การจัดลำดับความสำคัญของคำดัชนีของบทความวิจัยในรูปแบบเอ็กซ์เอ็มแอล

RANKING INDEX TERM OF RESEARCH PAPER IN XML FORMAT

เกียรติณรงค์ ทองประเสริฐ

KIATNARONG TONGPRASERT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2548

ISBN 974-15-1809-9

RANKING INDEX TERM OF RESEARCH PAPER IN XML FORMAT

KIATNARONG TONGPRASERT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN COMPUTER ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2005

ISBN 974-15-1809-9

หัวข้อวิทยานิพนธ์	การจัดลำดับความสำคัญของคำดัชนีของบทความวิจัยในรูปแบบเอ็กซ์เอ็มแอล
นักศึกษา	นายเกียรติณรงค์ ทองประเสริฐ
รหัสนักศึกษา	43061612
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
พ.ศ.	2548
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ดร.วิศิษฏ์ หิรัญกิตติ

บทคัดย่อ

การจัดเรียงลำดับคำดัชนีถือว่าเป็นขั้นตอนสำคัญสำหรับการค้นคืนเอกสาร ในวิทยานิพนธ์นี้เราได้วิจัยเพื่อค้นหาวิธีที่ดีในการจัดเรียงลำดับคำดัชนีในเอกสาร XML [1] ตามแบบเวคเตอร์โมเดล [2] ซึ่งตามวิธีเวคเตอร์มาตรฐานจะจัดเรียงลำดับคำดัชนีตามความถี่ที่ปรากฏในเอกสารเพียงอย่างเดียว [3] งานวิจัยนี้ได้ปรับปรุงวิธีการจัดเรียงลำดับคำดัชนีแบบเวคเตอร์สำหรับเอกสาร XML โดยมีการเพิ่มค่าน้ำหนักที่แตกต่างกันให้กับคำที่ปรากฏในแท็ก XML ที่ต่างกัน โดยแท็กที่มีความสำคัญมากกว่าจะกำหนดให้มีค่าน้ำหนักมากกว่า นอกจากนี้ยังใช้จำนวนของคำในแท็กมาใช้ร่วมคำนวณเพื่อเป็นค่าน้ำหนักด้วย โดยการคิดค่าน้ำหนัก 2 วิธีดังกล่าวร่วมกับวิธีการคำนวณกับความถี่ของคำแล้ว ทำให้เราแยกแยะพิจารณาการคำนวณค่าน้ำหนักคำดัชนีได้เป็น 4 วิธี คือ 1. วิธีรวมค่าน้ำหนักแท็กที่พบ 2. วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบ 3. วิธีรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก 4. วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก จากผลการทดลองหาวิธีการกำหนดค่าน้ำหนักแท็กสำหรับแต่ละวิธีคำนวณค่าน้ำหนักคำดัชนี โดยใช้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยเป็นเครื่องมือวัดประสิทธิภาพ พบว่าการให้ค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึมเป็นการให้ค่าน้ำหนักแท็กที่ดีที่สุด และผลการทดลองหาวิธีคำนวณน้ำหนักคำดัชนีที่ดีที่สุดสำหรับ 4 วิธีข้างต้น โดยใช้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยเป็นเครื่องมือวัดประสิทธิภาพ และใช้การให้ค่าน้ำหนักแท็กโดยผู้ใช้เป็นค่าน้ำหนักแท็กที่นำมาทดลอง พบว่าวิธีที่ 4 มีประสิทธิภาพดีกว่าวิธีการคำนวณอื่น ๆ รองลงมาคือวิธีที่ 3 , วิธีที่ 1 และวิธีที่ 2 ตามลำดับ

Thesis Title	Ranking Index Term of Research Paper in XML Format
Student	Kiatnarong Tongprasert
Student ID.	43061612
Degree	Master of engineering
Programme	Computer engineering
Year	2005
Thesis Advisor	Dr. Visit Hirankitti

ABSTRACT

Ranking index terms is an important step for efficient information retrieval. In this thesis we have investigated in order to find a good method for ranking index terms based on a vector model for XML documents [1]. According to this model [2], indexes from documents are ranked according to only the frequency of their appearance in the documents [3]. In this thesis we have improved the conventional vector model by adding different weights to words appearing in different tags of XML documents. The more important the tag, the more weight is assigned to it. In addition, the number of words appearing (a word count) in a tag is also taken into account for determining weights for the index terms. By considering the 3 factors, i.e. tag weights, the word count in a tag, and word frequency (according to the conventional vector model), we have identified 4 ways (methods) for ranking index terms in XML documents: (1) using tag weights only (2) using tag weights and word frequency (3) using tag weights and tags' word counts (4) using tag weights, word frequency, and tags' word counts altogether. In the experimentation we use average Precision and average R-Precision to compare each method with vary tag weight. The result has shown that defining tag weight by genetic algorithm is the best method for every method and for each method the fourth method is the best method follow by the third, the first and the second method.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ดร.วิศิษฏ์ หิรัญกิตติ ที่ให้ความช่วยเหลือ ให้คำชี้แนะในการแก้ปัญหาที่ดีแก่ข้าพเจ้า

ขอขอบพระคุณ ดร.ชุติเมษฎ์ ศรีนิลทา และ ดร.สุรินทร์ กิตติธรกุล กรรมการสอบหัวข้อและโครงร่างวิทยานิพนธ์ที่ได้กรุณาให้คำแนะนำตลอดจนข้อชี้แนะ จนในที่สุดทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลงได้

ขอขอบคุณ บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่สนับสนุนให้ทุนในงานวิจัยนี้

ขอขอบคุณ วารสารพระจอมเกล้าลาดกระบัง และ วิศวสารลาดกระบัง ที่ให้ข้อมูลที่นำมาใช้ในงานวิจัยนี้

สุดท้ายต้องขอขอบคุณเพื่อน พี่ และน้อง ที่ให้ความช่วยเหลือและคำปรึกษาที่ดี

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้กับบิดามารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า

