การพัฒนาประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทยแบบอัตโนมัติ

นิเวศ จิระวิชิตชัย ปริญญา สงวนสัตย์** พยุง มีสัจ****

บทคัดย่อ

บทความนี้ได้นำเสนอแบบจำลองการจัดหมวดหมู่เอกสารภาษาไทย โดยลดคุณลักษณะของเอกสารก่อนประมวลผลด้วยเครื่องจักรการเรียนรู้ เพื่อประโยชน์ในการลดมิติของข้อมูล ลดระยะเวลาประมวลผล ประหยัดทรัพยากรของระบบ และเพิ่มประสิทธิภาพในการจัดหมวดหมู่เอกสารภาษาไทย จากการทดลอง พบว่า การลดคุณลักษณะด้วยวิธี Information Gain และประมวลผลด้วยเครื่องจักรการเรียนรู้แบบต่าง ๆ บนแบบจำลองการจัดหมวดหมู่เอกสารภาษาไทย โดยวัดประสิทธิภาพการจัดหมวดหมู่เอกสารที่ F-Measure ดีที่สุด พบว่า อัลกอริทึม SVM ให้ประสิทธิภาพสูงสุด คือ 94.3% รองลงมาเป็นอัลกอริทึม Naïve Baye 86.2% อัลกอริทึม RBF 86.1% อัลกอริทึม J48 79.7% อัลกอริทึม Ripper 78.9% อัลกอริทึม KNN 69.5% ตามลำดับ เมื่อพิจารณาด้านการลดขนาดคุณลักษณะจากกลุ่มตัวอย่างของอัลกอริทึม SVM พบว่า สามารถลดลงได้มากถึง 90% โดยการลดลงของคุณลักษณะดังกล่าวไม่ส่งผลให้ประสิทธิภาพในการจัดหมวดหมู่เอกสารลดลงแต่อย่างใด

คำสำคัญ: *การจัดหมวดหมู่เอกสาร การลดคุณลักษณะ เครื่องจักรเรียนรู้*

---

* อาจารย์ บัณฑิตวิทยาลัย มหาวิทยาลัยศรีปทุม วิทยาเขตชลบุรี
เลขที่ 79 ถนนบางนา-ตราด ตำบลคลองตำหรุ อำเภอเมือง จังหวัดชลบุรี 20000

** ผู้ช่วยศาสตราจารย์ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยรังสิต
เลขที่ 52/347 หมู่บ้านเมืองเอก ถนนพหลโยธิน ตำบลหลักหก อำเภอเมือง จังหวัดปทุมธานี 12000

*** ผู้ช่วยศาสตราจารย์ ภาควิชาครุศาสตร์ไฟฟ้า คณะครุศาสตร์อุตสาหกรรม
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
เลขที่ 1518 ถนนพิบูลสงคราม แขวงวงศ์สว่าง เขตบางซื่อ กรุงเทพมหานคร 10800

Developing and Effective Automatic Thai Document Categorization

Nivet Chirawichitchai Parinya Sanguansat** Phayung Meesad****

Abstract

This research presented thai document categorization framework. The reduction feature of the document before processing by machine learning. To reduce the time and resources in the system of processing and increase efficiency of thai document categorization.

The experimental results showed that reducing the feature by Information Gain method and process with different learning machines on thai document categorization framework. When testing the performance of categorization documents found SVM algorithms is most powerful 94.3%. Followed by the Naïve Baye 86.2%, RBF 86.1%, J48 79.7%, Ripper 78.9%, KNN 69.5% algorithms respectively. The experimental results reducing the size of the sample found to decrease by up to 90%. The reduction of the feature does not affect the performance of document categorization decreased.

Keywords: *Text Categorization, Dimension Reduction, Machine Learning*

* Lecturer, Graduate School, Sripatum University Chonburi Campus
79 Bangna-Trad Road, Muang District, Chonburi 20000, THAILAND

** Assistant Professor, Faculty of Information Technology, Rangsit University
52/347 Paholyothin Road, Muang District, Pathumtani 12000, THAILAND

*** Assistant Professor, Department of Teacher Training in Electrical Engineering
Faculty of Technical Education
King Mongkut's University of Technology North Bangkok
1518 Pibulsongkram Road, Bangsue District, Bangkok 10800, THAILAND

ความเป็นมาและความสำคัญของปัญหา

การขยายตัวทางการใช้งานระบบคอมพิวเตอร์และอินเทอร์เน็ต ตลอดช่วงระยะเวลาที่ผ่านมาจนถึงปัจจุบันมีแนวโน้มในการใช้งานเพิ่มมากขึ้นอย่างรวดเร็ว ส่งผลให้เกิดการสร้างและเก็บข้อมูลหลายชนิดในรูปแบบอิเล็กทรอนิกส์ ซึ่งหนึ่งในข้อมูลอิเล็กทรอนิกส์เหล่านี้ คือ ข้อมูลประเภทเอกสาร เช่น จดหมายอิเล็กทรอนิกส์ (E-mail) เว็บเพจ (Web page) เอกสารข่าว (News) ไฟล์งานเอกสารต่าง ๆ (Document) ซึ่งเป็นข้อมูลที่มีปริมาณและเนื้อหาที่หลากหลายมากขึ้น ทำให้ยากต่อการค้นหาและจัดเก็บหมวดหมู่เอกสาร ดังนั้น การสืบค้นและการจัดการเอกสารจะง่ายและเป็นไปตามความต้องการ ต้องอาศัยการจัดแบ่งเอกสารเป็นกลุ่มหรือหมวดหมู่ให้สอดคล้องและตรงกับดัชนีเพื่อให้จัดเก็บและค้นคืนเอกสารได้อย่างรวดเร็วและมีประสิทธิภาพ จึงมีความจำเป็นต้องอาศัยผู้เชี่ยวชาญในการจัดกลุ่มข้อมูล ฉะนั้น จึงเป็นการยากในการที่จะจัดกลุ่มหรือแยกประเภทเอกสาร ยิ่งหากเอกสารมีปริมาณมากขึ้นทุก ๆ วัน ซึ่งต้องพึ่งพาทรัพยากรบุคคลในการจัดหมวดหมู่เอกสารเหล่านี้มากตามไปด้วยเช่นกัน ทำให้มีการคิดค้นพัฒนากระบวนการในการจัดหมวดข้อมูลที่มีขนาดใหญ่เหล่านี้ให้เป็นไปแบบอัตโนมัติ เพื่อที่จะสามารถจำแนกกลุ่มข้อมูลเพื่อใช้ประโยชน์จากข้อมูลและการจัดการกับข้อมูลให้มีประสิทธิภาพ รองรับการสืบค้นจากผู้ใช้งานเอกสารอย่างถูกต้องและเหมาะสม (Sebastiani, 2002)

ปัจจุบันได้มีการศึกษาเกี่ยวกับการนำวิธีการเรียนรู้ด้วยคอมพิวเตอร์ มาประยุกต์ร่วมกับการประมวลผลภาษาธรรมชาติเพื่อใช้จัดแบ่งกลุ่มเอกสารนั้น สามารถแบ่งได้ 2 ลักษณะ คือ การจัดกลุ่ม (Clustering) และการจำแนกหมวดหมู่ (Classification หรือ Categorization) การจัดกลุ่มเอกสาร คือ การแบ่งกลุ่มตามเนื้อหาของเอกสารโดยไม่มีการกำหนดกลุ่มหรือหมวดหมู่ของเอกสารไว้ก่อน ซึ่งจะเป็นการแบ่งกลุ่มตามลักษณะของเอกสาร โดยเอกสารที่มีลักษณะเหมือนกันจะอยู่ด้วยกัน ส่วนการจำแนกหมวดหมู่เอกสาร คือ การแบ่งกลุ่มตามเนื้อหาของเอกสาร โดยที่มีการกำหนดกลุ่มหรือหมวดหมู่ของเอกสารไว้ก่อน โดยจะเปรียบเทียบเอกสารกับต้นแบบในแต่ละหมวดหมู่เอกสารจะถูกจัดอยู่ในหมวดหมู่ที่ต้นแบบมีลักษณะคล้ายกับตัวมันเองมากที่สุด โดยผลลัพธ์ที่ได้จากวิธีการเรียนรู้ด้วยคอมพิวเตอร์นั้น ความถูกต้องใกล้เคียงกับผลการจำแนกหมวดหมู่ของเอกสารที่ทำโดยมนุษย์ ทำให้ประหยัดแรงงานมนุษย์เป็นอย่างมากเพราะไม่ต้องอาศัยผู้เชี่ยวชาญในการจำแนกประเภทเอกสารหรือปรับเปลี่ยนหมวดหมู่ของเอกสาร (Sebastiani, 2002; Marquez, 2000)

