยก (Separating Hyper-Plane) จะแบ่งจุด S ออกเป็น 2 ส่วน โดยแต่ละด้านของระนาบนี้ก็จะประกอบด้วยจุดที่อยู่ในหมวดหมู่เดียวกันเท่านั้น ระนาบหลายมิติดังกล่าวนี้สามารถมีได้หลายรูปแบบ ขอเพียงให้สามารถแบ่งจุดออกเป็นสองด้านได้เท่านั้น โดยสามารถเขียนระนาบดังกล่าวให้อยู่ในรูปแบบของคู่อันดับ (w, b) ที่สอดคล้องกับสมการสองสมการด้านล่าง

$$w \cdot x + b = 0 \text{, } w \in R^N, b \in R \tag{7}$$

$$\text{และ } \begin{cases} w \cdot x_i + b \geq +1 & \text{if } y_i = +1 \\ w \cdot x_i + b \leq -1 & \text{if } y_i = -1 \end{cases} \tag{8}$$

สำหรับค่า i = 1,2,...N



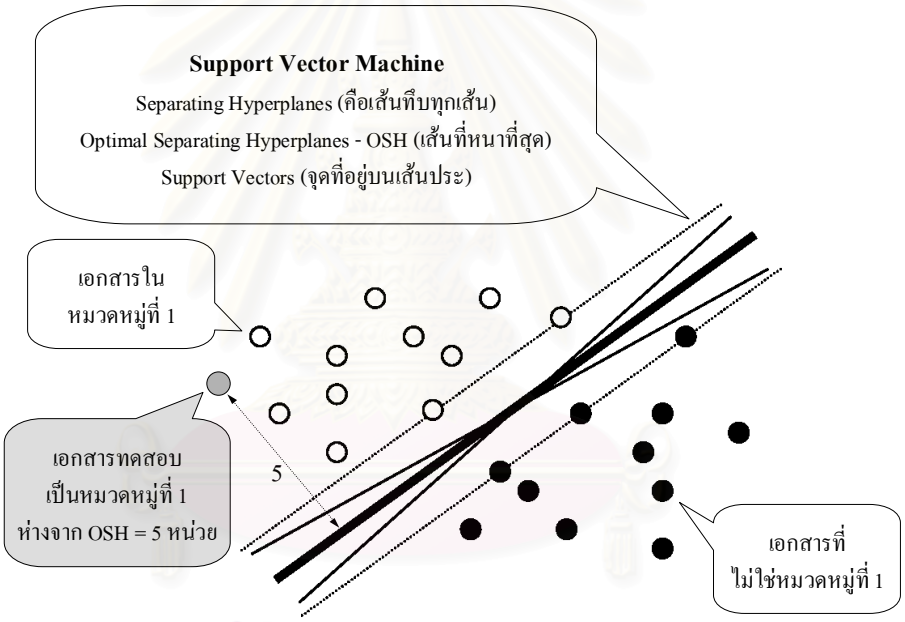
รูปที่ 5 ระนาบหลายมิติที่ใช้แยกดีที่สุด จะให้ระยะห่างระหว่างกลุ่มทั้งสองกลุ่มเป็น 2/||w||

เราพบว่าระยะห่างระหว่างกลุ่มทั้งสองคือ 2/||w|| ดังที่แสดงในรูปที่ 5 โดยระยะห่างดังกล่าวนี้จะมีค่ามากหรือน้อยขึ้นอยู่กับเวกเตอร์ w เป้าหมายของการเรียนรู้แบบเอสวีเอ็มคือการหาระนาบหลายมิติที่ใช้แยกดีที่สุด (Optimal Separating Hyper-Plane – OSH) ซึ่งจะห่างจากข้อมูลทั้งสองข้างมากที่สุด สามารถหาได้จาก

การหาค่าต่ำที่สุดของ 2/||w||

ในสมการ $\begin{cases} w \cdot x + b \geq +1 & \text{if } y_i = +1 \\ w \cdot x + b \leq -1 & \text{if } y_i = -1 \end{cases}$ $\text{for } i = 1,2,....,N$ (9)

จุดใน S ที่อยู่ใกล้ระนาบหลายมิติที่ใช้แยกดีที่สุดมากที่สุดจะเรียกว่า ซัพพอร์ตเวกเตอร์ จากรูปที่ 6 ระนาบหลายมิติที่ใช้แยกแสดงด้วยเส้นทึบทุกเส้น ระนาบหลายมิติที่ใช้แยกที่ดีที่สุดแสดงด้วยเส้นทึบหนา และซัพพอร์ตเวกเตอร์แสดงด้วยจุดที่อยู่บนเส้นประ



รูปที่ 6 การใช้เอสวีเอ็มในการแยกหมวดหมู่ให้เอกสาร

เอสวีเอ็มยังสามารถใช้ในการแก้ปัญหาที่ข้อมูลไม่มีการแบ่งกันอย่างชัดเจน (Non-Separable) และปัญหาที่แบ่งเป็นเชิงเส้นไม่ได้ (Non-Linear) ได้ด้วย โดยการใช้เคอร์เนล (Kernel) แมปข้อมูลเข้าไปสู่ปริภูมิอันดับสูงขึ้น ซึ่งจะทำให้ข้อมูลทั้งสองสามารถแบ่งออกเป็นเชิงเส้นได้

ในงานวิจัยนี้จะใช้เอสวีเอ็มในการแยกหมวดหมู่ให้กับเวบเพจด้วย โดยใช้คำเป็นมิติของอินพุตสเปซ (Input Space) แต่ละเอกสารจะเป็นแต่ละจุด เมื่อเอสวีเอ็มสามารถหาระนาบหลายมิติที่ใช้แยกดีที่สุดได้จึงนำเอกสารทดสอบมาทดสอบว่าอยู่บนระนาบ หรือใต้ระนาบ ซึ่งถ้าได้

ผลลัพท์เป็นบวกแสดงว่าเอกสารทดสอบอยู่ในหมวดหมู่เดียวกับตัวอย่างบวก แต่ถ้าได้ผลเป็นลบแสดงว่าเอกสารทดสอบไม่ได้อยู่ในหมวดหมู่เดียวกับตัวอย่างบวกดังแสดงในรูปที่ 6

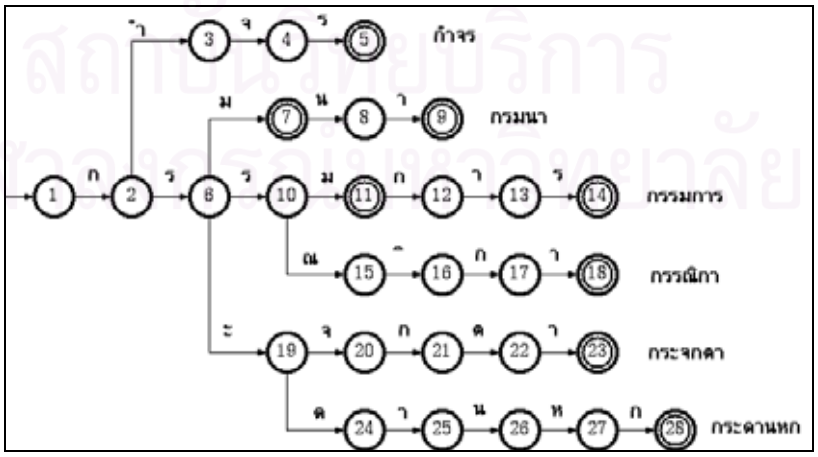
2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 การตัดคำภาษาไทย (CTTEX) [1,7]

โปรแกรม CTTEX นี้เป็นโปรแกรมที่ช่วยในการตัดคำภาษาไทย โดยเริ่มแจกจ่ายให้คนทั่วไปใช้ตั้งแต่ปี พ.ศ. 2537 ล่าสุดเป็นรุ่น 1.21 (พัฒนาวันที่ 19 พฤษภาคม พ.ศ. 2542) วัตถุประสงค์หลักของโปรแกรม CTTEX นั้นออกแบบมาเพื่อใช้ในการตัดคำของเอกสาร LaTeX ภาษาไทย นอกจากนี้ CTTEX ยังได้นำมาใช้ในการแทรก <wbr> (Word Break) ในเวบเพจ เพื่อให้เวบบราวเซอร์สามารถแสดงผลเวบเพจภาษาไทยได้อย่างถูกต้อง

ใน CTTEX ในรุ่น 1.21 มีการใช้รายการคำของราชบัณฑิตยสถานมาช่วยให้ระบบการตัดคำถูกต้องมากยิ่งขึ้น วิธีค้นหาคำจะใช้วิธี DFA (Trie) และกระบวนการแยกคำจะเป็นกระบวนการเรียกซ้ำ (Recursive) เพื่อหาวิธีการตัดคำที่เป็นไปได้ทั้งหมดออกมาก่อน แล้วเลือกวิธีตัดคำที่ให้จำนวนคำน้อยที่สุด

Trie เป็นวิธีการสร้างดรรชนีที่มีประสิทธิภาพสูงมาก โดยถือเป็นรูปแบบหนึ่งของ DFA (Deterministic Finite Automata) ลักษณะตามรูปที่ 7 วงกลมปกติคือแต่ละโนดซึ่งจะเก็บค่าสถานะของโนดนั้นๆ ขณะที่เส้นจะมีค่าเพื่อให้สามารถบอกเส้นทางไปยังโนดถัดไป ที่โนดหมายเลข 5, 7, 9, 11, 14, 18, 23 และ 28 จะเป็นสถานะสิ้นสุด ซึ่งหมายความว่าพบข้อมูลดังกล่าวอยู่ใน Trie แต่ถ้าไม่สามารถให้สถานะสิ้นสุดได้แสดงว่าข้อมูลดังกล่าวไม่อยู่ใน Trie