เกียรติณรงค์ ทองประเสริฐ

สารบัญ

สารบัญ(ต่อ)

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

สารบัญตาราง(ต่อ)

สารบัญตาราง(ต่อ)

สารบัญรูป

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เมื่อหลายปีที่ผ่านมา ห้องสมุดเป็นแหล่งของหนังสือและข้อมูลความรู้จำนวนมาก เพื่อให้ได้ข้อมูลในหนังสือที่ต้องการในห้องสมุด ผู้ใช้ต้องค้นหาหนังสือผ่านเลขทะเบียนหนังสือ แล้วค้นหาในหนังสืออีกครั้งว่ามีข้อมูลที่ต้องการหรือไม่ ซึ่งการค้นหาข้อมูลแบบนี้ต้องกระทำด้วยมือ เมื่อนำการค้นหาหนังสือในห้องสมุดมาเทียบเคียงกับการค้นหาข้อมูลในอินเตอร์เน็ท ที่ซึ่งประกอบด้วยข้อมูลจำนวนมากและชนิดที่แตกต่างกัน ทำให้การค้นหาข้อมูลแบบเดิมยากในการที่จะค้นคืนข้อมูลที่ต้องการ สิ่งที่กล่าวมาข้างต้นเป็นสาเหตุที่ทำให้เกิดการพัฒนาระบบค้นคืนข้อมูล

ปัจจุบันนี้อินเทอร์เน็ท (Internet) ได้เข้ามามีบทบาทในชีวิตประจำวันมากขึ้น การบริการข้อมูลความรู้ผ่านทางอินเทอร์เน็ทในรูป WWW (World Wide Web) ก็เป็นอีกบริการหนึ่งที่กำลังได้รับความนิยมเป็นอย่างมากในปัจจุบัน ข้อมูลทั้งหมดนี้สามารถเข้าถึงได้ในรูปแบบอิเล็กทรอนิกส์และมีหลากหลายรูปแบบ ข้อมูลส่วนมากสามารถดูผ่านเว็บบราวเซอร์ได้โดยอยู่ในรูปของไฟล์ HTML ไฟล์ HTML เขียนขึ้นจากภาษา HTML ซึ่งภาษานี้จัดเตรียมเพียงเซตของแท็กเพื่อใช้นำเสนอข้อมูลมากกว่าเนื้อหาของข้อมูลในเอกสาร โดยแบ่งเอกสารเป็นหน่วย ๆ ด้วยแท็กที่ไม่ได้แสดงความหมายของข้อมูลภายในทำให้ไม่สื่อถึงเอกสารที่มีโครงสร้าง

เอกสารที่มีโครงสร้างคือเอกสารที่ใส่โครงสร้างในเอกสารนั้น เนื่องจาก เอกสารชนิดนี้ได้พัฒนาเพื่อแลกข้อมูลระหว่างระบบกับระบบ ซึ่งนำไปสู่การพัฒนา SGML (Standard Generalized Markup Language) ซึ่งเป็น Meta language ที่ใช้อธิบายโครงสร้างเอกสาร ซึ่งทำให้ตัวแสดงผลแสดงเอกสาร HTML ที่อยู่ในแต่ละส่วนให้แตกต่างกัน นอกจากนี้ยังเป็นแหล่งความรู้เกี่ยวกับเอกสารแต่ละชนิดด้วย การค้นหาสามารถค้นหาโดยระบุเฉพาะส่วนในเอกสารได้ ซึ่งเป็นการง่ายขึ้นสำหรับผู้ใช้ในการหาความแตกต่างระหว่างเอกสารที่ตรงกับความต้องการกับเอกสารที่ไม่ตรงกับความต้องการด้วยการระบุส่วนของเอกสารที่ต้องการโดยผู้ใช้

Extensible Markup Language (XML) เป็นภาษามาร์คอัพ (Markup language) ที่เริ่มต้นพัฒนาโดย World Wide Web Consortium (W3C) จุดประสงค์หลักของภาษานี้ใช้เพื่อเป็นผู้ช่วยในการแลกเปลี่ยนของมูลในรูปแบบของเอกสารในเครือข่ายอินเตอร์เน็ท ทั้ง XML และ HTML เป็นสับเซตของ SGML XML ได้รับการออกแบบมาให้เป็นเวอร์ชันขยายของ HTML เพื่ออนุญาตให้ผู้ใช้สามารถกำหนดแท็กขึ้นมาใช้ในเอกสารได้ ขณะที่ HTML มีการนำมาใช้เพื่ออธิบายว่าจะแสดงข้อมูลที่ต้องการอย่างไร XML ได้รับการพัฒนาเพื่อให้โอกาสสร้างระบบที่มีมาตรฐานและทำ

แอพพริเคชั่นต่าง ๆ สามารถเข้าไปใช้ข้อมูลในแต่ละแท็กได้เหมือนกัน การใช้ XML เหมือนเป็นเอกสารหนึ่งซึ่งมีโครงสร้างได้ด้วยการใช้แท็กใน XML ซึ่งกำหนดเฉพาะส่วน ข้อมูลใน XML เป็นเหมือนหนึ่งเขตข้อมูล (field) หนึ่งในฐานข้อมูล เพื่อให้ความหมายข้อมูลในแต่ละแท็ก

ไวยากรณ์อย่างเป็นทางการของเอกสารเรียกว่า Document Type Definition (DTD) ซึ่งเป็นโครงสร้างเอกสารที่ได้กำหนดขึ้นไว้ก่อนและใช้ในการตรวจสอบความถูกต้องเอกสาร XML ซึ่งภายใน DTD ได้กำหนดไวยากรณ์เอกสาร XML และทำให้ผู้นำไปใช้สามารถเข้าใจโครงสร้างของข้อมูลได้โดยง่าย