เนื่องจากงานด้านการจัดหมวดหมู่เอกสารส่วนใหญ่ในปัจจุบัน มุ่งเน้นไปในการจัดหมวดหมู่เอกสารภาษาอังกฤษ ยังขาดงานการพัฒนาด้านการจัดหมวดหมู่เอกสารภาษาไทย จากความสำคัญของปัญหาดังกล่าว ผู้วิจัยจึงมีแนวคิดที่จะพัฒนาประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทย โดยนำเสนอแบบจำลองการจัดหมวดหมู่ของเอกสารภาษาไทยแบบอัตโนมัติ (นิเวศ

จิระวิชิตชัย, ปริญญา สงวนสัตย์ และพยุง มีสัจ, 2553) โดยทดสอบประสิทธิภาพแบบจำลองการจัดหมวดหมู่เอกสารภาษาไทยกับ อัลกอริทึมต้นไม้ตัดสินใจ (Decision Tree) ซัพพอร์ทเวก-เตอร์แมชชีน (Support Vector Machine) เนอีฟเบย์ (Naïve-Bayes) เครือข่ายฟังก์ชันฐานรัศมี (Radial Basis Function Network) เคเนียเรสเนเบอร์ (K-Nearest Neighbor) และกฎริปเปอร์ (Ripper Rules) ในบทความนี้ ประกอบด้วย ส่วนต่าง ๆ ดังนี้ ส่วนที่ 2 กล่าวถึงทฤษฎีที่เกี่ยวข้อง ส่วนที่ 3 กล่าวถึงวิธีการดำเนินงานและแบบจำลองด้านการจัดหมวดหมู่เอกสารภาษาไทย ส่วนที่ 4 ผลการทดลอง และส่วนที่ 5 สรุปผลและข้อเสนอแนะ

ทฤษฎีที่เกี่ยวข้อง

การตัดคำ (Word Segmentation)

การตัดคำ (Word Segmentation) การประมวลผลจำแนกหมวดหมู่เอกสารภาษาไทยได้อย่างมีประสิทธิภาพนั้น มีปัญหาเบื้องต้น คือ การตัดคำในภาษาไทย ซึ่งลักษณะการเขียนภาษาไทยจะมีการเขียนติดต่อกันเป็นสายอักขระโดยไม่มีเครื่องหมายวรรคตอนแสดงการแบ่งคำ ดังเช่นภาษาอังกฤษซึ่งใช้ช่องว่าง (Space) คั่นระหว่างคำ ซึ่งเป็นอุปสรรคอย่างหนึ่งเพื่อแบ่งสายอักขระไทยออกเป็นคำ ๆ จึงได้มีผู้คิดค้นพัฒนาวิธีการตัดคำแบ่งได้เป็น หลักการตัดคำโดยใช้กฎ (Rule Base Approach) หลักการตัดคำโดยใช้อัลกอริทึม (Algorithm Approach) หลักการตัดคำโดยใช้พจนานุกรม (Dictionary Approach) และหลักการตัดคำโดยใช้คลังข้อมูล (CorpusBase Approach) แต่ละวิธีการต่าง ๆ ก็ให้ผลในด้านความถูกต้อง ความรวดเร็วของการทำงานและปริมาณการใช้ทรัพยากรต่าง ๆ ที่แตกต่างกัน จากการศึกษาเรื่องตัดคำสำหรับการจัดหมวดหมู่เอกสารภาษาไทย พบปัญหาด้านการหาขอบเขตของคำ เนื่องจากไม่มีการเขียนแบ่งพยางค์ คำ หรือประโยค ไม่มีหลักเกณฑ์ตายตัวในการใช้ช่องว่างในภาษาเขียน การสะกดคำมีรูปแบบซับซ้อน มีคำยืม คำทับศัพท์ คำเฉพาะจำนวนมาก และคำมีความกำกวมสูง จากการศึกษาเปรียบเทียบประสิทธิภาพวิธีดังกล่าว พบว่า วิธีตัดคำที่เหมาะสมกับการจัดหมวดหมู่เอกสาร คือ วิธีการตัดคำแบบยาวที่สุด (Longest Matching) ซึ่งมีวิธีการตรวจสอบสายอักขระ (String) ที่เข้ามาจากซ้ายไปขวากับพยางค์ที่เก็บไว้ในพจนานุกรม ในกรณีที่ตรวจสอบแล้วปรากฏว่าพบพยางค์มากกว่า 1 พยางค์ในพจนานุกรม ก็ให้เลือกแบ่งพยางค์โดยเลือกพยางค์ที่ยาวที่สุด แล้วทำต่อไปเรื่อย ๆ จนจบสายอักขระ แต่ถ้ากรณีที่เลือกพยางค์ที่ยาวที่สุดแล้วทำให้เกิดพยางค์ที่ไม่ปรากฏในพจนานุกรม ก็ยอมให้มีการย้อนรอย (Back Tracking) กลับไปเลือกพยางค์ที่ยาวรองมาแทนทำต่อไปเรื่อย ๆ จนสิ้นสุดสายอักขระ (Charoenpornsawat, 1999)

การกำจัดคำหยุด (Stop-Word List Removal)

เป็นการนำคำที่ไม่มีนัยสำคัญออกโดยที่ไม่ทำให้ความหมายของเอกสารเปลี่ยนแปลงคำ ที่ไม่มีนัยสำคัญ ในที่นี้หมายถึงคำที่ใช้กันโดยทั่วไปไม่มีความหมายสำคัญต่อเอกสาร เมื่อตัดออกจากเอกสารแล้วไม่ทำให้ใจความของเอกสารเปลี่ยนแปลง ตัวอย่างเช่น คำบุพบทเป็นคำที่ใช้เชื่อมคำหรือกลุ่มคำให้สัมพันธ์กัน คำสันธานเป็นคำที่ทำหน้าที่เชื่อมคำกับคำ คำสรรพนามเป็นคำที่ใช้แทนคำนามที่กล่าวถึงมาแล้วในประโยค เป็นต้น คำหยุดมักเป็นคำที่ปรากฏขึ้นบ่อยครั้งในเอกสาร และปรากฏในเอกสารเกือบทุกฉบับ จึงถือได้ว่าคำหยุดเป็นคุณลักษณะที่ไม่เกี่ยวข้องหรือไม่มีประโยชน์ในการค้นคืนหรือการจำแนกหมวดหมู่ ดังนั้น การกำจัดคำหยุดจึงเป็นกระบวนการที่ควรทำก่อนการจัดทำดัชนี เพื่อกำจัดคุณลักษณะที่ไม่เป็นประโยชน์และลดขนาดของดัชนีลง ซึ่งจะช่วยประหยัดทั้งพื้นที่และเวลาในการประมวลผล ตัวอย่างคำหยุด เช่น ที่ ใน ว่า และ จะ มี ได้ ของ ให้ เป็นต้น (Jaruskulchai, 1998)

การหารากศัพท์ (Stemming)

จึงเป็นการหารูปเดิมของคำ หรือหาคำที่มีความหมายคล้ายกัน เพื่อปรับรวมให้เป็นคำเดียวกัน การหารากศัพท์เป็นกระบวนการที่ควรทำก่อนการจัดทำดัชนี ทำให้สามารถลดขนาดของดัชนีลงและเพิ่มประสิทธิภาพในการค้นคืนหรือการจำแนกหมวดหมู่ การหารากศัพท์ของคำภาษาไทยนั้นจะใช้วิธีการรวบรวมคำศัพท์ที่มีความหมายคล้ายกัน หรือมีรากศัพท์เดียวกันไว้เป็นรายการคำศัพท์ หรือจัดเก็บในคลังคำ เพื่อใช้ในการเปรียบเทียบหารากศัพท์ ซึ่งวิธีการนี้ต้องอาศัยมนุษย์เป็นผู้กำหนดไว้ก่อนว่า คำแต่ละคำมีรากศัพท์ เป็นคำใด เช่น กิน ทาน รับประทาน เสวย หรือ พิจาร พิจารณ์ วิจารณ์ วิธีการนี้ต้องอาศัยผู้เชี่ยวชาญทางภาษาและใช้เวลาในการเก็บรวบรวมและจัดทำรายการคำศัพท์ (Jaruskulchai, 1998; ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ)