รูปที่ 7 ลักษณะ Tree ของ Trie

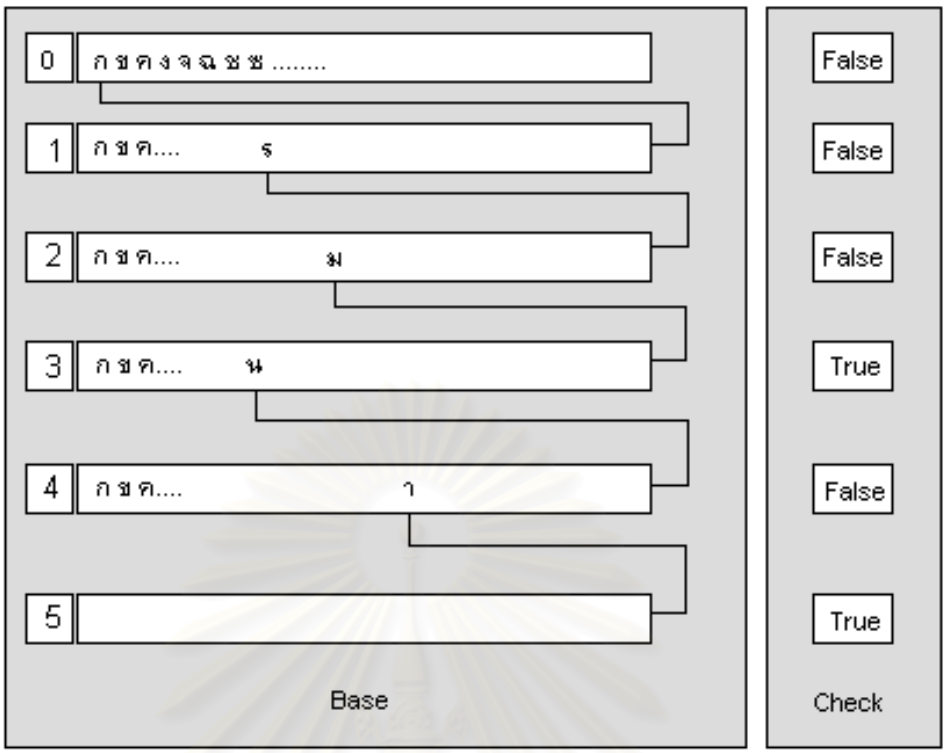
จากรูปที่ 7 ถ้านำข้อมูลคำว่า กระจกตา มาท่องจะผ่านโนดหมายเลข 1, 2, 6, 19, 20, 21, 22 และ 23 ซึ่ง 23 จะเป็นสถานะสิ้นสุดแสดงว่าคำดังกล่าวมีอยู่ใน Trie ขณะเดียวกันถ้านำคำว่า กรรมกร มาท่องจะไม่สามารถท่องไปจนถึงสถานะสิ้นสุดได้แสดงว่าคำดังกล่าวไม่ได้อยู่ใน Trie นี้

การสร้าง Trie สามารถทำได้สองวิธีคือ Tripple-Array Trie และ Double-Array Trie สำหรับโปรแกรม CTTEX นั้นจะใช้ Double-Array Trie โดยการสร้างอะเรย์ขึ้นมาสองอัน อันหนึ่งมีชื่อว่า base และอีกอันหนึ่งคือ check โดย base จะเป็นอะเรย์แบบสองมิติ มิติแรกจะเป็นจำนวนโนด และมิติที่สองจะมีค่าตั้งแต่ 0 ถึง 95 การที่มี 96 ช่องเพราะว่าตัวอักษรภาษาไทยมีทั้งหมด 96 ตัว ส่วน check จะเป็นอะเรย์แบบหนึ่งมิติโดยมีช่องเท่ากับจำนวนโนด

ก่อนหน้าที่จะสร้าง Trie จะไม่สามารถทราบจำนวนโนดได้ อะเรย์สำหรับสร้างจำนวนโนดจึงจำเป็นจะต้องกำหนดไว้มาก หรือใช้โครงสร้างข้อมูลแบบที่สามารถขยายได้ แต่เนื่องจากโปรแกรม CTTEX ได้มีการนำพจนานุกรมรวมเข้าไปกับโปรแกรมจึงทำให้จำนวนโนดมีค่าแน่นอนและไม่มีปัญหาข้อนี้

ยกตัวอย่างเช่น ต้องการนำคำ กรม และ กรมนา ใส่ลงใน Trie โดยเริ่มต้น Trie ยังไม่มีคำศัพท์ใดๆ ทั้งสิ้นเริ่มต้นที่โนดที่ 0 นำ ก เข้ามาตรวจสอบพบว่า base[0][ก] = 0 จึงทำการสร้างโนดใหม่ให้ ก เป็นโนดที่ 1 และกำหนดค่า base[0][ก] = 1 นำ ร เข้ามาตรวจพบว่า base[1][ร] = 0 จึงทำการสร้างโนดใหม่ให้ ร เป็นโนดที่ 2 และกำหนดค่า base[1][ร] = 2 นำ ม เข้ามาตรวจพบว่า base[2][ม] = 0 จึงทำการสร้างโนดใหม่ให้ ม เป็นโนดที่ 3 และกำหนดค่า base[2][ม] = 3 เมื่อหมดคำแล้วจึงไปเปลี่ยนค่า check[3] ให้เป็นจริง

ต่อมานำคำว่า กรมนา ใส่ลงใน Trie เริ่มต้นนำ ก เข้ามาตรวจพบว่า base[0][ก] = 1 จึงนำ ร เข้ามาตรวจกับ base[1][ร] พบว่ามีค่าเป็น 2 จึงนำ ม เข้ามาตรวจกับ base[2][ม] พอว่ามีค่าเป็น 3 จึงนำ น เข้ามาตรวจกับ base[3][น] พบว่า base[3][น] = 0 จึงทำการสร้างโนดใหม่ให้กับ น เป็นโนดที่ 4 และกำหนดค่า base[3][น] = 4 แล้วจึงนำ า เข้ามาตรวจพบว่า base[4][า] = 0 จึงทำการสร้างโนดใหม่ให้ า มีค่าเป็น 5 แล้วเปลี่ยนค่า base[4][า] = 5 เมื่อหมดคำแล้วจึงเปลี่ยนค่า check[5] ให้เป็นจริง ผลจากการสร้าง Trie แสดงในรูปที่ 8



รูปที่ 8 ลักษณะของ Double-Array Trie

เมื่อต้องการนำมาตรวจสอบว่าสามารถตัดคำได้กี่วิธีก็เพียงแต่ท่องไปตาม Trie และตรวจสอบค่า check ไปเรื่อยๆ ตัวอักษรใดที่ check เป็นจริงแสดงว่าสามารถตัดคำที่ตำแหน่งนั้นได้ อย่างคำว่า กรมนา จะพบ check 2 ที่คือตรงคำว่า กรม และคำว่า กรมนา จึงเก็บเอาไว้เพื่อพิจารณาตัดคำต่อไป

หลังจากท่องไปใน Trie แล้วจะได้กลุ่มของคำที่เป็นไปได้ทั้งหมด โปรแกรมจะทำการวนเรียกตัวเองจนทำครบทั้งประโยค เพื่อตัดคำทุกวิธีการและเลือกผลลัพท์ที่ให้จำนวนคำน้อยที่สุดมาใช้ในการตัดคำให้ประโยคนั้น

2.2.2 การศึกษาเปรียบเทียบวิธีในการแยกหมวดหมู่ให้กับคำภาษาจีน [3]

งานวิจัยนี้ได้ทำการศึกษาเปรียบเทียบ วิธีการแยกหมวดหมู่ให้กับเอกสารภาษาจีนแบบอัตโนมัติ ในงานวิจัยนี้จะทดสอบทั้งหมด 3 วิธีคือ k Nearest Neighbor (kNN), Support Vector Machines (SVM) และ Adaptive Resonance Associative Map (ARAM)

โดยเอกสารที่นำมาใช้ในการทดลองคือ TREC-5 People's Daily News ซึ่งมีเอกสารอยู่ประมาณ 33,000 เอกสาร และสามารถแบ่งออกได้เป็น 6 หมวดหมู่ใหญ่ๆ ได้แก่ การเมือง กฎหมายและสังคม (Politics, Law and Society) วรรณคดีและศิลปะ (Literature and

Arts) การศึกษา วิทยาศาสตร์และวัฒนธรรม (Education, Science and Culture) กีฬา (Sports) ทฤษฎีและสถาบันการศึกษา (Theory and Academy) และ เศรษฐศาสตร์ (Economics)

ผลการทดลองได้สรุปไว้ว่า ที่จำนวนเอกสารที่มากพอ (มากกว่า 200 เอกสารต่อหมวดหมู่) จะให้ผลการแยกหมวดหมู่อยู่ในเกณฑ์ที่ดี ทั้ง 3 วิธี โดยที่ ARAM จะให้ผลดีกว่า kNN และ SVM อยู่เล็กน้อย แต่เมื่อลดจำนวนข้อมูลลง SVM จะให้ผลแย่ลงมากที่สุด

เนื่องจากงานวิจัยนี้ใช้ภาษาจีนซึ่งต้องการการตัดคำเหมือนกับคำภาษาไทย จึงนำมาใช้ศึกษาขั้นตอนการวิจัย เพื่อเป็นแนวทางในการทำการวิจัยกับเวบเพจภาษาไทย

บทที่ 3
การเตรียมข้อมูล

ในบทนี้จะกล่าวถึงขั้นตอนเตรียมข้อมูลเวบเพจภาษาไทย เพื่อที่จะนำไปใช้ในการวิเคราะห์ผล ตั้งแต่การเก็บข้อมูล การแยกส่วนประกอบของเวบเพจ การตัดคำ และการเตรียมรายการคำทั่วไป

3.1 การเก็บข้อมูล

ในงานวิจัยนี้จะใช้ข้อมูลเวบเพจภาษาไทยทั้งหมด 2 ชุด โดยชุดแรก (ThaiCat2) เป็นเวบไซต์วิทยาศาสตร์และเทคโนโลยี ส่วนชุดที่ 2 (ThaiCat3) เป็นเวบไซต์กีฬาและนันทนาการ อิงตามเวบไดเรกทอรี่ของสยามกูรู [16] โดยหมวดหมู่ย่อยของแต่ละชุดจะแสดงตามตารางที่ 1

ตารางที่ 1 หมวดหมู่ย่อยในข้อมูลแต่ละชุด

วิทยาศาสตร์และเทคโนโลยี (230 เวบไซต์)	จำนวนหน้า
การเกษตร	510 หน้า
ดาราศาสตร์	218 หน้า
วิศวกรรมศาสตร์	661 หน้า
สิ่งแวดล้อม	251 หน้า
กีฬาและนันทนาการ (378 เวบไซต์)	**จำนวนหน้า**
งานอดิเรก ของสะสม	326 หน้า
สัตว์เลี้ยง	508 หน้า
กีฬา	1013 หน้า
ร้านอาหาร ภัตตาคาร	269 หน้า

ในงานวิจัยนี้ใช้โปรแกรมท่องเวบที่พัฒนาขึ้นจากภาษาพีเอชพี (PHP) โดยมีขั้นตอนคราวๆ ดังต่อไปนี้

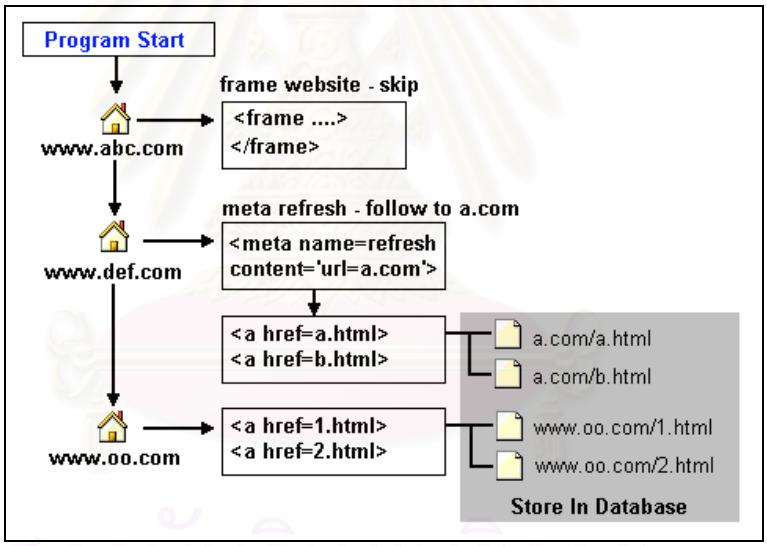
3.1.1 เก็บรายชื่อเวบไซต์ที่เกี่ยวข้องกับหัวข้อที่ต้องการจากเวบไดเรกทอรี่ของสยามกูรู ในแต่ละหมวดหมู่จะต้องมีเวบไซต์มากกว่า 50 เวบไซต์ เพื่อตอนที่ไปเก็บข้อมูลเวบเพจมาจะได้มีข้อมูลมากเพียงพอต่อการทดลอง