การค้นคืนเอกสาร XML กระทำได้หลายวิธี ได้แก่ การใช้คำสั่งคิวรี่จากเอกสาร XML เช่น XQuery, XML-QL, XQL, XPath เพื่อหาคำตอบจากเอกสาร XML เลย และการค้นหาเอกสารผ่านโมเดลการค้นคืนเอกสารเช่น บูลลีนโมเดล หรือ เวคเตอร์โมเดล งานวิจัยนี้ให้ความสำคัญการค้นคืนเอกสารผ่านโมเดลการค้นคืนเอกสารแบบเวคเตอร์โมเดล เนื่องจากเป็นโมเดลที่นิยมแพร่หลายในปัจจุบันและสามารถให้น้ำหนักกับคำดัชนี ซึ่งถ้าคำดัชนีนั้นสื่อถึงเอกสารที่มันอยู่ได้ดีก็จะทำให้การค้นหานั้นมีความถูกต้องเพิ่มขึ้นด้วย ซึ่งขึ้นอยู่กับการให้น้ำหนักของคำดัชนีในแต่ละเอกสาร

เนื่องจากเอกสาร XML เป็นเอกสารที่มีแท็กเป็นตัวบอกความสำคัญของข้อมูลในแท็กได้ ดังนั้นการให้ค่าน้ำหนักสำหรับแท็กจึงเป็นการให้น้ำหนักอย่างหนึ่งกับคำดัชนี ที่ใช้กันอยู่โดยทั่วไปที่ใช้เฉพาะความถี่ของคำที่พบในเอกสาร และในระบบเท่านั้น นอกจากค่าน้ำหนักแท็กที่กำหนดให้ในแต่ละแท็กซึ่งเป็นค่าคงที่แล้ว ยังมีค่าที่เปลี่ยนไปตามจำนวนของคำในแท็กซึ่งใช้เป็นค่าน้ำหนักของคำดัชนีอีกตัวหนึ่งด้วยนั่นคือ ค่าส่วนกลับความยาวแท็ก ซึ่งถ้าแท็กที่มีจำนวนของคำมากก็จะมีค่าน้ำหนักน้อยกว่าแท็กที่มีความยากของคำในแท็กน้อยกว่า

การคำนวณค่าน้ำหนักแท็กนี้มีหลายวิธี ซึ่งให้ผลการคำนวณที่คล้ายกันแต่ไม่เหมือนกันทีเดียวดังนี้การหาว่าวิธีใดน่าจะเหมาะสมที่สุดในการนำมาใช้คำนวณหาค่าน้ำหนักนี้จึงวัตถุประสงค์ของงานวิจัยนี้ นอกจากกนี้การหาค่าน้ำหนักแท็กที่มีจำนวนแท็กมากจึงเป็นการยากที่จะกำหนดค่าที่เหมาะสมให้ในแต่ละแท็กได้ จึงมีการนำเอาเจนิติกอัลกอริทึม ซึ่งเป็นอัลกอริทึมที่ใช้ในการค้นหาค่าแบบสุ่มโดยมีการกำหนดค่าฟังก์ชันที่เหมาะสมไว้เพื่อให้อัลกอริทึมปรับค่าตัวแปรให้ได้ค่าที่มีค่าฟังก์ชันที่เหมาะสมสูงสุด มาช่วยในการหาค่าน้ำหนักแท็กแต่ละแท็กด้วยเพื่อให้ได้ผลการคำนวณดัชนีที่ดี

1.2 ความมุ่งหมายและวัตถุประสงค์ของการวิจัย

1. เพื่อศึกษาการจัดเรียงลำดับคำดัชนีด้วยวิธีถ่วงน้ำหนักด้วยค่าน้ำหนักแท็กวิธีต่าง ๆ
2. เพื่อปรับปรุงให้การค้นคืนข้อมูลในเอกสาร XML มีความถูกต้องสูง
3. เพื่อค้นหาวิธีการจัดเรียงลำดับคำดัชนีของเอกสาร XML แบบเวคเตอร์ที่มีประสิทธิภาพสูง

1.3 แนวความคิดที่ใช้ในการวิจัย

ในการจัดเรียงลำดับความสำคัญของคำดัชนีในเอกสารเอกสาร XML ใช้แนวความคิดจากการที่เอกสาร XML มีแท็กที่สื่อความหมาย และทราบว่าข้อมูลที่อยู่แท็กนั้นหมายถึงอะไร และมีความสำคัญมากน้อยอย่างไร ดังนั้นจึงควรมีการกำหนดค่าน้ำหนักแท็กด้วยจึงจะช่วยเพิ่มค่าน้ำหนักให้กับคำดัชนีที่อยู่ในแท็กที่มีความสำคัญให้มีค่ามากกว่าค่าน้ำหนักของคำดัชนีที่อยู่ในแท็กที่มีความสำคัญน้อยกว่า และคิดค้นวิธีการจัดเรียงลำดับคำดัชนีโดยยึดแนวทางแบบเวคเตอร์โมเดล จากนั้นประเมินผลการจัดเรียงลำดับคำดัชนีเทียบกับการจัดเรียงคำดัชนีที่กระทำโดยผู้เชี่ยวชาญ

นอกจากที่จะพิจารณาค่าน้ำหนักของแท็กแล้วในงานวิจัยนี้ยังให้ความสำคัญกับค่าความยาวแท็ก (จำนวนคำที่มีภายในแท็กนั้น ๆ) ด้วย โดยนำค่าความยาวแท็กมาคิดเป็นค่าน้ำหนักด้วย ตรงนี้เราได้แนวคิดมาจากความจริงที่ว่าแท็กที่ความยาวมากมีการอธิบายความมาก น่าจะมีความสำคัญน้อยกว่าแท็กที่มีความยาวน้อยกว่า