การสกัดคุณลักษณะ (Feature Extraction)

วัตถุประสงค์ของขั้นตอนการสกัดคุณลักษณะเอกสาร คือ การดึงคุณลักษณะ (Feature) ของเอกสารออกมา กับการลดขนาดเอกสารลง ซึ่งการดึงคุณลักษณะออกมานั้น ต้องกำหนดก่อนว่าจะใช้อะไร เป็นตัวแทนคุณลักษณะของเอกสาร และใช้ค่าใดแทนคุณลักษณะเอกสารนั้น จากการสำรวจบทความที่ผ่านมาทั้งในประเทศและต่างประเทศ พบว่า ส่วนใหญ่จะใช้คำเป็นตัวแทนคุณลักษณะของเอกสาร และใช้พื้นฐานค่าความถี่ของคำเป็นค่าของคุณลักษณะ นอกจากการใช้คำเดี่ยวแล้ว ยังสามารถใช้ วลี หรือกลุ่มของคำ ประโยค แทนคุณลักษณะของเอกสารได้เช่นกัน ตัวแทนคุณลักษณะของเอกสารที่นิยมใช้ในการจัดหมวดหมู่เอกสารประเภทข้อความ คือ ถุงคำ (Bag of words) ซึ่งจะเก็บอยู่ในรูปแบบของเวกเตอร์ โดยองค์ประกอบของเวกเตอร์อาจจะแทน

ด้วยคุณลักษณะของค่าความจริง (Boolean) แทนด้วยค่าความถี่ของคำ (Word Frequency) หรือแทนด้วยค่าน้ำหนักของคำแบบอื่น ๆ (วัลลภ อินทร์ฉ่ำ, 2548; Aas, 1999) ซึ่งบทความนี้ใช้การเลือกคุณลักษณะแบบคำเดี่ยว (Single word) ซึ่งได้จากการตัดคำโดยใช้พจนานุกรมเรียบร้อยแล้ว ผลลัพธ์ที่ได้จากการตัดคำจะได้เป็นคำเดียวจำนวนมาก เพื่อมาใช้เป็นตัวแทนเอกสารในการเรียนรู้

การสร้างดัชนี (indexing)

เนื่องจากคอมพิวเตอร์ไม่สามารถจำแนกหมวดหมู่ของเอกสารซึ่งเป็นภาษาธรรมชาติโดยตรงได้ ดังนั้น จึงต้องแปลงเอกสารให้อยู่ในรูปแบบที่คอมพิวเตอร์สามารถใช้ในการเรียนรู้ได้ ขั้นตอนในการแปลงเอกสาร เรียกว่า การทำดัชนี (Indexing) เพื่อสร้างตัวแทนเนื้อหาของเอกสาร (Document Representation) สำหรับใช้ในกระบวนการเรียนรู้ วัตถุประสงค์ของการสร้างดัชนี คือ การคำนวณหาค่าที่จะมาใช้เป็นค่าคุณลักษณะของเอกสาร หรืออาจจะเรียกได้ว่าการหาค่าน้ำหนัก (Term Weighting) การสร้างดัชนี โดยทั่วไปที่นิยมใช้กัน จะเริ่มจากการสร้างเวกเตอร์ตัวแทนเอกสาร จากนั้นจะสร้างเมตริกซ์ของกลุ่มเอกสารขึ้นจากเวกเตอร์เอกสารทั้งหมดในกลุ่ม (วัลลภ อินทร์ฉ่ำ, 2548; Aas, 1999) ซึ่งบทความนี้ใช้วิธีความถี่ของคำที่ปรากฏในเอกสารที่ผ่านจากการตัดคำมาเป็นค่าน้ำหนัก ถ้าคำใดที่ผ่านการตัดคำมีปริมาณมาก ก็จะมีค่าความถี่มาก ซึ่งจะส่งผลให้ได้ค่าน้ำหนักที่มีค่าสูงมากตาม เมื่อถึงขั้นตอนนี้จะได้รูปแบบที่มีลักษณะของการแสดงความสัมพันธ์ระหว่างคำ (Words: w) และเอกสารสารทั้งหมด (Documents: d) ด้วยเวกเตอร์ 2 มิติ ซึ่งคำที่ได้นั้นต้องผ่านทำดัชนีและตัดคำหยุด (Stop-words) ออกไป และเอกสารทั้งหมดรูปแบบ Vector Space Model หรือบางครั้งเรียกรูปแบบนี้ว่า Bag of Words โดยสามารถแสดงได้ดังภาพ

	w_1	w_2	...	w_k	...	w_v
d_1	w_{11}	w_{12}	...	w_{1k}	...	w_{1v}
d_2	w_{21}	w_{22}	...	w_{2k}	...	w_{2v}
...	...	...	...	...	...	
d_N	w_{N1}	w_{N2}	...	w_{Nk}	...	w_{Nv}

การเลือกคุณลักษณะ (Feature Selection)

จากที่กล่าวมาจะพบว่าเอกสารนั้นมีแนวโน้มที่จะเพิ่มปริมาณสูงขึ้นทุกวัน ทำให้เอกสารมีจำนวนคุณลักษณะมากขึ้น ทำให้วิธีเบื้องต้นในการลดขนาดเอกสาร คือ การใช้การนำคำที่ไม่มีนัยสำคัญออกกับการทำรากศัพท์แล้วยังไม่เพียงพอ ซึ่งจำนวนคุณลักษณะมีผลต่อประสิทธิภาพของ

การจำแนกหมวดหมู่เอกสาร เนื่องจากอัลกอริทึมที่ใช้ในการเรียนรู้เพื่อสร้างตัวจำแนกหมวดหมู่โดยทั่วไปไม่สามารถรองรับการทำงานกับจำนวนคุณลักษณะของเอกสารที่สูงมากได้ดี การลดขนาดเอกสารจึงเป็นขั้นตอนหนึ่งที่จะต้องทำก่อน บทความนี้ใช้ค่าการเพิ่มของข้อมูล (IG: Information Gain) เป็นตัวลดคุณลักษณะของเอกสาร ซึ่งค่า IG จะคำนวณจากจำนวนบิตที่ได้รับสำหรับการทำนายกลุ่มโดยการดูจากการมีอยู่หรือไม่มีอยู่ของคำในเอกสาร ให้ $C_i, \ldots, C_K$ แทนเซตที่เป็นไปได้ของกลุ่ม ค่า IG ของคำ w นิยามโดย (Yang, 1997; วัลลภ อินทร์ฉ่ำ, 2548)

$$IG(w) = -\sum_{j=1}^{K} P(c_j)\log P(c_j) + P(w)\sum_{j=1}^{K} P(c_j \mid w)\log P(c_j \mid w) + P(\overline{w})\sum_{j=1}^{K} P(c_j \mid \overline{w})\log P(c_j \mid \overline{w})$$

ค่า $P(C_j)$ คำนวณได้จาก เศษส่วนของจำนวนเอกสารที่อยู่กลุ่ม C_j กับจำนวนเอกสารทั้งหมด

ค่า $P(w)$ คำนวณได้จาก เศษส่วนของจำนวนเอกสารที่มีคำ w กับจำนวนเอกสารทั้งหมด

ค่า $P(C_j \mid w)$ คำนวณได้จาก เศษส่วนของคำจำนวนเอกกลุ่ม C_j ที่มีคำ w กับเอกสารทั้งหมด

ค่า $P(C_j \mid \overline{w})$ คำนวณได้จาก เศษส่วนของคำจำนวนเอกกลุ่ม C_j ที่ไม่มีคำ w กับเอกสารทั้งหมด

เมื่อทำการคำนวณค่า IG ของแต่ละคุณลักษณะได้ จากทำการจัดเรียงคุณลักษณะที่มีค่า IG มากไปหาน้อยและทำการตัดคุณลักษณะที่มีค่าต่ำกว่าเกณฑ์ทิ้งไป ซึ่งบทความนี้ใช้ค่าคุณลักษณะเด่นที่มีค่าสูงสุด 3000 ตัวแรก เหตุผลที่ใช้จำนวนดังกล่าวเพราะเป็นจำนวนที่เหมาะสมไม่เปลืองเวลาในการประมวลผล และยังคงความแม่นยำในการแยกแยะเอกสาร (นิเวศ จิระวิชิตชัย, ปริญญา สงวนสัตย์ และพยุง มีสัจ, 2553)