3.1.2 โปรแกรมท่องเวบจะออกไปเก็บหน้าเวบเพจหน้าแรกของเวบไซต์แต่ละเวบไซต์ ถ้าหน้าแรกของเวบไซต์ใช้เฟรม โปรแกรมจะข้ามเวบไซต์ดังกล่าวไป ถ้าหน้าแรกของเวบไซต์

ดังกล่าวมีการใช้เมต้าแท็ก (Meta Tag) แบบรีเฟรช (Refresh) โปรแกรมจะอ่านค่าเวบเพจที่อยู่ในแท็กดังกล่าวออกมา และไปอ่านข้อมูลจากเวบเพจดังกล่าวแทน และในกรณีที่หน้าแรกของเวบไซต์ดังกล่าวเปลี่ยนเส้นทางไปยังเวบไซต์อื่น (Redirection) โปรแกรมจะเปลี่ยนไปตามเส้นทางดังกล่าวด้วยเช่นกัน

3.1.3 หาตัวเชื่อมโยง (Link) ไปยังเวบเพจอื่นๆ ที่อยู่ในเวบไซต์เดียวกัน (Internal Link) แล้วจึงไปเก็บหน้าเวบเพจตามลิงค์เหล่านั้นมา เอกสารที่เก็บจะต้องมีจริง คือโค้ดที่ตอบกลับมาจากเวบเซิร์ฟเวอร์จะต้องมีค่าเท่ากับ 200 และจะต้องเป็นเอกสารที่มีชนิดของเอกสาร (Content-type) เป็น text/html เท่านั้น หากว่าเป็นแบบอื่น โปรแกรมจะไม่สนใจข้อมูลเหล่านั้นและข้ามไปอ่านตัวเชื่อมโยงถัดไป แต่ละตัวเชื่อมโยงที่เรียกข้อมูลสำเร็จ โปรแกรมจะเก็บหน้าเวบเพจลงฐานข้อมูล

ขั้นตอนการทำงานคร่าวๆ ของโปรแกรมท่องเวบไซต์จะทำงานดังรูปที่ 9 หลังจากท่องเวบไซต์จนครบทุกเวบไซต์แล้วจะได้ฐานข้อมูลเวบเพจ ซึ่งจะนำไปประมวลผลเวบเพจต่อไป



รูปที่ 9 สรุปขั้นตอนการทำงานของโปรแกรมท่องเวบไซต์

โปรแกรมท่องเวบเพื่อเก็บข้อมูลมาไว้ในฐานข้อมูลนี้เขียนด้วยภาษาพีเอชพี ส่วนหนึ่งเพราะว่าต้องการใช้ทำงานได้ทั้งบนวินโดว์ และลีนุกซ์ โดยในงานวิจัยนี้จะทำงานบนเครื่องลีนุกซ์ ซึ่งเป็นเครื่องเวบเซิร์ฟเวอร์ของเวบไซต์สยามโซน.คอม เหตุที่เลือกเซิร์ฟเวอร์ดังกล่าวเพราะว่าเซิร์ฟเวอร์ตั้งอยู่ที่บริษัทอินเตอร์เน็ตประเทศไทย ซึ่งมีช่องสัญญาณที่ใหญ่มาก ทำให้การเก็บข้อมูลจำนวนร่วม 2,000 หน้า ใช้เวลาเพียง 2 วัน

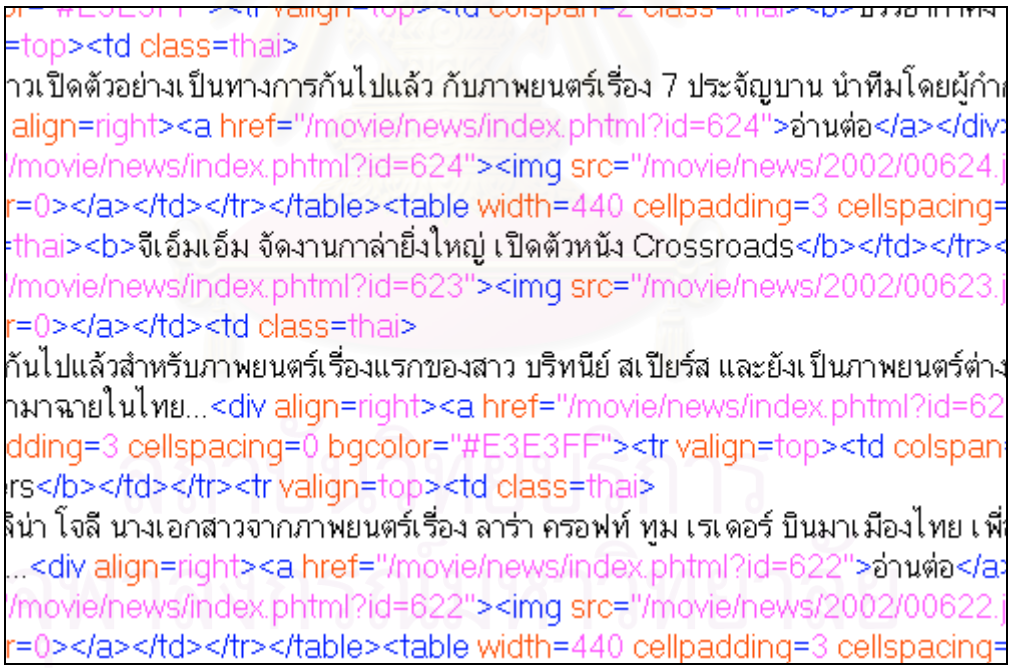
3.2 การแยกส่วนประกอบของเวบเพจ

เวบเพจนั้นจะต่างจากเอกสารปกติตรงที่จะมีการใส่แท็กต่างๆ คร่อมอยู่ระหว่างเอกสาร (รูปที่ 10) ซึ่งเมื่อนำไปแสดงบนเวบบราวเซอร์แล้วจะให้ผลที่สวยงาม มีการเน้นคำ ทำตัวเอียง แทรกรูปได้ และสามารถเชื่อมโยงไปยังเวบเพจอื่นๆ ได้อีกด้วย (รูปที่ 11)

แต่นอกจากความสวยงามของการแสดงผลแล้ว คำที่อยู่ระหว่างแท็กต่างๆ ย่อมมีความสำคัญต่างกันด้วย เช่น คำที่อยู่ระหว่างแท็กที่ทำตัวหนา (...) น่าจะมีความสำคัญมากกว่าคำที่ไม่ได้ทำตัวหนา

นอกจากนี้ยังมีแท็กบางชนิดที่จะมีข้อมูลที่สำคัญซ่อนอยู่ในคุณลักษณะ (Attribute) ซึ่งข้อความเหล่านี้เป็นข้อความที่ผู้เขียนเวบเพจใส่เพิ่มเข้ามา เพื่อเป็นประโยชน์กับเวบไซต์ที่ให้บริการค้นหาข้อมูล แท็กประเภทนี้ เช่น META และ IMG เป็นต้น

ในการแยกหมวดหมู่ให้กับเวบเพจนั้น ถ้าเราให้ความสำคัญกับคำที่อยู่ในแท็กที่สำคัญมากขึ้น น่าจะทำให้ความถูกต้องเพิ่มมากขึ้นด้วยเช่นกัน ดังนั้นหลังจากที่ได้เวบเพจมาแล้วจึงนำมาเข้าโปรแกรมสำหรับดึงข้อมูลที่อยู่ในแท็กเอชทีเอ็มแอล



รูปที่ 10 ตัวอย่างโค้ดเอชทีเอ็มแอลของเวบเพจ

รูปที่ 11 เวบเพจเมื่อนำไปแสดงผลในเวบบราวเซอร์

ส่วนที่ทำหน้าที่แยกส่วนประกอบของเอกสารเอชทีเอ็มแอลจะเรียกว่า HTML Parser โดยจะเป็นหนึ่งของงานวิจัยนี้ด้วย เขียนในรูปของออปเจ็คของเดลไฟล์ รุ่นที่ 6 (ไม่สามารถใช้กับเดลไฟล์รุ่นต่ำกว่าได้)

ขั้นตอนหลักในการนำเวบเพจมาแยกส่วนประกอบมีดังต่อไปนี้

3.2.1 นำเวบเพจมาลบแท็กและข้อมูลที่อยู่ระหว่างแท็กที่ไม่มีความจำเป็น และไม่มีข้อมูลต่อการแยกหมวดหมู่ก่อน แท็กดังกล่าวได้แก่

Script เป็นแท็กที่แจ้งไปเวบบราวเซอร์ประมวลผลคำสั่งพิเศษ (Client Site Script)

Style เป็นแท็กที่เก็บคลังรูปแบบ (Style Sheet) สำหรับการแสดงผล

Iframe เป็นแท็กที่จะแทรกเวบเพจ ไว้ในเวบเพจ นิยมใช้ในการแสดงแบนเนอร์โฆษณา

Embed เป็นแท็กที่จะแทรกโปรแกรมเสริม (Plug-in) ไว้ในหน้าเวบเพจ สำหรับโปรแกรมเวบบราวเซอร์ Netscape Navigator

Object เหมือน Embed แต่เป็นของเวบบราวเซอร์ Microsoft Internet Explorer

Applet เป็นแท็กที่จะแทรกโปแกรมจาวา ไว้ในหน้าเวบเพจ

3.2.2 หลังจากตัดแท็กที่ไม่ต้องการทิ้งไปแล้ว จึงเริ่มเก็บข้อมูลคุณลักษณะ ที่มีความสำคัญต่อการแยกหมวดหมู่ ในที่นี้คือ คุณลักษณะ ALT ของแท็ก IMG คุณลักษณะ CONTENT ของแท็ก META NAME=KEYWORD และ META NAME=DESCRIPTION แล้วจึง

เก็บข้อมูลจากแท็กที่เป็นคู่ โดยข้อมูลที่ต้องการจะเป็นข้อมูลที่อยู่ระหว่างแท็ก แท็กที่พิจารณาในงานวิจัยนี้ได้แก่ H1, H2, H3, H4, H5, H6, TITLE, B, I, A HREF

3.2.3 นำเวบเพจดังกล่าวมาลบแท็กออกทั้งหมด เพื่อให้เหลือแต่เฉพาะข้อความ

3.3 การตัดคำภาษาไทย

สำหรับภาษาไทยนั้น คำแต่ละคำไม่มีการแบ่งอย่างชัดเจน ดังนั้นหากต้องการคำของเอกสาร จะต้องทำการตัดคำเสียก่อน ในส่วนนี้ได้นำโปรแกรม CTTEX [1] ซึ่งพัฒนาโดย วุฒิชัย อัมพรอร่ามเวทย์ มาพัฒนาต่อ โดยใช้อัลกอริทึมในการค้นหาคำจากพจนานุกรมเดิม