จากแนวความคิดข้างต้นนำมาสู่การปรับปรุงสูตรที่ใช้ในการจัดเรียงลำดับคำดัชนีโดยนำเอาค่าน้ำหนักแท็กและความยาวแท็กรวมไว้ในสูตรการคำนวณด้วย ซึ่งสามารถแบ่งสูตรการคำนวณได้เป็น 4 วิธี คือ 1. วิธีรวมค่าน้ำหนักแท็กที่พบ 2. วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบ 3. วิธีรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก 4. วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก ซึ่งมีวิธีใหญ่ ๆ ในการคำนวณค่าน้ำหนักแท็ก 2 วิธี ซึ่งต่างกันที่การนำค่าน้ำหนักแท็กที่พบในแต่ละแท็กรวมกันก่อนแล้วจึงนำมารวมกับความถี่ของคำที่พบในเอกสารภายหลัง กับอีกแบบหนึ่งจะคำนวณค่าน้ำหนักแท็กกับค่าความถี่ของคำที่พบในแท็กนั้นแล้วจึงนำค่าที่ได้มารวมกันภายหลัง ส่วน 2 วิธีที่เหลือได้มาจากการปรับปรุง 2 วิธีแรกโดยนำค่าความยาวแท็กร่วมคิดด้วย

เนื่องจากการจะทำให้วิธีการเหล่านี้ทำงานได้ดีต้องมีการกำหนดน้ำหนักแท็กที่เหมาะสมให้กับแต่ละแท็ก ในที่นี้ได้นำเสนอวิธีการปรับค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึม ในการค่าน้ำหนักแท็กที่เหมาะสมเนื่องจากเป็นอัลกอริทึมที่มีประสิทธิภาพและสามารถนำมาใช้งานได้ง่าย

หลังจากได้ค่าน้ำหนักแท็กที่เหมาะสมในแต่ละวิธีแล้ว จึงใช้ค่าน้ำหนักแท็กที่ได้ไปทดลองคำนวณในทุก ๆ วิธีเพื่อหาว่าวิธีการคำนวณวิธีใดได้ค่าเฉลี่ยการจัดเรียงลำดับของคำดัชนีดีที่สุดเมื่อเปรียบเทียบกับการจัดเรียงลำดับคำดัชนีโดยผู้เชี่ยวชาญ

1.4 ขอบเขตการวิจัย

เอกสาร XML ที่ใช้ในงานวิจัยนี้เป็นเอกสารที่เก็บข้อมูลบทความทางวิชาการทั้งภาษาไทยและภาษาอังกฤษ โดยเก็บฉพาะตัวอักขระ ไม่รวมรูปภาพ จำนวน 300 เอกสาร ซึ่งมีโครงสร้าง

ตามที่ได้ออกแบบสำหรับงานวิจัยนี้ การจัดเก็บเอกสาร XML ใช้โปรแกรมฐานข้อมูลที่รองรับการทำงานกับเอกสาร XML

1.5 ประโยชน์ที่เกิดขึ้นจากงานวิจัย

1. วิทยานิพนธ์นี้ได้เสนอวิธีการจัดเรียงลำดับคำดัชนีในเอกสาร XML โดยใช้ค่าน้ำหนักแท็กและค่าจำนวนคำในแท็ก ซึ่งได้วิธีที่ดีที่สุดในการทดลองคือวิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก วิธีนี้มีข้อดีกว่าวิธีอื่น ๆ ที่สามารถจัดเรียงลำดับคำดัชนีได้ตรงกับผลการจัดเรียงคำดัชนีที่ทำขึ้นโดยผู้เชี่ยวชาญมากที่สุด ด้วยการคำนวณวิธีนี้ทำให้การให้น้ำหนักกับคำในเอกสารมีความละเอียดกว่าทุกวิธี รวมกับการใช้ค่าจำนวนคำในแท็กซึ่งเป็นค่าที่ไม่คงที่สามารถปรับเปลี่ยนได้ในแต่ละแท็กของแต่ละเอกสารทำให้การคำนวณค่าน้ำหนักของคำดัชนีด้วยวิธีนี้ให้ผลการจัดเรียงลำดับคำดัชนีที่ดีที่สุด
2. วิทยานิพนธ์นี้ได้เสนอการการปรับค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึม ซึ่งเป็นวิธีการหาค่าที่เหมาะสมให้กับปัญหาที่ต้องการ โดยการกำหนดค่าฟังก์ชันความเหมาะสม และรูปแบบโครโมโซม ซึ่งมีข้อดีคือทำให้คำตอบที่ได้มีความถูกต้องสูง ทำให้ค่าน้ำหนักแท็กที่ได้มีความถูกต้องสูง

1.6 ขั้นตอนของการศึกษา

ขั้นตอนของการวิจัยมีดังนี้

1. ออกแบบโครงสร้างเอกสาร XML ที่ใช้ในการเก็บข้อมูล
   1.1. ศึกษารูปแบบของบทความว่ามีส่วนหลักและส่วนที่ซ้ำกันส่วนใด ซึ่งข้อมูลที่จะจัดเก็บเป็นข้อมูลตัวอักษรภายในบทความทั้งหมด ไม่รวมรูปภาพ และตัวอักษรในรูปภาพซึ่งเป็นไฟล์แยกจากไฟล์อักษรในบทความ โดยบทความที่จัดเก็บสามารถจัดเก็บได้ทั้งภาษาไทยและภาษาอังกฤษ
   1.2. กำหนดชื่อแท็กให้กับส่วนของบทความเพื่อให้สื่อความหมาย
   1.3. ทดลองนำข้อมูลในบทความจัดเก็บลงในโครงสร้างเอกสาร XML ที่กำหนดขึ้นเพื่อสังเกตความสามารถการจัดเก็บข้อมูลว่าสามารถจัดเก็บข้อมูลที่ต้องการได้ครบถ้วนหรือไม่
   1.4. ปรับปรุงแก้ไขให้โครงสร้างเอกสาร XML ที่ออกแบบสามารถเก็บข้อมูลได้ครบและไม่ซับซ้อนเกินไป
2. ศึกษาการจัดเก็บเอกสาร XML ในฐานข้อมูล