อัลกอริทึมการจัดหมวดหมู่ (Classifier Algorithm)

อัลกอริทึมในการจัดหมวดหมู่การเรียนรู้แบบมีผลเฉลย (Supervised Learning) สามารถแบ่งขั้นตอนวิธีการจัดหมวดหมู่เอกสารแบ่งได้เป็น 2 ขั้นตอน คือ การเรียนรู้เพื่อสร้างกลุ่มเอกสารต้นแบบและแยกหมวดหมู่ของเอกสารที่สนใจ โดยการตรวจสอบหาความคล้ายกับกลุ่มเอกสารต้นแบบ

ต้นไม้ตัดสินใจ (Decision Tree) (Quinlan, 1993; พรพล ธรรมรงค์รัตน์, ลัดดา ปรีชา วีรกุล และวิภาดา เวทย์ประสิทธิ์, 2008) ต้นไม้จะประกอบด้วย โหนดแทนคุณลักษณะ และโหนดล่างสุดแทนหมวดหมู่ การสร้างกิ่งสาขาจะพิจารณาจากค่าความจริงของคุณลักษณะ โดยค่าที่ใช้จะมาจากการคำนวณจากค่า Information Gain การสร้างต้นไม้ตัดสินใจ C4.5 ใช้ค่ามาตรฐานอัตราส่วนเกน (Gain Ratio) เพื่อเลือกคุณลักษณะที่จะใช้เป็นรากหรือโหนด ถ้าให้ชุดข้อมูล M ประกอบด้วย ค่าที่เป็นไปได้ คือ $\{m_1, m_2, \ldots, m_n\}$ และให้ความน่าจะเป็นที่จะเกิดค่า m_1 มีค่า

เท่ากับ $P(m_i)$ จะได้ว่าค่าเกนสารสนเทศ (Information Gain) ของ M เขียนแทนด้วย I(M) คำนวณได้ ดังสมการ

$$I(M) = \sum_{i=1}^{n} - P(m_i)\log_2 P(m_i)$$

ถ้าให้ข้อมูลสอน คือ T และคุณลักษณะที่เป็นโหนด คือ x และมีค่าทั้งหมดที่เป็นไปได้ n ค่าโหนดปัจจุบันจะแบ่งตัวอย่าง T ออกตามกิ่งเป็น $\{t_1, t_2, \ldots, t_n\}$ ตามค่าที่เป็นไปได้ของ x ดังนั้น จึงสามารถคำนวณค่าเกนสารสนเทศหลังจากแบ่งตามคุณลักษณะ x ได้ดังสมการ

$$I_x(T) = \sum_{i=1}^{n} \frac{|t_i|}{|T|} I(t_1)$$

ค่ามาตรฐานเกน (GAIN) ของคุณลักษณะ x ได้ดังสมการ

$$Gain(x) - I(T) - I_x(T)$$

จากนั้นคำนวณค่าสารสนเทศของการแบ่งแยก (Split Information) ของคุณลักษณะแต่ละตัว ถ้าให้ T คือ ชุดของตัวอย่าง เมื่อแบ่งตัวอย่างนี้ตามคุณลักษณะ x จะได้ชุดของตัวอย่างย่อยในแต่ละกิ่ง คือ $\{t_1, t_2, \ldots, t_n\}$ จำนวน n ชุด ตามค่าที่เป็นไปได้ในคุณสมบัติ x เมื่อคำนวณค่าสารสนเทศของการแบ่งแยกได้ดังสมการ

$$\text{Split Information} = \sum_{i=1}^{n} \frac{|t_i|}{|T|} \log_2 \frac{|t_i|}{|T|}$$

คำนวนค่ามาตรฐานอัตราส่วนเกน (Gain ratio) ได้จาก Gain Ratio = Gain – Split Information ท้ายสุดจึงเลือกค่า Gain ratio สูงสุดเป็นคุณลักษณะเริ่มต้น และเลือกคุณสมบัติถัดไปตามค่า Gain ratio น้อยลงตามลำดับ ในบทความนี้ใช้อัลกอริทึมต้นไม้ตัดสินใจ C 4.5 แบบมาตรฐานในการจำแนกเอกสาร

ซัพพอร์ทเวกเตอร์แมชชีน (Support Vector Machine) (Cortes & Vapnik, 1995; พรพล ธรรมรงค์รัตน์, ลัดดา ปรีชาวีรกุล และวิภาดา เวทย์ประสิทธิ์, 2008) แนวคิดหลักของวิธีการนี้ใช้เพื่อหาระนาบการตัดสินใจในการแบ่งข้อมูลออกเป็นสองส่วน โดยใช้สมการเส้นตรงเพื่อแบ่งเขตข้อมูล 2 กลุ่มออกจากกัน โดยจะพยายามสร้างเส้นแบ่งตรงกึ่งกลางระหว่างกลุ่มให้มีระยะห่างระหว่างขอบเขตของทั้งสองกลุ่มมากที่สุด SVM จะใช้ฟังก์ชันแม็ปสำหรับย้ายข้อมูลจาก Input Space ไปยัง Feature Space และสร้างฟังก์ชันวัดความคล้ายที่เรียกว่าเคอร์เนลฟังก์ชัน (Kernel Function) บน Feature Space เหมาะใช้สำหรับข้อมูลที่มีมิติของข้อมูลสูง กำหนดให้

$(x_i,y_i),\ldots,(x_n,y_n)$ เป็นตัวอย่างที่ใช้สำหรับการสอน n คือ จำนวนข้อมูลตัวอย่าง m คือ จำนวนมิติข้อมูลเข้า และ y คือ ผลลัพธ์มีค่า +1 หรือ -1 ดังสมการ

$$(x_i,y_i),\ldots,(x_n,y_n) \text{ เมื่อ } x \in R^m,\ y \in \{+1,-1\}$$

สำหรับปัญหาเชิงเส้น มิติข้อมูลขนาดสูงได้ถูกแบ่งเป็น 2 กลุ่ม โดยระนาบตัดสินใจ ซึ่งคำนวณได้ดังสมการ

$$(w*x)+b = 0$$

เมื่อ w คือ ค่าน้ำหนัก และ b คือ ค่า bias สมการ ใช้สำหรับจำแนกประเภทของข้อมูล

$$(w*x)+b>0 \text{ ถ้า } y_i = +1 \text{ และ } (w*x)+b<0 \text{ ถ้า } y_i = -1$$

อย่างไรก็ตาม SVM มีเคอร์เนลฟังก์ชัน (Kernel Function) ที่ผู้สามารถประยุกต์ใช้ในการแก้ปัญหาได้หลายวิธี สำหรับบทความนี้ ได้เลือก Linear kernel เป็นฟังก์ชันที่ใช้ในการทดลอง โดยตั้งค่าพารามิเตอร์ (Parameter) C = 1

เนอีฟเบย์ (Naïve-Bayes) (Lewis, 1998) หลักการของวิธีการนี้ใช้การคำนวณความน่าจะเป็นซึ่งถูกใช้ในการทำนายผล เป็นเทคนิคในการแก้ปัญหาที่สามารถคาดการณ์ผลลัพธ์ได้และสามารถอธิบายได้ มันจะทำการวิเคราะห์ความสัมพันธ์ระหว่างตัวแปรเพื่อใช้ในการสร้างเงื่อนไขความน่าจะเป็นสำหรับแต่ละความสัมพันธ์ การเรียนรู้เบย์อย่างง่าย เป็นวิธีจำแนกประเภทข้อมูลที่มีประสิทธิภาพวิธีหนึ่ง โดยที่ใช้ในงานจัดหมวดหมู่เอกสาร ข้อความ (Text Classification) ได้ดี อัลกอริทึมในการทำงานที่ไม่ซับซ้อน เหมาะกับกรณีของเซตตัวอย่างมีจำนวนมากและคุณสมบัติ (Attribute) ของตัวอย่างไม่ขึ้นต่อกัน โดยกำหนดให้ความน่าจะเป็นของข้อมูลที่จะเป็น

$$P(a_1, a_2,...,a_n \mid v_j) = \prod_{i=1}^{n} P(a_i \mid v_j)$$

กลุ่ม v_j สำหรับข้อมูลที่มีคุณสมบัติ n ตัว X={ a_1, a_2, …, a_n } หรือ ใช้สัญลักษณ์ว่า $P(a_1, a_2, \ldots, a_n \mid v_j)$ โดยที่ $\prod$ หมายถึง ผลคูณของค่า $P(a_i \mid v_j)$ ทั้งหมด i = 1, 2, 3, …, n และ j = 1, 2, 3,…, n ดังนั้น เราจะได้ว่าวิธีการจำแนกประเภทแบบเบย์อย่างง่าย ดังสมการที่

$$v_{NB} = \underset{v_j \epsilon V}{\operatorname{argmax}} P(v_j) \times \prod_{i=1}^{n} P(a_i \mid v_j)$$