เนื่องจากโปรแกรมนี้ใช้กระบวนการเรียกซ้ำเพื่อหาวิธีการตัดคำที่เป็นไปได้มากที่สุด แล้วจึงใช้วิธีการที่ดีที่สุดเป็นคำตอบ ซึ่งถ้าประโยคที่ยาวมากๆ อาจจะต้องใช้เวลาหลายนาที อีกทั้งพจนานุกรมของโปรแกรม CTTEX ยังมีเพียง 7,000 คำเท่านั้น ในโปรแกรมที่พัฒนาขึ้นมาใหม่นั้นใช้คำมากถึง 25,000 คำ เพื่อให้ได้ผลการตัดคำถูกต้องมากขึ้น

จากข้อมูลที่เก็บมาจากเวบไซต์มีบางเวบเพจมีการพิมพ์ภาษาไทย ติดกันโดยไม่มีช่องว่างหรือสัญลักษณ์ถึง 400 ตัวอักษร ซึ่งถ้า 1 คำมีประมาณ 6 ตัวอักษร จะมีคำทั้งหมดถึง 66 คำในประโยคนั้นและใช้เวลาในการตัดคำนานถึง 30 นาทีต่อประโยค ซึ่งในทางปฏิบัติแล้วถือว่านานเกินไป เมื่อจำนวนคำในพจนานุกรมมากขึ้น ย่อมทำให้ความเร็วในการตัดคำลดลง วิธีการแก้ปัญหาคือ แทนที่จะค้นหาทุกกรณีที่สามารถตัดคำได้ จึงสร้างเงื่อนไขในการหยุดการค้นหา และกลับไปลองวิธีการตัดแบบอื่น แทนวิธีการเดิมซึ่งจะนับจากจำนวนคำหลังจากตัดคำเสร็จทั้งประโยคแล้ว

เนื่องจากโปรแกรมต้องการตัดคำให้ได้จำนวนคำน้อยที่สุด ดังนั้น จำนวนคำต่อจำนวนตัวอักษรก็ย่อมน้อยที่สุดด้วย กำหนดให้ WPL (Words per Length) เป็นจำนวนคำในประโยคนั้นหารด้วยจำนวนตัวอักษรของประโยคนั้น

ตัวอย่างเช่น ถ้าประโยคมี 20 ตัวอักษร วิธีการที่ตัดได้ 5 คำจะมีค่า WPL เป็น 0.25 แต่ถ้าตัดได้ 4 คำจะมีค่า WPL เป็น 0.2 ในโปรแกรมจะเลือกวิธีการที่ได้ WPL เป็น 0.2 นั่นเอง ประโยชน์ของ WPL คือ สามารถหาค่า ได้แม้ว่าจะยังตัดคำไม่ครบทั้งประโยค เช่น ตามตัวอย่างด้านบน ถ้าตัดไปแล้ว 10 ตัวอักษรได้ 2 คำ ค่า WPL จะเป็น 0.2 ไม่เหมือนกับวิธีการตัดโดยเลือกจำนวนคำน้อยที่สุด เพราะว่าจะทราบจำนวนคำได้ก็ต่อเมื่อตัดครบทั้งประโยคแล้ว

เนื่องจากการตัดคำนั้นมีความเป็นไปได้ที่จะเจอคำศัพท์ที่ไม่พบในพจนานุกรม อีกทั้งวิธีการตัดคำที่ได้คำจากพจนานุกรมน่าจะมีความถูกต้องมากกว่า จึงมีการเพิ่มความไม่

สำคัญให้กับคำที่ตัดได้แต่ไม่ได้อยู่ในพจนานุกรม โดยกำหนดขึ้นมาว่าถ้าตัดคำที่ไม่พบในพจนานุกรมค่า WPL จะต้องเพิ่มขึ้นมากกว่าตัดคำที่อยู่ในพจนานุกรม จึงเขียนสมการสำหรับหาค่าของ WPL ใหม่เป็น

$$WPL = \frac{WD + (WOD \times 3) + COD}{C} \quad (10)$$

WD = จำนวนคำที่อยู่ในพจนานุกรม

WOD = จำนวนคำที่ไม่อยู่ในพจนานุกรม

COD = จำนวนตัวอักษรของคำที่ไม่อยู่ในพจนานุกรม

C = จำนวนตัวอักษรตั้งแต่ต้นประโยคถึงจุดที่กำลังตัดคำ

เลข 3 นั้นเป็นเลขที่เกิดขึ้นจากการทดลอง ว่าได้ผลลัพท์ดีที่สุด ตัดคำได้ถูกต้องเป็นที่น่าพอใจ ในเวลาที่เหมาะสม โดยได้ลองปรับเป็นเลข 2 จะใช้เวลามากขึ้น ขณะที่ถ้าเป็นเลข 4 โปรแกรมจะพยายามตัดคำให้อยู่ในพจนานุกรมมากเกินไปจนทำให้ผลที่ได้ผิดพลาดสูง

โดยขณะเริ่มโปรแกรม จะหาวิธีการที่ตัดคำได้เป็นวิธีการแรกแล้วเก็บค่าไว้ใน WPL_{min} จากนั้นขณะที่กำลังตัดคำวิธีการอื่นอยู่ด้วยนั้น โปรแกรมจะหาค่า WPL ไปด้วยตลอด โดยทันทีที่ค่า WPL×ค่าแฟกเตอร์ความผิดพลาด (ในงานวิจัยนี้ใช้ 0.85) มีค่ามากกว่า WPL_{min} โปรแกรมจะหยุดตัดคำวิธีการดังกล่าว และไปลองตัดคำด้วยวิธีการอื่นทันที

ค่าแฟกเตอร์ความผิดพลาด ถ้ามีค่าน้อยจะทำให้เวลาในการตัดคำมากขึ้น แต่จะเพิ่มความถูกต้องให้กับการตัดคำด้วยเช่นกัน ขณะที่ถ้าค่านี้มีค่ามากความถูกต้องในการตัดคำจะน้อยลง แต่ใช้เวลาในการตัดคำน้อยลงด้วยเช่นกัน

จากการทดลองเอกสารปกติ ที่ไม่มีคำที่ยาวมากเกินกว่า 100 ตัวอักษรต่อประโยค จะใช้เวลาประมาณ 5-10 วินาทีเท่านั้น ที่เวบเพจประมาณ 2,000 เวบเพจ จะใช้เวลาตัดคำประมาณ 45 นาที (แต่ถ้าเปลี่ยนค่าแฟกเตอร์ความผิดพลาดเป็น 0.9 จะใช้เวลาประมาณ 15 นาที) (บนเครื่อง Celeron 800 MHz)

3.4 การเตรียมรายการคำทั่วไป (Stopword)

คำทั่วไปคือคำที่มักจะปรากฏในเอกสารได้ทั่วไป ไม่มีประโยชน์ในการแยกหมวดหมู่ เนื่องจากคำเหล่านี้มักจะมีประโยชน์ในแง่ของการทำให้รูปประโยคครบสมบูรณ์เท่านั้น วิธีการสร้างรายการคำทั่วไปจะแบ่งออกเป็น 2 ส่วน คือ ภาษาไทย และ ภาษาอังกฤษ

3.4.1 คำทั่วไปภาษาไทย

รายการคำทั่วไปภาษาไทยจะสร้างจากก้อนข้อมูลออร์คิด (Orchid Corpus)[12] ข้อมูลนี้จะมีการกำหนดประเภทของคำ (Part-of-Speech) เอาไว้ด้วยแล้ว จึงเลือกมาเฉพาะคำที่มีประเภทดังต่อคือ คำลงท้าย คำสรรพนาม คำบอกสถานที่ คำสันธาน และคำบุพบท

3.4.2 คำทั่วไปภาษาอังกฤษ

รายการคำทั่วไปภาษาอังกฤษนำมาจากเวบไซต์ Business & Management Practices [17] ซึ่งได้มีการรวบรวมคำทั่วไปจากระบบค้นหาข้อมูลภายในเวบไซต์ เหตุที่เลือกที่นี่เพราะว่ามีจำนวนคำที่ไม่มากไม่น้อยเกินไปและพบเป็นอันดับต้นระหว่างการค้นหาข้อมูล

บทที่ 4
การทดลองและผลการทดลอง

บทนี้กล่าวถึงขั้นตอนการทดลอง ผลการทดลอง โดยจะใช้ข้อมูลภาษาไทย 2 ชุด ชุดละ 4 หมวดหมู่ และ ข้อมูลภาษาอังกฤษ 1 ชุด ชุดละ 4 หมวดหมู่ โดยจะทดลอง ใน 3 ส่วนหลัก คือ ข้อความในแท็กเอชทีเอ็มแอลมีความสำคัญมากแค่ไหน การลดมิติของคำช่วยทำให้ผลการแยกหมวดหมู่ดีขึ้นหรือไม่ และ ประสิทธิภาพของเครื่องมือที่ใช้ในการเรียนรู้

4.1 วิธีการทดลอง

การทดลองนี้ใช้ข้อมูลจากการท่องไปตามเวบไซต์ต่างๆ ซึ่งในแต่ละเวบไซต์นั้นๆ ได้มีการกำหนดหมวดหมู่ให้กับเวบไซต์แต่ละเวบไซต์อยู่แล้ว และให้ถือว่าในแต่ละเวบเพจก็จะมีหมวดหมู่เดียวกับเวบไซต์ด้วย โดยเพื่อให้ผลการทดลองที่ได้มีความน่าเชื่อถือมากยิ่งขึ้นจึงได้เตรียมข้อมูลเวบเพจภาษาไทยเอาไว้สองชุด และข้อมูลภาษาอังกฤษสำหรับเปรียบเทียบอีก 1 ชุด

โดยลักษณะของข้อมูลที่นำมาใช้ในงานวิจัยนี้ทั้ง 3 ชุดนั้นจะแสดงตาม ตารางที่ 2 ตารางที่ 3 และ ตารางที่ 4

ตารางที่ 2 ลักษณะของชุดข้อมูลวิทยาศาสตร์และเทคโนโลยี (ภาษาไทย)

ThaiCat2 – เอกสารเกี่ยวกับวิทยาศาตร์และเทคโนโลยี http://www.siamguru.com/d/100260169.html	
เอกสารทั้งหมด	1,703 หน้า
หมวดหมู่	การเกษตร 510 หน้า ดาราศาสตร์ 281 หน้า วิศวกรรมศาสตร์ 661 หน้า สิ่งแวดล้อม 251 หน้า
จำนวนคำโดยประมาณ	27,000 คำ
จำนวนคำไทยโดยประมาณ	12,000 คำ
ข้อมูลสอน	1,135 หน้า
ข้อมูลทดสอบ	568 หน้า

ตารางที่ 3 ลักษณะของชุดข้อมูลกีฬาและนันทนาการ (ภาษาไทย)