2.1. ศึกษารูปแบบการจัดเก็บเอกสาร XML โดยใช้ฐานข้อมูล โดยการจัดเก็บแต่ละแบบเหมาะสมกับงานที่ต่างกัน

2.2. ทดลองนำเอกสาร XML ที่ออกแบบจัดเก็บลงในฐานข้อมูลเพื่อดูกลไกการทำงานในการจัดเก็บและการนำข้อมูลที่จัดเก็บออกมาใช้งาน

3. รวบรวมเอกสารที่มีโครงสร้างตามที่ออกแบบ

3.1. รวบรวมเอกสารบทความจากที่ต่าง ๆ เพื่อนำมาทดลอง โดยจะเน้นไปที่บทความภายในประเทศซึ่งทำให้ได้บทความภาษาไทยเป็นส่วนใหญ่

3.2. นำบทความที่รวบรวมได้ให้ผู้เชี่ยวชาญจัดเรียงลำดับคำดัชนี

3.3. นำบทความที่ได้ซึ่งอยู่ในรูปของไฟล์เอกสารเปลี่ยนให้อยู่ในรูปแบบของไฟล์เอกสาร XML ที่ได้ออกแบบไว้

4. ค้นคว้าวิจัยวิธีการจัดทำดัชนี

4.1. ศึกษาการให้ค่าน้ำหนักคำดัชนีด้วยวิธีเวคเตอร์โมเดล

4.2. ศึกษาสมการในการนำค่าน้ำหนักของแท็กมาร่วมคำนวณในการหาค่าน้ำหนักคำดัชนีโดยวิธีเวคเตอร์โมเดล

4.3. ออกแบบการนำจำนวนของคำในแท็กเข้ามาคำนวณเป็นค่าน้ำหนักแท็กและนำไปร่วมคำนวณในการหาค่าน้ำหนักคำดัชนีโดยวิธีเวคเตอร์โมเดล

5. วิจัยหาวิธีการนำเจนิติกอัลกอริทึมมาประยุกต์ใช้กับปัญหา

5.1. ศึกษากระบวนการเจนิติกอัลกอริทึมว่าทำงานอย่างไร

5.2. ออกแบบฟังก์ชันความเหมาะสมด้วยการวัดจากเปอร์เซ็นต์ความตรงกันของการเปรียบเทียบการจัดเรียงลำดับคำดัชนีที่จัดทำโดยเครื่องกับผู้เชี่ยวชาญ

5.3. ออกแบบโครโมโซมที่ใช้ในการทดลองหาค่าน้ำหนักแท็ก

6. ทดลองหาค่าน้ำหนักแท็กที่เหมาะสมกับโครงสร้างเอกสารที่ออกแบบ

7. วัดประสิทธิภาพวิธีคำนวณแต่ละแบบแล้วนำมาเปรียบเทียบกันหาวิธีที่มีประสิทธิภาพดีที่สุด

1.7 รายละเอียดในแต่ละบท

ในวิทยานิพนธ์ฉบับนี้แบ่งเนื้อหาเป็น 7 บท ดังนี้

- บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ แนวความคิด ขอบเขตของงานวิจัย และขั้นตอนการศึกษา
- บทที่ 2 กล่าวถึง XML และโครงสร้างเอกสาร XML โดยกล่าวถึงการเขียน DTD การกำหนดค่า Element และ Attribute การเขียน XML Schema และกล่าวถึงข้อแตกต่างระหว่าง DTD และ XML Schema ไว้ในส่วนสุดท้าย

- บทที่ 3 กล่าวถึงโมเดลระบบสารสนเทศที่นิยมใช้กันอยู่ คือ บูลลีนโมเดล และเวคเตอร์โมเดล โดยกล่าวถึงนิยามของโมเดล การหาค่าความเหมือนของคำค้นกับเอกสาร และการหาค่าน้ำหนักของคำดัชนีสำหรับเวคเตอร์โมเดล
- บทที่ 4 กล่าวถึงการประมวลผลเอกสารเริ่มจากการจัดเตรียมเอกสาร XML , ขั้นตอนการประมวลผลคำในเอกสาร XML จนได้คำหรือกลุ่มคำที่แยกจากกัน จากนั้นจะกล่าวถึงการหาค่าน้ำหนักคำดัชนีโดยใช้ค่าน้ำหนักแท็กโดยกล่าถึงการหาค่าน้ำหนักแท็กด้วยวิธีการต่าง ๆ โดยเริ่มจากการคำนวณค่าน้ำหนักด้วยวิธีเวคเตอร์โมเดลซึ่งเป็นวิธีที่ใช้อยู่ในปัจจุบัน หลังจากนั้นกล่าวถึงการปรับปรุงโดยเพิ่มน้ำหนักแท็กเข้ามาในสูตรการคำนวณด้วยวิธีเวคเตอร์โมเดลซึ่งมี 2 วิธีที่นำเสนอ หลังจากนั้นจะกล่าวถึงการนำความยาวแท็กมาใช้เป็นค่าน้ำหนักร่วมกับสูตรที่ได้ปรับปรุง
- บทที่ 5 กล่าวถึงการกำหนดค่าน้ำหนักแท็กด้วยวิธีต่าง ๆ เช่น การกำหนดโดยผู้ใช้ การกำหนดโดยใช้จำนวนคำในแท็ก และสุดท้ายกำหนดโดยใช้เจนิติกอัลกอริทึม ซึ่งในส่วนนี้จะกล่าวถึงการนำเจนิติกอัลกอริทึมมาใช้ในงานวิจัย โดยกล่าวถึงที่มาของการจำเจนิติกอัลกอริทึมมาใช้ หลังจากนั้นกล่าวถึง การกำหนดฟังก์ชันเป้าหมายหรือฟังก์ชันความเหมาะสมให้กับปัญหาที่สนใจ, รูปแบบโครโมโซมที่ใช้ในเจนิติกอัลกอริทึม, วัฏจักรการทำงานของเจนิติกอัลกอริทึมซึ่งแสดงขั้นตอนการคิดว่าเริ่มต้นอย่างไรและทำงานอย่างไร พร้อมแสดงตัวอย่างประกอบในแต่ละขั้นตอน และหาผลการทดลองหาค่าน้ำหนักแท็กด้วยวิธีต่าง ๆ จากการเรียนรู้ด้วยเจนิติกอัลกอริทึม
- บทที่ 6 กล่าวถึงวิธีการวัดประสิทธิภาพของระบบค้นคืนสารสนเทศด้วยตัววัดต่าง ๆ และผลการทดลองและการเปรียบเทียบวิธีการคำนวณน้ำหนักคำดัชนีโดยใช้เครื่องมือวัดต่าง ๆ ที่นำเสนอในบทที่ 3 มาใช้เปรียบเทียบวิธีการคำนวณดัชนีวิธีต่าง ๆ ที่ได้นำเสนอ
- บทที่ 7 กล่าวถึงผลสรุปงานวิจัย และข้อเสนอแนะ