สำหรับบทความนี้ ได้เลือกเนอีฟเบย์ (Normal Naïve-Bayes) แบบปกติ ไม่มีการปรับแต่งพารามิเตอร์แต่อย่างใด

เคเนียเรสเนเบอร์ (K-Nearest Neighbor) (Aha, Kibler & Albert, 1991) หลักการของวิธีการนี้ จะจำแนกประเภทข้อมูลโดยขึ้นกับข้อมูลที่มีคุณสมบัติใกล้เคียงที่สุด K ตัวจากข้อมูลบนชุดข้อมูลตัวอย่าง จะคำนวณหาเพื่อนบ้านที่ใกล้ที่สุด K ตัว หลังจากนั้นเราจะรวบรวมสมาชิกที่ใกล้เคียงที่สุด K ตัวแล้วเลือกคลาสที่สมาชิกส่วนใหญ่ที่ในกลุ่ม K ดังกล่าวสังกัดอยู่มากที่สุดให้กับสมาชิกใหม่ ข้อมูลการจำแนกโดยใช้ข้อมูลข้างเคียง K ตัว ประกอบด้วย แอททริบิวต์หลายตัวแปร X_i ซึ่งจะนำมาใช้ในการแบ่งกลุ่ม Y_i โดยระบุค่าตัวเลขจำนวนเต็มบวกให้กับ K ซึ่งค่านี้จะเป็นตัวบอกจำนวนของกรณี (Case) ที่จะต้องค้นหาในการทำนายกรณีใหม่ โดยบทความนี้กำหนด 1-KNN หมายถึง อัลกอริทึมนี้จะค้นหา 1 กรณีที่มีลักษณะใกล้เคียงกับกรณีใหม่ (1 Nearest Cases) การนำระยะทางที่หาได้จากสมาชิกในข้อมูลตัวอย่างฝึกฝน มาเรียงลำดับจากน้อยไปหามากแล้วเลือกสมาชิกที่มีระยะทาง (Distance) ใกล้เคียงที่สุดออกมา K ตัวโดยใช้การวัดระยะทางแบบ Euclidean distance มีหลักการ คือ การวัดระยะทางระหว่างสองวัตถุ ถ้าวัตถุห่างกันมากแสดงว่าวัตถุนั้นมีความคล้ายกันน้อย ถ้ามีค่าน้อยก็แสดงว่ามีความคล้ายคลึงกันมาก โดยที่ ค่า P_i แทนคุณสมบัติจากฐานข้อมูล Q_i แทนคุณสมบัติที่ผู้ใช้ระบุ ดังแสดงในสมการ

$$\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}.$$

กฎของริปเปอร์ (Cohen, 1995) หลักการของวิธีการ ประกอบด้วย 2 เฟส คือ เฟสแรกทำการระบุกฎเริ่มต้น และเฟสที่สองจะระบุค่า post-process rule optimization โดยข้อมูลที่ถูกการเรียนรู้ (Training) จะถูกแบ่งไปเป็น growing set และ pruning set โดยอัลกอริทึมนี้จะสร้างกฎความสัมพันธ์ใน greedy ในขณะที่สร้างกฎริปเปอร์นั้น จะหาค่าที่ดีที่สุดสำหรับ growing set ใน rulespace ซึ่งจะอธิบายได้จาก BNF หลังจากได้ growing set ก็จะทำการ pruning ข้อมูลเมื่อเสร็จจะได้กลุ่มตัวอย่างที่เหมือนกันออกมาที่ครอบคลุมกฎของ training set จากนั้นก็จะลบทิ้งซึ่ง training data ที่เหลือจะถูกแบ่งใหม่อีกครั้ง หลังจากเรียนรู้ตามกฎแล้ว เพื่อช่วยแก้ปัญหาที่เกิดมาจากการแบ่งแยกกลุ่มที่ผิดพลาด ซึ่งกระบวนการนี้จะกระทำซ้ำจนกระทั่งผลเป็นที่น่าพอใจ

เครือข่ายฟังก์ชันฐานรัศมี (Radial Basis Function Network หรือ RBF network) (อาทิตย์ ศรีแก้ว, 2552; พรพล ธรรมรงค์รัตน์, ลัดดา ปรีชาวีรกุล และ วิภาดา เวทย์ประสิทธิ์, 2008) เป็นเครือข่ายไปข้างหน้าประเภทหนึ่ง ที่ได้รับการยอมรับว่ามีประสิทธิภาพสูงเครือข่ายหนึ่ง เครือข่าย RBF แตกต่างไปจากเครือข่ายเพอร์เซ็พตรอนแบบหลายชั้น (Multi-layer perceptron) ตรงที่เครือข่าย RBF นั้นมีชั้นซ่อนเร้นเพียงชั้นเดียว เครือข่ายฟังก์ชันฐานรัศมี สามารถพิจารณาเป็นฟังก์ชันการส่ง (Mapping function) ของความสัมพันธ์ระหว่างคู่รูปแบบอินพุตและเอาต์พุตได้ โดยการเรียนรู้ของเครือข่ายเป็นการปรับค่าน้ำหนักประสาทให้ได้ฟังก์ชันการส่งที่เหมาะที่สุด RBF Neural Network ประกอบด้วย ชั้นข้อมูลเข้า (Input Layer) ขั้นซ่อน (Hidden Layer) และ ชั้นข้อมูลออก (Output Layer) ซึ่งมีเกาซ์เซียนฟังก์ชัน (Gaussian Function) เป็นฟังก์ชันกระตุ้นในชั้นซ่อน ดังสมการ

$$\emptyset j(x) = \exp\left[\left[-\frac{\|x - cjE\|}{2\sigma^2 j}\right]\right] \text{ เมื่อ } j = 1, 2, ..., n$$

โดยที่ $\emptyset$ คือ ข้อมูลออกของนิวรอลที่ j ในชั้นซ่อน x คือ เวกเตอร์ข้อมูลเข้า c_j และ σ_j คือ ศูนย์กลางและช่วงกว้างของนิวรอลที่ j ตามลำดับ ข้อมูลออกของโครงข่าย RBF คำนวนดังสมการ

$$y = i_c(k+1) = \sum_{i=1}^{n} w_j \emptyset_j(x)$$

โดยที่ n คือ จำนวนของนิวรอลในชั้นซ่อน w_j คือ น้ำหนักระหว่างชั้นซ่อนและชั้นข้อมูลออก และ y คือ ผลลัพธ์ สำหรับเครือข่ายฟังก์ชันฐานรัศมี บทความนี้ ได้เลือก Normalized Gaussian เป็นฟังก์ชันที่ใช้ในการทดลอง โดยตั้งค่าพารามิเตอร์ numClusters 2 และ minStdDev 0.1

วิธีการดำเนินการ

บทความนี้ทำการทดลองการลดคุณลักษณะร่วมกับอัลกอริทึมการจัดหมวดหมู่เอกสาร โดยมุ่งเน้นจัดหมวดหมู่เอกสารภาษาไทย โดยทดสอบกับเอกสารประเภทข่าวอิเล็กทรอนิกส์จากหนังสือพิมพ์ไทยรัฐ จำนวน 10 กลุ่ม ได้แก่ กลุ่มการศึกษา กลุ่มบันเทิง กลุ่มสังคม กลุ่มการเมือง กลุ่มเทคโนโลยี กลุ่มกีฬา กลุ่มข่าวต่างประเทศ กลุ่มเกษตร กลุ่มเศรษฐกิจ กลุ่มวัฒนธรรม โดยมีจำนวนกลุ่มตัวอย่างทั้งหมด 12000 เอกสาร ทำการทดสอบด้วยวิธี 10-fold cross validation โดยแบบจำลองที่นำเสนอในบทความนี้ จะทำการตัดคำโดยวิธีการตัดคำแบบยาวที่สุด (Longest Matching) โดยใช้พจนานุกรมฉบับ Lexitron และทำการกำจัดคำหยุดและทำรากศัพท์จากฐานข้อมูลที่กำหนดขึ้น หลังจากนั้นทำการลดขนาดคุณลักษณะด้วย วิธีการเพิ่มของข้อมูล (Information Gain) ซึ่งวิธีการลดขนาดคุณลักษณะดังกล่าวเป็นวิธีที่ง่ายและใช้เวลาการประมวลผลไม่มากนักแต่สามารถคัดเลือกคุณลักษณะที่มีนัยสำคัญได้ดี ผลที่ได้จากขั้นตอนดังกล่าวจะทำการลดขนาดของคุณลักษณะของเอกสารลง แล้วจะถูกส่งเข้าเครื่องจักรการเรียนรู้แบบมีผลเฉลยด้วยอัลกอริทึม