ThaiCat3 – เอกสารเกี่ยวกับกีฬาและนันทนาการ http://www.siamguru.com/d/100320256.html	
เอกสารทั้งหมด	2,116 หน้า
หมวดหมู่	งานอดิเรกของสะสม 326 หน้า สัตว์เลี้ยง 508 หน้า กีฬา 1013 หน้า ร้านอาหารและภัตตาคาร 269 หน้า
จำนวนคำโดยประมาณ	30,000 คำ
จำนวนคำไทยโดยประมาณ	14,000 คำ
ข้อมูลสอน	1411 หน้า
ข้อมูลทดสอบ	705 หน้า

ตารางที่ 4 ลักษณะของชุดข้อมูลงานอดิเรก (ภาษาอังกฤษ)

ThaiCat4 – เอกสารเกี่ยวกับงานอดิเรก	
เอกสารทั้งหมด	192 หน้า
หมวดหมู่	ดาราศาสตร์ 48 หน้า รถยนต์ 48 หน้า รถจักรยานยนต์ 48 หน้า ดนตรี 48 หน้า
จำนวนคำโดยประมาณ	16,000 คำ
จำนวนคำไทยโดยประมาณ	0 คำ
ข้อมูลสอน	128 หน้า
ข้อมูลทดสอบ	64 หน้า

ข้อมูล ThaiCat2 และ ThaiCat3 แม้ว่าจะเป็นข้อมูลภาษาไทยเหมือนกัน แต่ข้อมูล ThaiCat2 จะมีข้อมูลแต่ละหมวดหมู่ที่ใกล้เคียงกันมากกว่า ThaiCat3 ซึ่งจะเนื้อหาที่แบ่งเป็นหมวดหมู่อย่างชัดเจนกว่า

ทุกๆ การทดลองจะนำข้อมูลที่ต้องการทดสอบมาแบ่งเป็น 3 ส่วน (3 Folds) โดยให้มีหน้าเวบเพจจากแต่ละหมวดหมู่ในสัดส่วนที่เท่ากัน การทดสอบครั้งแรกจะนำข้อมูลส่วนที่ 2 และ 3 ไปสอน และนำข้อมูลส่วนที่ 1 ไปทดสอบ การทดสอบครั้งที่สองจะนำข้อมูลส่วนที่ 1 และ 3 ไปสอน และนำข้อมูลส่วนที่ 2 มาทดสอบ การทดสอบครั้งที่สามจะนำข้อมูลส่วนที่ 1 และ 2 ไปสอน และนำข้อมูลส่วนที่ 3 มาทดสอบ ค่าความถูกต้องจะได้จากการเฉลี่ยค่าทั้ง 3 ค่า

สำหรับข้อมูลชุดที่ 3 (ThaiCat4) เป็นข้อมูลที่นำมาเปรียบเทียบเท่านั้น ดังนั้นจึงไม่ได้ทำการทดลองทุกแบบเหมือนข้อมูลชุดที่ 1 และ ข้อมูลชุดที่ 2

ในงานวิจัยนี้จะใช้โปรแกรม Rainbow [10] ซึ่งพัฒนาโดย McCallum, Andrew Kachites จุดมุ่งหมายหลักของโปรแกรมนี้คือต้องการให้เป็นเครื่องมือสำหรับ การเรียกคืนข้อความ (Text Retrieval) การแยกหมวดหมู่ (Classification) และ การหากลุ่มก้อน (Clustering) โดยใช้รูปแบบทางสถิติ ภายในโปรแกรมสามารถการแยกหมวดหมู่ได้มากมายหลายวิธี ไม่ว่าจะเป็น Active Learning, Maximum Entropy, Probabilistic Indexing, Naïve Bayes, K-nearest neighbor และ Support Vector Machines เดิมทีโปรแกรมนี้พัฒนาขึ้นมาใช้บนระบบปฏิบัติการลีนุกซ์ แต่ในงานวิจัยนี้ได้นำโปรแกรมดังกล่าวมาคอมไพล์ใหม่ให้สามารถใช้งานบนระบบวินโดวส์ได้ โดยจะใช้โปรแกรม Rainbow ในการคำนวณค่า Information Gain การแยกหมวดหมู่ด้วยตัวแยกแยะเบย์อย่างง่ายและเอสวีเอ็มแบบเชิงเส้น

ในงานวิจัยนี้ยังใช้โปรแกรม SVM^{Light} [14] สำหรับการทดลองที่ใช้การแยกหมวดหมู่แบบเอสวีเอ็ม ที่ใช้เคอร์เนลแบบอาร์บีเอฟ โดย การแยกหมวดหมู่แบบหลายหมวดหมู่จะใช้วิธีการ 1-v-R

โดยจุดมุ่งหมายหลักของการทดลองแบ่งออกเป็น 5 ข้อดังต่อไปนี้

4.1.1 การทดลองเพื่อหาความสำคัญของการเตรียมข้อมูลแบบต่างๆ

ในการทดลองนี้จะนำข้อมูลในแต่ละหมวดหมู่มาสร้างข้อมูลใหม่ 4 แบบ คือ

1. ข้อมูลที่ลบแท็กเอชทีเอ็มแอลทิ้ง (Normal)
2. ข้อมูลที่เพิ่มความถี่ของคำในแท็กเอชทีเอ็มแอล 1 ครั้ง (HTML+) โดยนำข้อมูลจากข้อที่ 1 มาเพิ่มคำที่พบในแท็ก B, I, H1, H2, H3, H4, H5, H6, IMG Alt, META Keyword, META Description และ Title เข้าไปต่อท้ายเอกสาร

3. ข้อมูลที่เพิ่มความถี่ของคำในแท็กเอชทีเอ็มแอล 2 ครั้ง (HTML+2) โดยนำข้อมูลจากข้อที่ 2 มาเพิ่มคำที่พบในแท็กเอชทีเอ็มแอลในข้อที่ 2 เข้าไปอีก 1 ครั้ง
4. ข้อมูลจากข้อ 3 ลบคำทั่วไปทิ้ง (No Stopword)

การทดลองนี้จะเป็นการทดลองอย่างคร่าวๆ ก่อนว่า การเพิ่มคำในแท็กเอชทีเอ็มแอล 1 ครั้ง และ 2 ครั้ง รวมไปถึงการลดจำนวนคำด้วยวิธีการลบคำทั่วไปทิ้ง จะได้ผลความถูกต้องเป็นเช่นไร โดยจะใช้ทั้งตัวแยกแยะเบย์อย่างง่ายและเอสวีเอ็ม นอกจากนี้ยังทดลองกับข้อมูล ThaiCat4 ด้วยเพื่อดูว่าข้อมูลภาษาไทย และข้อมูลภาษาอังกฤษมีผลต่อการแยกหมวดหมู่หรือไม่

4.1.2 การทดลองผลจากการลดมิติของคำ

จากงานวิจัยของ Yang, Y., Pedersen, J. [20] ได้สรุปไว้ว่าค่า IG และค่า CHI ให้ผลในการลดมิติดีที่สุด แต่เนื่องจากค่า CHI นั้นเหมาะกับข้อมูลที่มีความถี่ของคำสูง ในงานวิจัยนี้จึงเน้นไปที่การใช้ค่า IG เข้ามาลดคำโดย จะลองลดจำนวนคำที่มีค่า IG น้อยออกไปทีละ 1,000 คำ ตั้งแต่จำนวนคำสูงสุดของแต่ชุดข้อมูล จนเหลือคำอยู่ 2,000 คำ จากนั้นจึงนำค่าความถูกต้องมาสร้างกราฟ

ในการทดลองนี้เพื่อแสดงให้เห็นว่าการลดมิติของคำโดยการใช้ Information Gain จะมีผลต่อการแยกหมวดหมู่กับตัวแยกแยะเบย์อย่างง่ายและเอสวีเอ็มมากหรือไม่ โดยจะแสดงผลเป็นรูปกราฟแทนเพื่อจะได้เห็นแนวโน้มการเปลี่ยนแปลง

4.1.3 การทดลองประสิทธิภาพของวิธีการแยกหมวดหมู่

ในการทดลองนี้จะแสดงให้เห็นในข้อมูลแบบต่างๆ ตัวแยกแยะเบย์อย่างง่าย และเอสวีเอ็ม จะให้ความถูกต้องเป็นเช่นไร

4.1.4 การทดลองเพื่อหาความสำคัญของข้อมูลในแท็กเอชทีเอ็มแอล

ในการทดลองนี้จะทดสอบว่าการเพิ่มความถี่ให้กับคำที่อยู่ในแท็กเอชทีเอ็มแอลจะเพิ่มความถูกต้องได้หรือไม่ โดยเตรียมข้อมูลจะนำเวบเพจไปผ่านกระบวนการแยกส่วนประกอบเอชทีเอ็มแอล ออกมาเป็น 2 ส่วนใหญ่ๆ คือ (1) ข้อมูลที่ได้มีการลบแท็กเอชทีเอ็มแอลทิ้งไป (2) ข้อมูลที่อยู่ในแท็กเอชทีเอ็มแอล เช่น <B>ปากกา</B> ข้อมูลที่เราสนใจคือคำว่า ปากกา โดยแท็กที่จะดึงข้อมูลเหล่านี้มาใช้จะมีอยู่ 8 ชนิดได้แก่ ตัวหนา (B) ตัวเอียง (I) ข้อมูลหัวข้อ (H1, H2, H3, H4, H5 และ H6) คำอธิบายรูปภาพ (IMG ALT) คำที่อยู่ระหว่างตัวเชื่อมโยง (A) ชื่อเวบเพจ (TITLE) คำสำคัญ (META Keyword) และ คำอธิบาย (META Description)

จากนั้นจึงนำข้อมูลที่ได้จากการแยกส่วนประกอบเอชทีเอ็มแอลทั้ง 2 ส่วน สร้างเป็นข้อมูลใหม่ 16 ชุด โดยชุดแรกจะเป็นข้อมูลที่ได้มีการลบแท็กเอชทีเอ็มแอลทิ้งไป ชุดที่ 2 จะเป็นข้อมูลชุดแรกเพิ่มคำที่อยู่ในแท็กเอชทีเอ็มแอล ชุดที่ 3 จะเป็นข้อมูลชุดที่ 2 มาเพิ่มคำที่อยู่ในแท็กเอชทีเอ็มแอล ชุดที่ 4 จะเป็นข้อมูลชุดที่ 3 มาเพิ่มคำที่อยู่ในแท็กเอชทีเอ็มแอล และเป็นเช่นนี้ไปเรื่อยๆ จนครบทั้ง 16 ชุด

4.1.5 การทดลองว่า Information Gain ช่วยลดเวลาในการใช้เอสวีเอ็ม

เนื่องการเอสวีเอ็มจะต้องมีการสร้างฐานความรู้จากข้อมูลที่สอนให้ ซึ่งระยะเวลาในการสอนขึ้นอยู่กับข้อมูล ต่างกับตัวแยกแยะเบย์อย่างง่ายที่ใช้เพียงความน่าจะเป็นของคำในแต่ละหมวดหมู่เท่านั้น ในการทดลองนี้จึงจับเวลาที่เอสวีเอ็มใช้ในการเรียนรู้ข้อมูลที่ยังไม่ได้มีการลดคำ และข้อมูลที่มีการลดคำแล้ว โดยเวลาทั้งหมดที่ได้ในการทดลองนี้ไม่รวมเวลาในการอ่านเขียนดีสก์ เป็นเวลาที่ใช้ในการคำนวณจริงๆ