บทที่ 2
ภาษา XML และโรงสร้างภาษา

2.1 XML

XML ย่อมาจาก Extensible Markup Language คือ ภาษามาตรฐานใหม่ที่ใช้แทนเนื้อหาของเว็บแทนที่ ภาษา HTML (Hypertext Markup Language) โดย XML เป็นภาษาที่ใช้อธิบายข้อความในเอกสารเพื่อทำให้ข้อมูลนั้นมีโครงสร้าง โดยผู้ใช้สามารถกำหนดแท็กขึ้นมาใช้ได้เอง ในการกำหนดแท็กขึ้นมาใช้เองนั้นจะต้องมีการประกาศโครงสร้างของเอกสารขึ้นมาก่อนโดยใช้ DTD หรือ XML Schema [4]

2.2 ประเภทโครงสร้างเอกสาร XML

มีอยู่ 2 ประเภทคือ DTD และ XML Schema

2.2.1 DTD

DTD ย่อมาจาก Document Type Definition ซึ่งทำหน้าที่อธิบายไวยากรณ์ของเอกสาร XML ที่กำหนดขึ้น แต่ในปัจจุบันมีการใช้น้อยลงเนื่องจากมีข้อจำกัดมาก แต่ก็ยังมีใช้อยู่บ้างในภาษาเช่น WML และ แอพพริเคชั่นบางตัว เช่น DB2 XML Extender ที่ใช้ DTD ในการกำหนดโครงสร้างของ เอกสาร XML ที่จะใช้งาน

2.2.1.1 ลักษณะของเอกสาร DTD

ตัวอย่างด้านล่างแสดง DTD อย่างง่าย ๆ ของเอกสาร XML ที่ใช้เก็บข้อมูลบัญชีหนังสือ ซึ่งในบัญชีหนังสือหนึ่งจะมีหนังสือกี่เล่มก็ได้ หรือไม่มีก็ได้ ภายในหนังสือประกอบด้วย ชื่อเรื่อง, ชื่อผู้แต่ง, วันที่, เลข ISBN และสำนักพิมพ์ซึ่งทุกแท็กเป็นข้อความทั่วไป

```
<!ELEMENT BookCatalogue (Book)*>
<!ELEMENT Book (Title, Author, Date, ISBN, Publisher)>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Author (#PCDATA)>
<!ELEMENT Date (#PCDATA)>
<!ELEMENT ISBN (#PCDATA)>
<!ELEMENT Publisher (#PCDATA)>
```

จาก DTD ข้างต้นสามารถสร้างเอกสาร XML ที่เป็นไปตามข้อกำหนดข้างต้นได้ดังนี้

```
<?xml version="1.0"?>
<BookCatalogue>
        <Book>
                <Title>Modern  Information Retrieval</Title>
                <Author>Ricardo Baeza-Yates</Author>
                <Data>1999</Date>
                <ISBN>0-201-39829-9</ISBN>
                <Publisher>ACM Press</Publisher>
        </Book>
</BookCatalogue>
```

ซึ่งสามาถอธิบายเอกสาร XML ได้ดังนี้คือ เอกสาร XML นี้เก็บข้อมูล BookCatalogue ซึ่งภายในเป็นหนังสือชื่อ Modern Information Retrieval ผู้แต่งชื่อ Ricardo Baeza-Yates แต่งปี 1999 มีเลข ISBN คือ 0-201-39829-9 พิมพ์โดยสำนักพิมพ์ ACM Press

2.2.1.2 การประกาศ DTD ในเอกสาร XML

DTD นั้นจะประกาศไว้ในส่วนของ Document Type Declaration ซึ่งจะแทรก DTD ไว้ในเอกสาร XML ก็ได้ ซึ่งเรียกว่าการประกาศแบบ Internal DTD หรือจะแยกเป็นไฟล์ต่างหากก็ได้ (มีนามสกุล .dtd) ซึ่งเรียกว่าการประกาศแบบ External DTD

ตัวอย่างการประกาศ DTD ทั้ง 2 แบบแสดงได้ดังนี้

- **การประกาศแบบ Internal DTD**

```
<?xml version="1.0"?>
<!DOCTYPE BookCatalogue [
   <!ELEMENT BookCatalogue (Book)*>
   <!ELEMENT Book (Title, Author, Date, ISBN, Publisher)>
   <!ELEMENT Title (#PCDATA)>
   <!ELEMENT Author (#PCDATA)>
   <!ELEMENT Date (#PCDATA)>
   <!ELEMENT ISBN (#PCDATA)>
   <!ELEMENT Publisher (#PCDATA)>
]>
```

```
<BookCatalogue>
        <Book>
                <Title>Modern  Information Retrieval</Title>
                <Author>Ricardo Baeza-Yates</Author>
                <Data>1999</Date>
                <ISBN>0-201-39829-9</ISBN>
                <Publisher>ACM Press</Publisher>
        </Book>
</BookCatalogue>
```