ต้นไม้ตัดสินใจ (Decision Tree) **ซัพพอร์ทเวกเตอร์แมชชีน** (Support Vector Machine) **เนอีฟเบย์** (Naïve-Bayes) **เครือข่ายฟังก์ชันฐานรัศมี** (Radial Basis Function Network) **เคเนียเรสเนเบอร์** (K-Nearest Neighbor) **และ กฎริปเปอร์** (Ripper Rules) **เรียนรู้ ทำการทดสอบเปรียบเทียบประสิทธิภาพด้านความถูกต้อง ความแม่นยำ โดยใช้วิธีการประเมินความสามารถของแบบจำลอง โดยวัดที่ประสิทธิผลของการจำแนกหมวดหมู่ตามแนวคิดทางด้านการค้นคืนสารสนเทศ ซึ่งก็คือ การวัดค่าความแม่นยำ** (Precision) **และค่าความระลึก** (Recall) **และค่า** F-Measure **ซึ่งคำนวณได้ดังสมการ (นิเวศ จิระวิชิตชัย, ปริญญา สงวนสัตย์ และพยุง มีสัจ, 2553)**

$$recall = \frac{a}{a + c}$$

$$precision = \frac{a}{a + b}$$

$$F - measure = \frac{2 \text{ x } (Precision \text{ x } Recall)}{Precision + Recall}$$

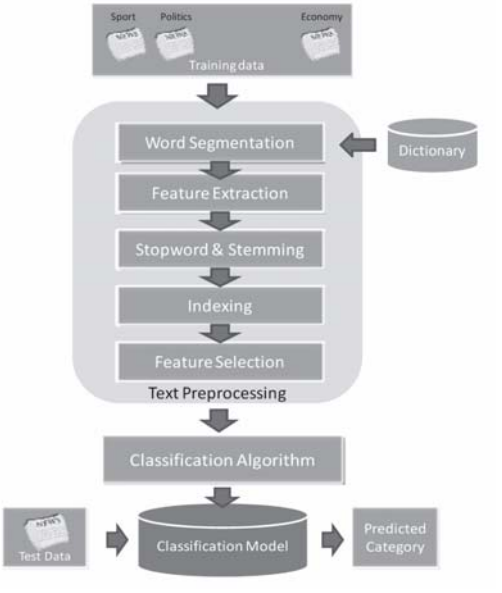
โดยให้ a = จำนวนเอกสารที่อยู่ในหมวดหมู่ C_j และตัวจำแนกทำนายว่าอยู่ในหมวดหมู่ C_j

b = จำนวนเอกสารที่ไม่อยู่ในหมวดหมู่ C_j และตัวจำแนกทำนายว่าอยู่ในหมวดหมู่ C_j

c = จำนวนเอกสารที่อยู่ในหมวดหมู่ C_j และตัวจำแนกทำนายว่าไม่อยู่ในหมวดหมู่ C_j

d = จำนวนเอกสารที่ไม่อยู่ในหมวดหมู่ C_j และตัวจำแนกทำนายว่าไม่อยู่ในหมวดหมู่ C_j

C_j = กลุ่มประเภทของเอกสารที่สนใจวัดประสิทธิภาพ



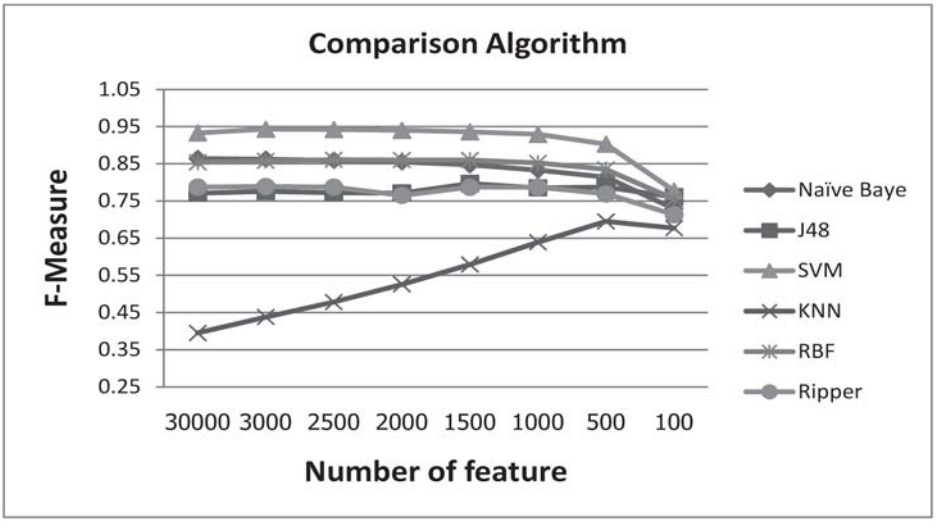
ภาพที่ 1: แบบจำลองการจัดหมวดหมู่เอกสารภาษาไทย

ผลการทดลอง

การทดลองการลดคุณลักษณะร่วมกับอัลกอริทึมการจัดหมวดหมู่เอกสารซึ่งประกอบด้วย อัลกอริทึมต้นไม้ตัดสินใจ (Decision Tree) ซัพพอร์ทเวกเตอร์แมชชีน (Support Vector Machine) เนอีฟเบย์ (Naïve-Bayes) เครือข่ายฟังก์ชันฐานรัศมี (Radial Basis Function Network) เคเนียเรสเนเบอร์ (K-Nearest Neighbor) และกฎริปเปอร์ (Ripper Rules) โดยทดลองกับกลุ่มตัวอย่างเอกสารข่าวภาษาไทยจำนวน 10 ประเภท ซึ่งได้ผลการทดลอง ดังนี้

ตารางที่ 1: เปรียบเทียบประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทยด้าน F-Measure

Feature	Naïve Baye	J48	SVM	KNN	RBF	Ripper
30000	0.861	0.771	0.933	0.395	0.854	0.785
3000	**0.862**	0.776	**0.943**	0.438	0.858	**0.789**
2500	0.859	0.772	0.942	0.478	**0.861**	0.787
2000	0.855	0.771	0.940	0.526	0.860	0.765
1500	0.847	**0.797**	0.936	0.579	0.860	0.787
1000	0.833	0.785	0.929	0.639	0.852	0.787
500	0.814	0.787	0.903	**0.695**	0.835	0.769
100	0.729	0.761	0.777	0.677	0.755	0.713
Average	**0.833**	**0.778**	**0.913**	**0.553**	**0.842**	**0.773**

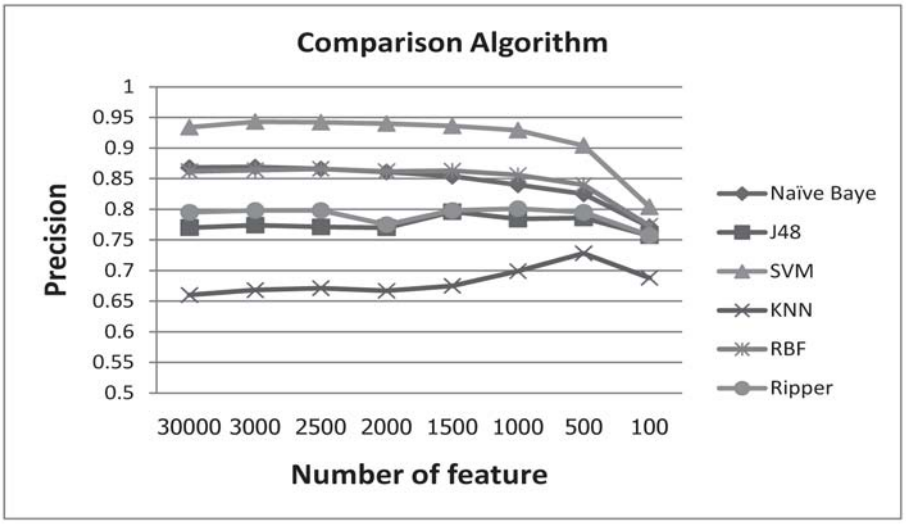


ภาพที่ 2: เปรียบเทียบประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทยด้าน F-Measure