4.2 ผลการทดลอง

4.2.1 การทดลองเพื่อหาความสำคัญของการเตรียมข้อมูลแบบต่างๆ

ในการทดลองนี้ได้นำข้อมูลทั้ง 3 กลุ่มมาสร้างข้อมูลใหม่กลุ่มละ 4 ประเภทคือ Normal, HTML+, HTML+2 และ No Stopword และจึงนำไปแยกหมวดหมู่ด้วยตัวแยกแยะเบย์อย่างง่ายและเอสวีเอ็ม ได้ผลการทดลองตามตารางที่ 5

ตารางที่ 5 เปรียบเทียบความถูกต้องของการเตรียมข้อมูลแบบต่างๆ

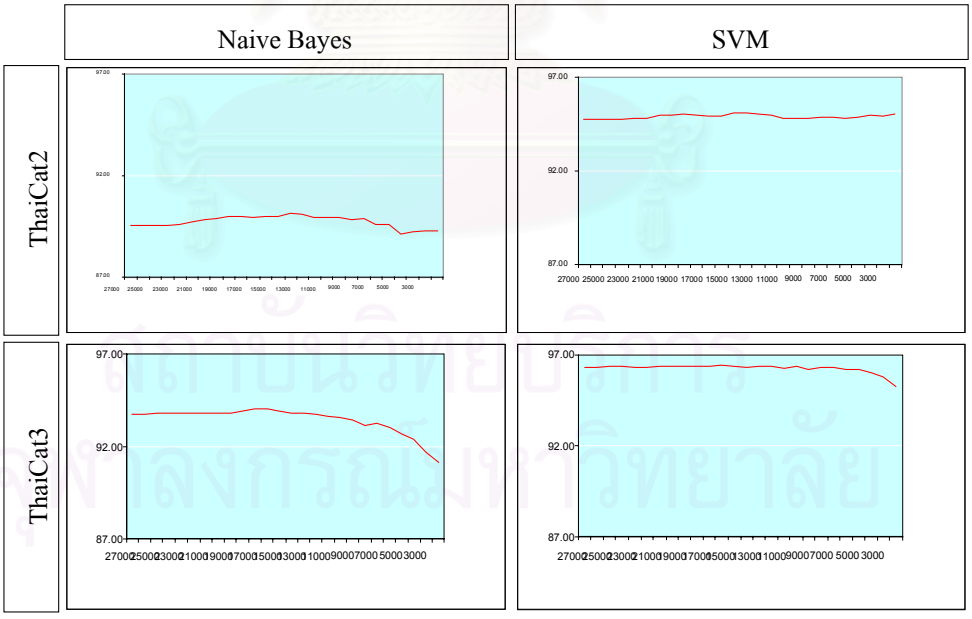
	Naïve Bayes			
	Normal	HTML+	HTML+2	No Stopword
ThaiCat2	86.08%	88.96%	89.55%	90.72%
ThaiCat3	89.98%	93.15%	93.72%	94.47%
ThaiCat4	94.85%	96.31%	96.32%	96.50%
	SVM			
	Normal	HTML+	HTML+2	No Stopword
ThaiCat2	93.66%	94.89%	94.71%	94.89%
ThaiCat3	94.85%	96.31%	96.32%	96.50%
ThaiCat4	94.81%	95.33%	95.85%	97.39%

ผลการทดลองจากตารางที่ 5 แสดงให้เห็นว่าการเตรียมข้อมูลก่อนที่จะทำการแยกหมวดหมู่โดยการเพิ่มคำที่อยู่ในแท็กเอชทีเอ็มแอล และลบคำทั่วไป จะสามารถให้ความถูกต้องมากขึ้นได้ โดยการเพิ่มคำได้แท็กเอชทีเอ็มแอลจะสามารถเพิ่มความถูกต้องให้กับตัวแยกแยะเบย์อย่างง่ายได้มากถึงประมาณ 3 เปอร์เซ็นต์ ซึ่งมากกว่ากับเอสวีเอ็มที่เพิ่มเพียงประมาณ 1.3 เปอร์เซ็นต์เท่านั้น และการเตรียมข้อมูลแบบ No Stopword จะให้ความถูกต้องเพิ่มมากขึ้นประมาณ 4.5 เปอร์เซ็นต์สำหรับตัวแยกแยะเบย์อย่างง่าย และ ประมาณ 1.4 เปอร์เซ็นต์สำหรับเอสวีเอ็ม

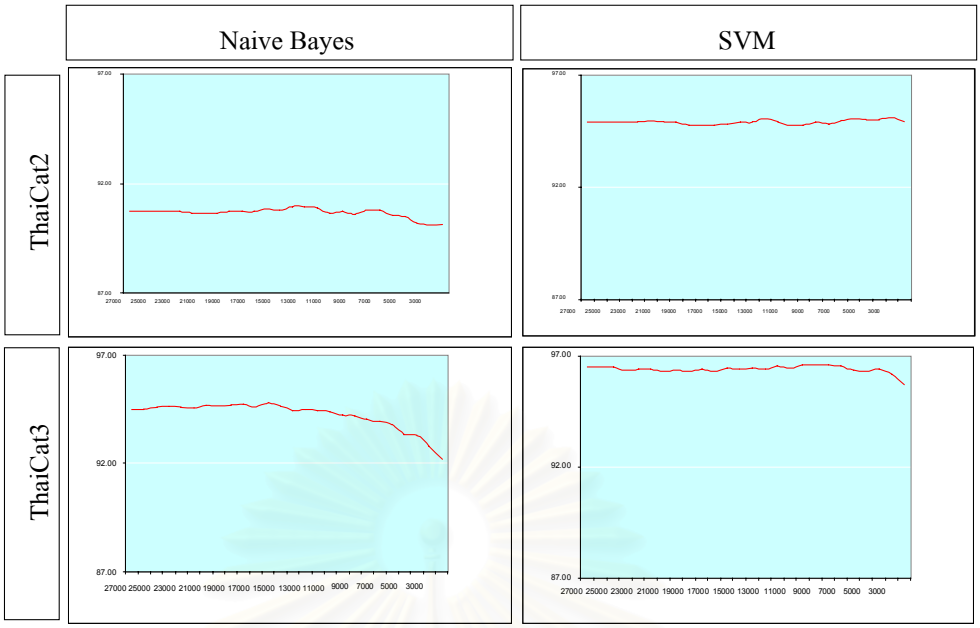
แต่สำหรับข้อมูลภาษาอังกฤษจะให้ความถูกต้องมากกว่าข้อมูลที่มีทั้งภาษาไทยและภาษาอังกฤษอย่างมาก โดยสำหรับตัวแยกแยะเบย์อย่างง่ายจะอยู่ประมาณ 2-6 เปอร์เซ็นต์ ขณะที่เอสวีเอ็มจะอยู่ที่ประมาณ 1-3 เปอร์เซ็นต์

4.2.2 การทดลองผลจากการลดมิติของคำ

การทดลองนี้จะนำข้อมูล HTML+2 และ No Stopword ซึ่งได้รับความถูกต้องสูงที่สุดมาทำการลดมิติของคำด้วยการเลือกคำที่มี Information Gain ต่ำทิ้งไปทีละ 1,000 คำ ตั้งแต่ 27,000 คำลดลงมาถึง 3,000 คำ นำผลที่ได้มาเขียนกราฟเพื่อดูแนวโน้มการเพิ่มและลดความถูกต้องขอบการแยกหมวดหมู่ๆ



รูปที่ 12 กราฟความถูกต้องเมื่อมีการลดมิติในการเตรียมข้อมูลแบบ HTML+2



รูปที่ 13 กราฟความถูกต้องเมื่อมีการลดมิติในการเตรียมข้อมูลแบบ No Stopword

จากรูปที่ 12 และรูปที่ 13 การใช้ Information Gain กับข้อมูล HTML+2 จะสามารถเพิ่มความถูกต้องได้มากกว่าใช้กับข้อมูล No Stopword แสดงว่าคำที่ถูกตัดทิ้งไปเพราะว่ามีค่า Information Gain ต่ำ ส่วนหนึ่งเป็นคำที่อยู่ในรายการคำทั่วไป

ตัวแยกแยะเบย์อย่างง่ายไม่มีความทนทานต่อการลดจำนวนคำด้วย Information Gain โดยเมื่อลดจำนวนคำให้เหลือต่ำว่า 50 เปอร์เซ็นต์ของคำทั้งหมด ความถูกต้องจะลดลง ขณะที่เอสวีเอ็มสามารถลดลงได้ถึง 20 เปอร์เซ็นต์ของคำทั้งหมด

การลดจำนวนคำด้วย Information Gain สามารถเพิ่มความถูกต้องให้กับตัวแยกแยะเบย์อย่างง่ายโดยจะเพิ่มสูงสุดที่การลดคำออกไป 50 เปอร์เซ็นต์ของคำทั้งหมด ขณะที่สำหรับเอสวีเอ็มจะเพิ่มความถูกต้องน้อยมาก

4.2.3 การทดลองประสิทธิภาพของวิธีการแยกหมวดหมู่

ในการทดลองนี้จะแสดงให้เห็นว่าระหว่างตัวแยกแยะเบย์อย่างง่าย และเอสวีเอ็ม วิธีการใดจะให้ผลที่ดีกว่า ทดลองกับข้อมูล ThaiCat2, ThaiCat3 และ ThaiCat4 ที่มีการเตรียมข้อมูลแบบ No Stopword โดยใช้ Information Gain เข้ามาลดมิติของคำด้วย ผลการทดลองสรุปตามตารางที่ 6

ตารางที่ 6 ความถูกต้องของตัวแยกแยะเบย์อย่างง่ายและเอสวีเอ็ม

	No IG		17500		15000		12500	
	NB	SVM	NB	SVM	NB	SVM	NB	SVM
ThaiCat2	90.72%	94.89%	90.61%	94.77%	90.78%	94.89%	90.95%	94.89%
ThaiCat3	94.47%	96.50%	94.61%	96.41%	94.61%	96.41%	94.42%	96.41%
ThaiCat4	96.50%	97.39%	N/A		N/A		96.41%	97.94%

* NB หมายถึงตัวแยกแยะเบย์อย่างง่าย

จากตารางที่ 6 แสดงให้เห็นว่าตัวแยกแยะเบย์อย่างง่ายให้ความถูกต้องต่ำกว่าเอสวีเอ็มในทุกกรณี โดยกับข้อมูล ThaiCat2 จะให้ความถูกต้องมากกว่าประมาณ 4 เปอร์เซ็นต์ ข้อมูล ThaiCat3 จะให้ความถูกต้องมากกว่าประมาณ 2 เปอร์เซ็นต์ และข้อมูล ThaiCat4 ประมาณ 1.21 เปอร์เซ็นต์