ตัวหนังสือที่เป็นตัวเข้มแสดงส่วนการประกาศ Internal DTD

- การประกาศแบบ **Internal DTD**

```
<?xml version="1.0"?>
<!DOCTYPE BookCatalogue SYSTEM “bookcatalogue.dtd”>
<BookCatalogue>
        <Book>
                <Title>Modern  Information Retrieval</Title>
                <Author>Ricardo Baeza-Yates</Author>
                <Data>1999</Date>
                <ISBN>0-201-39829-9</ISBN>
                <Publisher>ACM Press</Publisher>
        </Book>
</BookCatalogue>
```

ตัวหนังสือที่เป็นตัวเข้มแสดงส่วนการประกาศ External DTD

2.2.1.3 การสร้าง DTD

การสร้าง DTD แบ่งออกเป็น 3 หัวข้อหลัก ๆ คือ การประกาศค่าอีลิเมนต์, การประกาศค่าแอตทริบิวต์ และการประกาศค่าเอ็นทิตี้

2.2.1.3.1 การประกาศค่าอีลิเมนต์

ในวิทยานิพนธ์ใช้คำว่า "แท็ก" ซึ่งกล่าวโดยมีความหมายเดียวกับคำว่า "อีลิเมนต์" เพื่อความสะดวกในการกล่าวถึง จึงขอใช้คำว่า "แท็ก" แทนคำว่า "อีลิเมนต์" ในบทอื่นๆ นอกจากบทนี้ รูปแบบการประกาศค่าอีลิเมนต์เป็นดังนี้

```
<!ELEMENT ชื่ออีลิเมนต์ (เนื้อหาภายในอีลิเมนต์)>
```

เนื้อหาภายในอีลิเมนต์หนึ่ง ๆ อาจเป็นได้ 3 อย่างคือ อาจเป็นอีลิเมนต์อื่น ๆ, ข้อความปกติ หรืออาจจะไม่มีอะไรอยู่เลย (เป็นแท็กว่าง) หรือบางครั้งอาจจะมีอีลิเมนต์ปนอยู่กับข้อความปกติได้

ตัวอย่างแบบต่าง ๆ

- แบบมีอีลิเมนต์อื่น ๆ อยู่ภายใน เช่น

```
<Book>
        <Title>Modern  Information Retrieval</Title>
        <Author>Ricardo Baeza-Yates</Author>
</Book>
```

ในตัวอย่างนี้อีลิเมนต์ Book มีอีลิเมนต์ Title และ Author อยู่ภายใน

- แบบที่มีข้อความอยู่ภายใน เช่น

```
<Publisher>ACM Press</Publisher>
```

- แบบไม่มีอะไรอยู่ หรือ แท็กว่าง เช่น

```
<Book isbn="975-85421-9-6"/>
```

- แบบที่มีอีลิเมนต์ปนอยู่กับข้อความ เรียกว่า "Mixed content" เช่น

```
<Message> Hello my XML
        <Note>See again<Note>
</Message>
```

ต่อไปจะกล่าวถึงวิธีประกาศอีลิเมนต์ใน DTD

— **กรณีที่ภายในอีลิเมนต์นั้นเป็นอะไรก็ได้ให้ระบุด้วย ANY**

กรณีนี้จะใช้เมื่อไม่บังคับว่าข้างในอีลิเมนต์เป็นอะไร นั้นคือจะมีอะไรก็ได้

```
<!ELEMENT ชื่ออีลิเมนต์ ANY>
```

— **ข้อความธรรดาให้ระบุด้วย PCDATA**

กรณีที่ต้องการบังคับให้เนื้อหาภายในอีลิเมนต์เป็นข้อความธรรดาเท่านั้น

```
<!ELEMENT ชื่ออีลิเมนต์ (#PCDATA)>
```

โดยที่ PCDATA ย่อมาจาก Parsed Character Data หมายความว่าเนื้อหาส่วนที่เป็นข้อความนี้ เอาไว้ให้โปรแกรม XML Parser อ่านเพื่อประมวลผล

— **อีลิเมนต์ว่าง ๆ ระบุด้วย EMPTY**

กรณีของอีลิเมนต์ว่าง ให้ประกาศตามรูปแบบดังนี้

```
<!ELEMENT ชื่ออีลิเมนต์ EMPTY>
```

— **การใช้ลิสต์ลำดับและตัวเลือก**

การประกาศแบบเรียงลำดับจะใช้ (,) (คอมมา) เพื่อแสดงลำดับของอีลิเมนต์ และมีอย่างละ 1 อีลิเมนต์เท่านั้น และจะต้องมีเสมอ รูปแบบการประกาศเป็นดังนี้

```
<!ELEMENT ชื่ออีลิเมนต์ (อีลิเมนต์ 1, อีลิเมนต์ 2, อีลิเมนต์ 3)>
```

การประกาศแบบตัวเลือกจะใช้ (|) (ไปป์) คั่นระหว่างอีลิเมนต์ หมายความว่า ให้เลือกเพียงอีลิเมนต์ใดอีลิเมนต์หนึ่งเท่านั้น รูปแบบการประกาศเป็นดังนี้

```
<!ELEMENT ชื่ออิลิเมนต์ (อิลิเมนต์ 1 | อิลิเมนต์ 2 | อิลิเมนต์ 3)>
```

การประกาศเมื่อภายในอีลิเมนต์นั้นมีทั้งเนื้อหาและข้อความ รูปแบบการประกาศเป็นดังนี้

```
<!ELEMENT ชื่ออิลิเมนต์ (#PCDATA, อิลิเมนต์ 1, อิลิเมนต์ 2 )>
```

— **การประกาศอีลิเมนต์ โดยระบุจำนวน**

ใน XML มีเครื่องหมายพิเศษ 3 แบบ คือ +, * และ ? ใช้เพื่อแสดงว่าอีลิเมนต์นั้นจะปรากฏได้กี่ครั้ง โดยแต่ละอันมีความหมายดังนี้