ผลทดลองโดยใช้ Information Gain ในการลดคุณลักษณะและทำการเรียนรู้ด้วยอัลกอริทึมแบบต่าง ๆ โดยวัดจากค่าเฉลี่ย F-Measure สามารถสรุปได้ว่า อัลกอริทึม SVM ให้ประสิทธิภาพการจัดหมวดหมู่โดยเฉลี่ยออกมาดีที่สุด คือ 91.3% รองลงมาเป็นอัลกอริทึม RBF ให้ประสิทธิภาพการจัดหมวดหมู่ 84.2% อัลกอริทึม Naïve Baye ให้ประสิทธิภาพการจัดหมวดหมู่ 83.3% อัลกอริทึม J48 ให้ประสิทธิภาพการจัดหมวดหมู่ 77.8% อัลกอริทึม Ripper ให้ประสิทธิภาพการจัดหมวดหมู่ 77.3% และตัวสุดท้าย คือ อัลกอริทึม KNN ให้ประสิทธิภาพการจัดหมวดหมู่ 55.3% ตามลำดับ เมื่อตรวจสอบการลดคุณลักษณะที่ทำให้ค่า F-Measure สูงสุดของแต่ละอัลกอริทึม พบว่า อัลกอริทึม SVM ที่จำนวน 3000 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 94.3% อัลกอริทึม Naïve Baye ที่จำนวน 3000 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 86.2% อัลกอริทึม RBF ที่จำนวน 2500 คุณลักษณะ ให้ประสิทธิภาพสูงสุดคือ 86.1% อัลกอริทึม J48 ที่จำนวน 1500 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 79.7% อัลกอริทึม Ripper ที่จำนวน 3000 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 78.9% อัลกอริทึม KNN ที่จำนวน 500 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 69.5% ตามลำดับ ดังภาพที่ 2

ตารางที่ 2: เปรียบเทียบประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทยด้าน Precision

Feature	Naïve Baye	J48	SVM	KNN	RBF	Ripper
30000	0.868	0.770	0.934	0.660	0.862	0.795
3000	**0.869**	0.774	**0.943**	0.668	0.864	0.798
2500	0.866	0.771	0.942	0.671	**0.866**	0.798
2000	0.861	0.770	0.940	0.667	0.862	0.775
1500	0.854	**0.796**	0.936	0.675	0.863	0.798
1000	0.840	0.784	0.929	0.699	0.856	**0.801**
500	0.825	0.786	0.904	**0.728**	0.840	0.795
100	0.771	0.757	0.804	0.688	0.773	0.757
Average	**0.844**	**0.776**	**0.917**	**0.682**	**0.848**	**0.790**

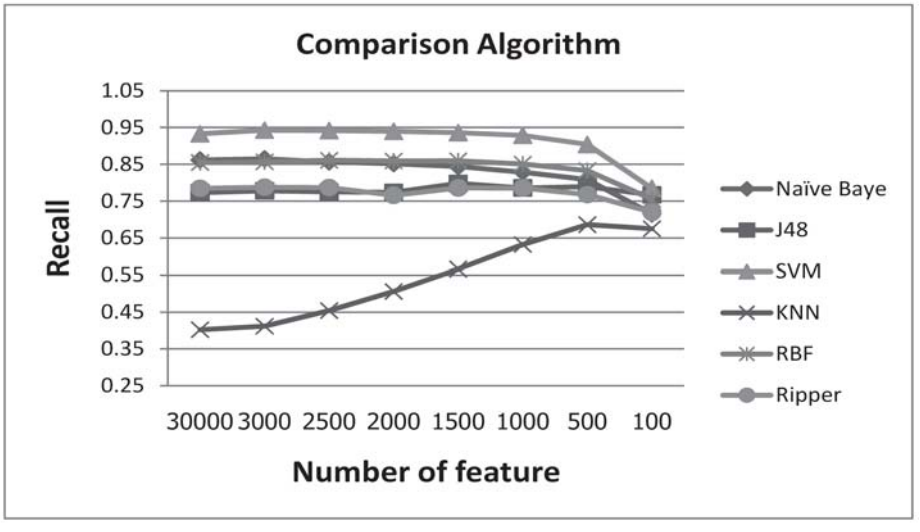


ภาพที่ 3: เปรียบเทียบประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทยด้าน Precision

ผลทดลองโดยใช้ Information Gain ในการลดคุณลักษณะและทำการเรียนรู้ด้วยอัลกอริทึมแบบต่าง ๆ โดยวัดจากผลรวมของค่าเฉลี่ย Precision สามารถสรุปได้ว่า อัลกอริทึม SVM ให้ประสิทธิภาพการจัดหมวดหมู่โดยเฉลี่ยออกมาดีที่สุด คือ 91.7% รองลงมาเป็นอัลกอริทึม RBF ให้ประสิทธิภาพการจัดหมวดหมู่ 84.8% อัลกอริทึม Naïve Baye ให้ประสิทธิภาพการจัดหมวดหมู่ 84.4% อัลกอริทึม Ripper ให้ประสิทธิภาพการจัดหมวดหมู่ 79.0% อัลกอริทึม J48 ให้ประสิทธิภาพการจัดหมวดหมู่ 77.6% และตัวสุดท้าย คือ อัลกอริทึม KNN ให้ประสิทธิภาพการจัดหมวดหมู่ 68.2% ตามลำดับ เมื่อตรวจสอบการลดคุณลักษณะที่ทำให้ค่าความแม่นยำสูงสุดของแต่ละอัลกอริทึม พบว่า อัลกอริทึม SVM ที่จำนวน 3000 คุณลักษณะ ให้ประสิทธิภาพสูงสุดคือ 94.3% อัลกอริทึม Naïve Baye ที่จำนวน 3000 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 86.9% อัลกอริทึม RBF ที่จำนวน 2500 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 86.6% อัลกอริทึม Ripper ที่จำนวน 1000 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 80.1% อัลกอริทึม J48 ที่จำนวน 1500 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 79.6% อัลกอริทึม KNN ที่จำนวน 500 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 72.8% ตามลำดับ ดังภาพที่ 3

ตารางที่ 3: เปรียบเทียบประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทยด้าน Recall

	Naïve Baye	J48	SVM	KNN	RBF	Ripper
30000	0.862	0.773	0.933	0.402	0.855	0.785
3000	**0.865**	0.778	**0.943**	0.412	0.857	**0.788**
2500	0.857	0.775	0.942	0.454	**0.861**	0.787
2000	0.852	0.774	0.940	0.506	0.859	0.766
1500	0.844	**0.798**	0.936	0.566	0.860	0.786
1000	0.829	0.786	0.929	0.632	0.851	0.786
500	0.809	0.789	0.904	**0.686**	0.833	0.768
100	0.715	0.767	0.785	0.675	0.755	0.720
Average	**0.829**	**0.780**	**0.914**	**0.542**	**0.841**	**0.773**



ภาพที่ 4: เปรียบเทียบประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทยด้าน Recall

ผลทดลองโดยใช้ Information Gain ในการลดคุณลักษณะและทำการเรียนรู้ด้วยอัลกอริทึมแบบต่าง ๆ โดยวัดจากผลรวมของค่าเฉลี่ย Recall สามารถสรุปได้ว่า อัลกอริทึม SVM ให้ประสิทธิภาพการจัดหมวดหมู่โดยเฉลี่ยออกมาดีที่สุด คือ 91.4% รองลงมาเป็นอัลกอริทึม RBF ให้ประสิทธิภาพการจัดหมวดหมู่ 84.1% อัลกอริทึม Naïve Baye ให้ประสิทธิภาพการจัดหมวดหมู่ 82.9% อัลกอริทึม J48 ให้ประสิทธิภาพการจัดหมวดหมู่ 78.0% อัลกอริทึม Ripper ให้ประสิทธิภาพการจัดหมวดหมู่ 77.3% และตัวสุดท้าย คือ อัลกอริทึม KNN ให้ประสิทธิภาพการจัดหมวดหมู่ 54.2% ตามลำดับ เมื่อตรวจสอบการลดคุณลักษณะที่ทำให้ค่าความระลึกสูงสุดของแต่ละอัลกอริทึม พบว่า อัลกอริทึม SVM ที่จำนวน 3000 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 94.3% อัลกอริทึม Naïve Baye ที่จำนวน 3000 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 86.5% อัลกอริทึม RBF ที่