จะเห็นได้ว่าข้อมูลภาษาอังกฤษจะให้ความแตกต่างระหว่างความถูกต้องของตัวแยกแยะเบย์อย่างง่ายกับเอสวีเอ็มน้อยกว่าข้อมูลที่มีภาษาไทยด้วย ซึ่งในจุดนี้สามารถสรุปได้ว่าตัวแยกแยะเบย์อย่างง่ายไม่เหมาะกับข้อมูลที่มีจำนวนคำมากอย่างภาษาไทย

ขณะที่เอสวีเอ็มจะสามารถรับข้อมูลที่มีความหลากหลายของคำได้ดี อีกทั้งข้อมูลเหล่านั้นยังไม่มีความจำเป็นต้องลดมิติของคำเพื่อเพิ่มความถูกต้องอีกด้วย สังเกตได้จากความถูกต้องที่การใช้ Information Gain เพื่อตัดคำทิ้งไปที่ค่าต่างๆ ไม่ได้ทำให้ความถูกต้องเพิ่มขึ้นหรือลดลงมากนัก

4.2.4 การทดลองเพื่อหาความสำคัญของข้อมูลในแท็กเอชทีเอ็มแอล

ในการทดลองนี้เพื่อจะทดสอบดูว่าการให้ความสำคัญของคำที่อยู่ในแท็กเอชทีเอ็มแอลตั้งแต่น้อยไปหามาก ตั้งแต่ไม่ให้ความสำคัญเลย จนไปถึงให้ความถี่ของคำในแท็กเอชทีเอ็มแอลเพิ่มถึง 15 เท่า ที่ระดับไหนจะให้ความถูกต้องมากกว่า

ตารางที่ 7 ความถูกต้องเมื่อเพิ่มความถี่คำในแท็กเอชทีเอ็มแอล

จำนวนครั้งที่เพิ่มคำที่อยู่ในแท็กเอชทีเอ็มแอลเข้าไปในข้อมูลที่มีการลบแท็กเอชทีเอ็มแอลแล้ว	Naïve Bayes		SVM	
	ThaiCat2	ThaiCat3	ThaiCat2	ThaiCat3
0	85.97%	91.06%	96.12%	95.04%
1	88.67%	94.74%	96.01%	96.17%

2	89.37%	94.94%	96.12%	96.17%
3	90.07%	95.17%	95.95%	95.93%
4	90.78%	95.22%	95.83%	95.98%
5	90.90%	95.41%	95.77%	95.93%
6	90.07%	95.55%	95.65%	95.89%
7	91.13%	95.60%	95.42%	95.84%
8	91.31%	95.55%	95.30%	95.89%
9	91.48%	95.41%	95.24%	95.89%
10	91.72%	95.46%	95.24%	96.07%
11	92.42%	95.46%	95.24%	96.03%
12	92.42%	95.55%	95.24%	95.98%
13	92.42%	95.65%	95.24%	95.98%
14	92.36%	95.65%	95.24%	95.98%
15	92.42%	95.60%	95.24%	95.93%

จากตารางที่ 7 จะแสดงให้เห็นชัดเจนว่า การเพิ่มคำที่อยู่ในแท็กเอชทีเอ็มแอล ไม่ว่าจะเพิ่มมากหรือเพิ่มน้อยจะให้ความถูกต้องในการแยกหมวดหมู่มากกว่าการไม่ได้เพิ่มทั้งหมดสำหรับตัวแยกแยะเบย์อย่างง่าย ยิ่งเพิ่มคำที่อยู่ในแท็กเอชทีเอ็มแอลเข้าไป ความถูกต้องจะยิ่งมากขึ้น แต่ที่ความถี่ 6 เท่าของ ThaiCat2 และที่ความถี่ 8 เท่าของ ThaiCat3 ความถูกต้องจะลดลง เหมือนกับว่าเมื่อเพิ่มความถี่ของคำที่อยู่ในแท็กเอชทีเอ็มแอลมากเกินไป จะทำให้ความสำคัญของคำธรรมดาลดลงไปมากเกิน และทำให้ผลจากการแยกหมวดหมู่มาจากคำที่อยู่ในแท็กเอชทีเอ็มแอลเท่านั้น ดังนั้นสำหรับตัวแยกแยะเบย์อย่างง่ายจึงไม่ควรจะเพิ่มความถี่ของคำที่อยู่ในแท็กเอชทีเอ็มแอลมากกว่า 5 เท่า

สำหรับเอสวีเอ็ม จะให้ความถูกต้องมากขึ้นเมื่อเพิ่มความถี่ของคำที่อยู่ในแท็กเอชทีเอ็มแอล ไม่เกิน 2 เท่า หลังจากมากกว่า 2 เท่าแล้วความถูกต้องจะลดลงเรื่อยๆ จึงแนะนำว่าสำหรับเอสวีเอ็มแล้ว ไม่ควรเพิ่มความถี่ของคำที่อยู่ในแท็กเอชทีเอ็มแอลมากกว่า 2 เท่า

4.2.7 เวลาที่ใช้ในการทำงานของเอสวีเอ็มเมื่อมีการลดมิติแล้ว

การทดลองนี้จะจับเวลาที่ใช้ในการคำนวณของเอสวีเอ็มของข้อมูล 2 ชุด โดยชุดหนึ่งคือข้อมูลที่ยังไม่มีการลดมิติ อีกชุดหนึ่งมีการใช้ Information Gain เพื่อลดมิติของคำให้เหลือคำเพียง 15,000 คำ เพื่อเปรียบเทียบเวลาที่เอสวีเอ็มใช้ในการคำนวณข้อมูลทั้ง 2 ชุด ผลการทดลองแสดงอยู่ตามตารางที่ 8

ตารางที่ 8 ผลการจับเวลาการทำงานของเอสวีเอ็มเมื่อมีการลดมิติแล้ว

ไม่ได้ใช้ Information Gain	ใช้ Information Gain	ความแตกต่าง
ครั้งที่ 1 ให้ โฟลด์ที่ 1 เป็นข้อมูลทดลอง และโฟลด์ที่ 2 และ 3 เป็นข้อมูลสอน		
	หาค่า IG 0.78 วินาที	-
เวลาสอน 16.66 วินาที	เวลาสอน 14.52 วินาที	2.14 วินาที (12.85%)
ขนาดข้อมูลที่ใช้สอน 15.23 MB	ขนาดข้อมูลที่ใช้สอน 13.41 MB	1.82 MB (11.95%)
เวลาทดสอบ 9.59 วินาที	เวลาทดสอบ 8.15 วินาที	1.44 วินาที (15.02%)
ครั้งที่ 2 ให้ โฟลด์ที่ 2 เป็นข้อมูลทดลอง และโฟลด์ที่ 1 และ 3 เป็นข้อมูลสอน		
	หาค่า IG 0.77 วินาที	-
เวลาสอน 17.69 วินาที	เวลาสอน 15.09 วินาที	2.60 วินาที (14.70%)
ขนาดข้อมูลที่ใช้สอน 15.14 MB	ขนาดข้อมูลที่ใช้สอน 13.30 MB	1.84 MB (12.15%)
เวลาทดสอบ 9.52 วินาที	เวลาทดสอบ 8.44 วินาที	1.08 วินาที (11.34%)
ครั้งที่ 3 ให้ โฟลด์ที่ 3 เป็นข้อมูลทดลอง และโฟลด์ที่ 1 และ 2 เป็นข้อมูลสอน		
	หาค่า IG 0.76 วินาที	-
เวลาสอน 17.18 วินาที	เวลาสอน 15.23 วินาที	1.95 วินาที (11.35%)
ขนาดข้อมูลที่ใช้สอน 15.29 MB	ขนาดข้อมูลที่ใช้สอน 13.58 MB	1.71 MB (11.18%)
เวลาทดสอบ 9.33 วินาที	เวลาทดสอบ 7.9 วินาที	1.43 วินาที (15.32%)
สรุป		8.33 วินาที (10.41%)
		5.37 MB (11.76%)

* การทดลองนี้ใช้ SVM^{Light} โดยใช้เคอร์เนลอาร์บีเอฟ และมีค่า Gamma เป็น 0.35

* เวลาที่ใช้ในการทดลองนี้ทั้งหมด ไม่รวมเวลาในการอ่านหรือเขียนดิสก์

ผลจากตารางที่ 8 จะแสดงให้เห็นว่าการใช้ Information Gain เข้ามาลดจำนวนคำก่อนที่จะแยกหมวดหมู่ด้วยเอสวีเอ็ม ทำให้ใช้เวลาลดลงประมาณ 10% และใช้พื้นที่เก็บข้อมูลลดลงประมาณ 10% ด้วย

4.3 **สรุปผลการทดลอง**

จากผลการทดลองตั้งแต่ ตารางที่ 6 ถึงตารางที่ 8 รูปที่ 12 และรูปที่ 13 ซึ่งเป็นผลจากการแยกหมวดหมู่ให้กับเวบเพจ ตั้งแต่การเตรียมเอกสาร กำหนดความสำคัญของคำ จนไปถึงการเลือกวิธีการแยกหมวดหมู่ พอจะสรุปเป็นข้อๆ ได้ดังนี้

1. ในขั้นตอนการเตรียมเอกสาร การใช้คำในแท็กเอชทีเอ็มแอล มาเพิ่มเข้าไปในเอกสารปกติที่มีการตัดแท็กเอชทีเอ็มแอลทิ้งแล้ว จะสามารถเพิ่มความถูกต้องในการแยกหมวดหมู่ได้ไม่ว่าจะเป็นตัวแยกแยะเบย์อย่างง่ายหรือ เอสวีเอ็ม
2. ด้วยข้อมูลลักษณะเดียวกันเอสวีเอ็มจะให้ความถูกต้องให้การแยกหมวดหมู่สูงกว่าตัวแยกแยะเบย์อย่างง่ายเสมอ
3. สำหรับการแยกหมวดหมู่ด้วยตัวแยกแยะเบย์อย่างง่ายการปรับแต่งข้อมูลก่อนที่จะเริ่มกระบวนการแยกหมวดหมู่ เช่น การเพิ่มข้อความในแท็กเอชทีเอ็มแอลไม่ว่าจะเพิ่มน้อยหรือเพิ่มมาก การลบคำทั่วไป และ การลดมิติด้วยการใช้ Information Gain ล้วนแล้วแต่เพิ่มความแม่นยำในการแยกหมวดหมู่ให้เอกสารทั้งสิ้น
4. สำหรับเอสวีเอ็ม การเพิ่มจำนวนคำ โดยการเพิ่มข้อความในแท็กเอชทีเอ็มแอล (ข้อความในแท็กเอชทีเอ็มแอล บางคำจะไม่มีในข้อมูลที่ลบแท็กเอชทีเอ็มแอลทิ้ง เช่นข้อมูลที่ได้จากแท็ก META และ แท็ก IMG) จะทำให้ความถูกต้องสูงขึ้น ขณะที่การลดจำนวนคำลงด้วยวิธีการตัดคำทั่วไป หรือการลดคำด้วยการใช้ Information Gain มีผลต่อความถูกต้องน้อยมาก เพียงแต่ช่วยให้สามารถเรียนรู้ได้เร็วขึ้นเท่านั้น
5. เอกสารภาษาอังกฤษล้วนๆ ยังให้ความถูกต้องในการแยกหมวดหมู่มากกว่าเอกสารที่มีทั้งภาษาไทย และภาษาอังกฤษ
6. จำนวนคำที่เลือกด้วยการใช้ Information Gain สูงสุด ที่ทำให้ตัวแยกแยะเบย์อย่างง่ายได้ความแม่นยำสูงสุดจะอยู่ที่ประมาณ 50% ของคำทั้งหมด ขณะที่ถ้าเป็นเอสวีเอ็มจะสามารถลดได้จนเหลือเพียง 20% ของคำทั้งหมดโดยที่ความแม่นยำยังไม่ลดลงได้
7. การเพิ่มจำนวนคำในแท็กเอชทีเอ็มแอลมากเกินไป (มากกว่า 2 เท่า) จะมีผลทำให้ความแม่นยำที่ได้จากการแยกหมวดหมู่ด้วยเอสวีเอ็มลดลง ขณะที่ตัวแยกแยะเบย์อย่างง่ายสามารถเพิ่มได้มากถึง 5 เท่าก่อนที่ความถูกต้องจะเริ่มลดลง