จำนวน 2500 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 86.1% อัลกอริทึม J48 ที่จำนวน 1500 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 79.8% อัลกอริทึม Ripper ที่จำนวน 3000 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 78.8% อัลกอริทึม KNN ที่จำนวน 500 คุณลักษณะ ให้ประสิทธิภาพสูงสุด คือ 68.6% ตามลำดับ ดังภาพที่ 4

สรุปผลและข้อเสนอแนะ

บทความนี้ได้นำเสนอวิธีการพัฒนาประสิทธิภาพการจัดหมวดหมู่เอกสารภาษาไทย โดยนำเสนอแบบจำลองการจัดหมวดหมู่ของเอกสารภาษาไทยแบบอัตโนมัติ ทดสอบประสิทธิภาพแบบจำลองการจัดหมวดหมู่เอกสารภาษาไทยกับ อัลกอริทึมต้นไม้ตัดสินใจ (Decision Tree) ซัพพอร์ทเวกเตอร์แมชชีน (Support Vector Machine) เนอีฟเบย์ (Naïve-Bayes) เครือข่ายฟังก์ชันฐานรัศมี (Radial Basis Function Network) เคเนียเรสเนเบอร์ (K-Nearest Neighbor) และ กฎริปเปอร์ (Ripper Rules) โดยใช้วิธีการลดคุณลักษณะร่วมกับอัลกอริทึมเครื่องจักรการเรียนรู้ เพื่อศึกษาวิธีการลดคุณลักษณะที่เหมาะสมและมีประสิทธิภาพในการจัดหมู่เอกสารข่าวภาษาไทย จากการทดลอง พบว่า การลดคุณลักษณะด้วยวิธี Information Gain เพื่อลดมิติของข้อมูล แล้วส่งเข้าเครื่องจักรการเรียนรู้และวัดประสิทธิภาพจากการลดคุณลักษณะที่ทำให้ค่า F-Measure สูงสุด สามารถสรุปได้ว่า อัลกอริทึม SVM ให้ประสิทธิภาพสูงสุด คือ 94.3% รองลงมาเป็นอัลกอริทึม Naïve Baye ให้ประสิทธิภาพสูงสุด คือ 86.2% อัลกอริทึม RBF ให้ประสิทธิภาพสูงสุด คือ 86.1% อัลกอริทึม J48 ให้ประสิทธิภาพสูงสุด คือ 79.7% อัลกอริทึม Ripper ให้ประสิทธิภาพสูงสุด คือ 78.9% อัลกอริทึม KNN ให้ประสิทธิภาพสูงสุด คือ 69.5% ตามลำดับ

บทความนี้สกัดคุณลักษณะโดยใช้คำเดี่ยว (Single word) เนื่องจากเป็นวิธีการตัดคำที่ให้ผลลัพธ์ในการตัดคำออกมาถูกต้องมากที่สุดในงานด้านจัดกลุ่มเอกสารและ อัลกอริทึม Support Vector Machine ให้ประสิทธิภาพดีที่สุด ทั้งนี้เนื่องมาจาก SVM มีพฤติกรรมที่จะแยกแยะข้อมูลโดยใช้สมการระนาบหลายมิติ โดยจะพยายามหาจุดข้อมูลที่ทำให้ได้สมการระนาบหลายมิติที่ใช้แบ่งแยกดีที่สุด (Optimal Hyperplane) ความถูกต้องที่สุด โดยพิจารณาจากระยะห่าง (Margin) ระหว่างคลาส ซึ่งเส้นระนาบที่ดีที่สุดนี้จะสามารถจำแนกกลุ่มเอกสารออกมาได้อย่างมีประสิทธิภาพ ผลจากการทดลองลดขนาดคุณลักษณะและทดสอบด้วยอัลกอริทึม SVM จากกลุ่มตัวอย่าง พบว่าสามารถลดคุณลักษณะลงได้มากถึง 90% โดยคุณลักษณะจริงที่สกัดมาจากเอกสารทั้งหมดได้มีจำนวน 30000 คุณลักษณะ ใช้ในทดลองเพียง 3000 คุณลักษณะ เนื่องจากเป็นจำนวนคุณลักษณะที่ส่งผลดีในการจำแนกเอกสารสูงสุดโดยการลดลงของคุณลักษณะดังกล่าว ไม่ส่งผลให้ประสิทธิภาพในการจัดหมวดหมู่เอกสารลดลงแต่อย่างใด แต่สามารถลดทรัพยากรของระบบและลดระยะเวลาในการประมวลผลได้เป็นอย่างมาก จากผลการทดลองนี้สามารถนำแบบจำลองนี้ไปประยุกต์ใช้ประโยชน์ในการสร้างระบบจัดหมวดหมู่เอกสารอัตโนมัติและสามารถนำมาประยุกต์ใช้กับงานด้าน

อื่น ๆ เช่น การคัดกรองเอกสาร (Document Filtering) การจัดทำดัชนีอัตโนมัติเพื่อใช้ในการค้นคืนเอกสาร (Automatic Indexing for IR System) การจัดหมวดหมู่ของเว็บเพจ (Web Page Classification) เป็นต้น

กิตติกรรมประกาศ

ขอขอบพระคุณ ดร.ชูชาติ หฤไชยะศักดิ์ ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ที่อนุเคราะห์กลุ่มตัวอย่างที่ใช้ในการทดลองในบทความนี้

เอกสารอ้างอิง

นิเวศ จิระวิชิตชัย, ปริญญา สงวนสัตย์ และพยุง มีสัจ. (2553). การศึกษาทดลองเทคนิคการลดคุณลักษณะ และอัลกอริทึมการจัดหมวดหมู่ของเอกสารภาษาไทย. *วารสารวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.* ปีที่ 18 ฉบับที่ 2.

พรพล ธรรมรงค์รัตน์, ลัดดา ปรีชาวีรกุล และวิภาดา เวทย์ประสิทธิ์. (2008). การจำแนกประเภทเว็บเพจโดยใช้ค่าความถี่เอกสารและซัพพอร์ตเวกเตอร์แมชชีน. *The 12th National Computer Science and Engineering Conference.*

วัลลภ อินทร์ฉ่ำ. (2548). *ระบบการจัดหมวดหมู่เอกสารภาษาไทยอัตโนมัติโดยใช้ SVM ร่วมกับการประมวลผลภาษา.* วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต มหาวิทยาลัยเกษตรศาสตร์.

ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ. ข้อมูลคำศัพท์ที่พบบ่อยจากฐานข้อมูลประโยคที่มาจากหนังสือพิมพ์. http://thailang.nectec.or.th/thaichar/word_thai.php

อาทิตย์ ศรีแก้ว. (2552). *ปัญญาเชิงคำนวณ.* สาขาวิชาวิศวกรรมไฟฟ้า สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี.

Aas., Eikvil. (1999). Text Categorization: a Survey. *Report Norwegian Computing Center.*

Cortes C. and Vapnik V. (1995). Support-Vector Networks. *Machine Learning.* Volume 20, Number 3, September.

Charoenpornsawat, P. (1999). *Feature-based Thai Word Segmentation. Master's Thesis.* Computer Engineering. Chulalongkorn University, Bangkok, Thailand.

Cohen, W. (1995). Fast effective rule induction. In Machine Learning: *the 12 th International Conference,* LakeTaho, CA, Morgan Kaufmann.

David W. Aha and Dennis Kibler and Marc K. Albert. (1991). Instance-based learning algorithms. *Machine Learning,* Volume 6, Number 1, pp. 37-66.

Jaruskulchai, C. (1998). *An Automatic Indexing for Thai Text Retrieval.* PhD Thesis, George Washington University, USA.

Lewis, D. (1998). Naïve bayes at forty: The independence assumption in information retrieval. *Proceedings of European Conference on Machine Learning,* pp. 4-15.

Marquez, Llus. (2000). Machine learning and natural language processing. *Technical Report Departament de Llenguatges Sistemes Informatics (LSI),* Barcelona, Spain.

Quinlan, J.R. (1993). *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers.

Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys* (CSUR). March.

Yang, Perderson, O. (1997). A Comparative Study on Feature Selection in Text Categorization. *Proceedings of ICML-97, 14 ^{tb} International Conference on Machine Learning.*