โดยสรุปแล้ว งานวิจัยชิ้นนี้ได้ทดลองการแยกหมวดหมู่ให้เวบเพจด้วย เอสวีเอ็ม และตัวแยกแยะเบย์อย่างง่ายโดยการใช้ข้อมูลที่ได้มีการปรับแต่งแบบต่างๆ ไม่ว่าจะกรณีใดๆ เอสวีเอ็ม ก็ให้ความแม่นยำที่สูงกว่า

บทที่ 5
สรุปผลการวิจัย และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

ขณะที่อินเตอร์เน็ตในประเทศไทยกำลังขยายตัวอย่างมาก การจะพิจารณาแต่ละหน้าเวบเพจเพื่อแยกหมวดหมู่ หรือหาหน้าเวบเพจที่ข้อมูลตรงกับหมวดหมู่ตรงกับสิ่งที่ต้องการทำได้ยากขึ้น การอ่านทีละหน้าเพื่อแยกหมวดหมู่ให้กับหน้าเวบเพจนั้นๆ ทำได้ยากและเสียเวลามากขึ้น วิธีการที่จะแยกหมวดหมู่ให้กับเวบเพจแบบอัตโนมัติ จึงเป็นเรื่องที่จำเป็น

หากแต่ว่าด้วยข้อจำกัดที่ภาษาไทย ไม่เหมือนกับภาษาอังกฤษที่มีการแบ่งแยกคำอย่างชัดเจน ทำให้ต้องมีกระบวนการตัดคำเข้ามาเกี่ยวข้องด้วย แน่นอนว่าไม่มีระบบตัดคำใดที่ให้ผลถูกต้องทั้งหมดในเวลาที่จำกัด ดังนั้นจึงมีความจำเป็นจะต้องหาวิธีการแยกหมวดหมู่ที่เหมาะสมกับเวบเพจภาษาไทย

เวบเพจต่างจากเอกสารทั่วไปตรงที่มีแท็ก และในแท็กนี้เองที่เต็มไปด้วยข้อมูลที่มีคุณภาพและมีความสำคัญ การนำประโยชน์จากการที่มีการให้ความสำคัญของคำ มาใช้ร่วมกับการแยกหมวดหมู่ด้วย ย่อมทำให้ความแม่นยำในการแยกหมวดหมู่ให้เวบเพจได้เพิ่มขึ้น

จากแนวคิดข้างต้นจึงได้มีการเตรียมข้อมูลเวบเพจภาษาไทยขึ้นมา 2 ชุด โดยชุดแรกเกี่ยวกับทางด้านวิทยาศาสตร์ ประกอบด้วย 4 หมวดหมู่ย่อย มีจำนวนเวบเพจทั้งหมด 1,703 หน้า และข้อมูลชุดที่ 2 เกี่ยวกับทางด้านนันทนาการและกีฬา ประกอบด้วย 4 หมวดหมู่ย่อย มีทั้งหมด 2,116 หน้า

จากการทดลองเมื่อนำเอกสารทั้งหมด มาแยกระหว่างข้อความธรรมดา กับข้อความที่อยู่ในแท็กเอชทีเอ็มแอล และให้ความสำคัญกับคำในแท็กเอชทีเอ็มแอลมากขึ้น โดยการเพิ่มความถี่ให้กับคำเหล่านี้ ทำให้ความถูกต้องที่ได้จากการแยกหมวดหมู่เพิ่มขึ้นทั้งสิ้น ไม่ว่าจะใช้เอสวีเอ็มหรือตัวแยกแยะเบย์อย่างง่าย

การใช้ Information Gain และคำทั่วไป เข้ามาลดมิติหรือลดจำนวนคำก่อนที่จะเริ่มแยกหมวดหมู่ให้ข้อมูลนั้น สำหรับตัวแยกแยะเบย์อย่างง่ายแล้วจะให้ผลที่ดีขึ้นในระดับหนึ่ง (ประมาณ 0.5 ถึง 1 เปอร์เซ็นต์) แต่สำหรับเอสวีเอ็มแล้วแทบจะไม่มีผลกับความถูกต้องเลย และมีผลก็เพียงแค่ทำให้เวลาที่ใช้ในการคำนวณลดลงเท่านั้น

นอกจากนั้นเอสวีเอ็ม สามารถแยกหมวดหมู่ให้เวบเพจภาษาไทยได้ดีกว่าตัวแยกแยะเบย์อย่างง่ายไม่ว่าจะเป็นข้อมูลอันไหนก็ตาม อีกทั้งยังมีความทนทานต่อข้อมูลที่ยังไม่ได้จัดรูปแบบอีกด้วย

5.2 ข้อเสนอแนะ

1. เนื่องจากในเอสวีเอ็มนั้นมีพารามิเตอร์มากมาย โดยพารามิเตอร์แต่ละตัวก็ล้วนแล้วแต่มีผลกับความแม่นยำในการแยกหมวดหมู่ทั้งสิ้น ลองปรับแต่งค่าพารามิเตอร์ต่างๆ ของเอสวีเอ็มดู เช่น เปลี่ยนเคอร์เนล หรือ เปลี่ยนค่าพารามิเตอร์ของเคอร์เนล เพื่อให้ได้ผลการแยกที่ดียิ่งขึ้น
2. ปรับปรุงการจำแนกแบบหลายหมวดหมู่ของเอสวีเอ็ม โดยใช้วิธีการ 1-v-1 หรือ เอดีเอจี [8] ซึ่งจะได้ความถูกต้องจากการแยกหมวดหมู่ที่ดีกว่า
3. เก็บข้อมูลหน้าเวบเพจเพิ่มและเพิ่มจำนวนหมวดหมู่ของเอกสาร เพื่อให้สามารถลดมิติโดยการใช้ χ^2 ได้
4. มีการเก็บคำจากเอกสารหลังจากที่มีการตัดคำแล้วด้วยวิธีการ bi-gram หรือมากกว่า ซึ่งแม้ว่าจะทำให้จำนวนคำเพิ่มขึ้น แต่ก็สามารถเพิ่มความถูกต้องให้กับการแยกหมวดหมู่ด้วยเช่นกัน
5. ใช้วิธีการแยกหมวดหมู่ 2 ชุดหรือมากกว่า มาช่วยกันจำแนกเวบเพจ นำคะแนนจากทุกวิธีการมาโหวตให้คะแนนว่าควรจะเป็นหมวดหมู่ใด

รายการอ้างอิง

[1] Ampornaramveth, V. Thai Software. http://thaigate.nacsis.ac.jp/files/index.html [2001, Feb 3].

[2] Filo, D. and Yang, J., 1997, Yahoo! Inc., http://www.yahoo.com/docs/pr/. [2001, Mar 11].

[3] He, J., Tan, A., and Tan, C. 2000. A Comparative Study on Chinese Text Categorization Methods. PRICAI Workshop on Text and Web Mining 2000, 24-35.

[4] Infoseek, Infoseek Content Classification Engine, http://software.infoseek.com/products/cce. [2001, Mar 5].

[5] Joachims, T. 1998. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. Proceedings of ECML-98, 10th European Conference on Machine Learning, 137-142.

[6] Koanantakool, T., and Agsorn-intara, A. Character codes and Input/Output method for the Thai Language, http://thaigate.nacsis.ac.jp/refer/thaiconf/index.html. [2001, Mar 3].

[7] Karoonboonyanan, T. An Implementation of Double-Array Trie, http://www.links.nectec.or.th/~thep/datrie/. [2002, Apr 24].

[8] Kijsirikul, B., and Ussivakul, N. 2002. Multiclass Support Vector Machines Using Adaptive Directed Acyclic Graph. The IEEE/INNS International Joint Conference on Neural Networks. Hawaii.

[9] Kosavisutte K. Basic Concept of Thai Language, http://zzzthai.fedu.uec.ac.jp/thailang/. [2001, Feb 10].

[10] McCallum, A. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/~mccallum/bow. 1996.

[11] Mladenie, D. 1998. Turning Yahoo into an Automatic Web-Page Classifier. European Conference on Artificial Intelligence, 473-474.

[12] Nectec, Orchid Corpus, http://www.links.nectec.or.th/orchid/. [2001, Apr 5].

[13] Raggett, D. Dave Raggett's Introduction to HTML, http://www.w3.org/MarkUp/Guide/. [2001, Mar 1].

[14] Schölkopf, B. 1997. Support Vector Learning. Munich: R. Oldenbourg Verlag Publications.

[15] Sebastiani, D. 1999. Machine Learning in Automated Text Categorisation. Machine learning in automated text categorisation: a survey

[16] SiamGURU. http://www.siamguru.com/. [2001, Apr 23].

[17] Stopwords List, http://rdsweb2.rdsinc.com/help/stopword_list.html. [2001, Apr 5].

[18] Vapnik, V. 1998. Statistical Learning Theory. New York: Wiley.

[19] W3C. HTML 4.01 Specification. http://www.w3.org/TR/html4/. [2002, Mar 1].

[20] Yang, Y., and Pedersen, J. 1997. A Comparative Study on Feature Selection in Text Categorization. Proceedings of ICML-97, 14th International Conference on Machine Learning. 412-420.

ประวัติผู้เขียนวิทยานิพนธ์

นาย อดุลย์ ตันธุวนิตย์ เกิดเมื่อวันที่ 21 ตุลาคม พ.ศ.2520 ที่กรุงเทพมหานคร สำเร็จการศึกษาหลักสูตรวิศวกรรมศาสตรบัณฑิต (วศ.บ.) สาขาวิชาวิศวกรรมไฟฟ้า จากภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหิดล เมื่อปีการศึกษา 2541 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต (วท.ม.) สาขาวิชาวิทยาศาสตรคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ ที่จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2542