\+ หมายถึง จะต้องมีอีลิเมนต์นั้นตั้งแต่ 1 อีลิเมนต์ขึ้นไป (คือ 1, 2, 3)

* หมายถึง จะมีอีลิเมนต์กี่อีลิเมนต์ก็ได้ หรือไม่มีก็ได้ (คือ 0, 1, 2)

? หมายถึง จะต้องมีอีลิเมนต์นั้นเพียง 1 อีลิเมนต์เท่านั้น หรือไม่มีก็ได้ (คือ 0, 1)

2.2.1.3.2 การประกาศค่าแอตทริบิวต์

การประกาศแอตทริบิวต์มีรูปแบบมาตรฐานดังนี้

```
<!ATTLIST ชื่ออิลิเมนต์ ชื่อแอตทริบิวต์ ชนิดข้อมูลของแอตทริบิวต์ (#REQUIRED |
#IMPLIED | #FIXED) ค่าปกติของแอตทริบิวต์>
```

ตัวอย่าง :

```
<!ATTLIST Document lang CDATA  #FIXED "TH" >
```

Document คือ ชื่ออีลิเมนต์

lang คือ ชื่อแอตทริบิวต์

Document lang คือ การประกาศแอตทริบิวต์ชื่อ lang ซึ่งอยู่ภายในอีลิเมนต์ Document

CDATA คือ การประกาศชนิดของค่าแอตทริบิวต์ ว่าเป็นข้อมูลประเภทตัวอักษร

#FIXED "TH" บอกให้ทราบว่า กำหนดค่าแอตทริบิวต์ไว้แน่นอนด้วยคำว่า "TH"

2.2.1.3.3 การประกาศค่าเอ็นทิตี้

เอ็นทิตี้ จะใช้สำหรับอ้างถึง "ทรัพยากร" หรือข้อมูลที่นิยามไว้ก่อน ทำให้สามารถนำข้อมูลนั้นกลับมาใช้ใหม่โดยไม่ต้องเขียนบ่อย ๆ การประกาศเอ็นทิตี้มีรูปแบบดังนี้

```
<!ENTITY ชื่อเอ็นทิตี้ ทรัพยากร >
```

ส่วนใหญ่แล้วทรัพยากรหรือข้อมูล ก็คือ ข้อความ (สตริง) หรือไฟล์

เอ็นทิตี้ในภาษา XML มี 2 ประเภทใหญ่ ๆ คือ

- General Entity
- Parameter Entity เป็น เอ็นทิตี้ประเภทที่ค่อนข้างซับซ้อนพอสมควรและแทบจะไม่ได้ใช้ จึงไม่ขอกล่าวในที่นี้

General Entity แบ่งออกได้เป็น 2 ประเภทย่อย คือ Parsed Entity และ Unparsed Entity โดย Parsed Entity เป็นเอ็นทิตี้ที่ XML Parser อ่านหรือแปลได้ ซึ่งก็คือข้อความประเภทตัวอักษร

- Parsed Entity ใช้กับข้อมูลประเภทข้อความ
- Unparsed Entity คือข้อมูลที่ XML Parser อ่านไม่ได้หรือแปลไม่ได้ เช่น ข้อมูลประเภทรูปภาพ ซึ่งเป็นไฟล์จำพวกไบนารี่

Parsed Entity จำแนกตามที่อยู่ของข้อมูลได้ 2 แบบ ได้แก่ แบบ Internal และแบบ External

- Internal คือประกาศเอ็นทิตี้นั้นไว้ใน DTD เลย
- External คือข้อมูลแยกไปอยู่อีกไฟล์หนึ่งต่างหาก ซึ่งมีรูปแบบการประกาศดังนี้

```
<!ENTITY ชื่อเอ็นทิตี้ SYSTEM "ชื่อไฟล์และพาธ" >
```

Unparsed Entity ใช้กับข้อมูลที่ไม่ใช่ข้อความ

ข้อมูลที่เป็น Parsed Entity มี 2 แบบ คือ Internal และ External แต่สำหรับ Unparsed Entity มีเฉพาะ External อย่างเดียว เพราะข้อมูลที่อ้างถึงนั้นอยู่คนละไฟล์กับเอกสาร XML

โดยการประกาศใน DTD จะต้องมี 2 ขั้นตอนหลัก ๆ คือ

1. ประกาศโนเทชัน (NOTATION) เพื่อบอกชนิดข้อมูลก่อน

2. ประกาศเอ็นทิตี้

การประกาศโนเทชัน มีรูปแบบดังนี้

<!NOTATION ชื่อโนเทชัน SYSTEM "External_ID" >

External_ID คือโปรแกรมภายนอกที่จะมาจัดการกับข้อมูลนั้น ซึ่งส่วนใหญ่จะระบุเป็น MIME type แทน เช่น ไฟล์รูปภาพ ก็ระบุเป็น image/jpg หรือ image/gif ขึ้นอยู่กับว่ารูปภาพนั้นเป็นไฟล์ประเภทใด ตัวอย่างเช่น ประกาศว่า <!NOTATION jpg SYSTEM "image/jpg" >

ต่อไปเป็นการประกาศเอ็นทิตี้ ซึ่งมีรูปแบบพิเศษออกไป ซึ่งประกาศดังนี้

<!ENTITY ชื่อเอ็นทิตี้ SYSTEM "ชื่อไฟล์และพาธ" NDATA ชื่อโนเทชัน >

ส่วนที่เพิ่มขึ้นมาคือคีย์เวิร์ด NDATA และชื่อโนเทชัน เพื่อเชื่อมโยงไปยังโนเทชันที่ประกาศไว้ก่อนแล้ว

2.2.2 XML Schema

XML Schema เป็นวิธีที่ใช้ในการกำหนดโครงสร้างเอกสาร XML และเป็นวิธีใหม่กว่า DTD มาก เพราะ DTD มีใช้มาตั้งแต่ภาษา SGML และ XML Schema ได้กลายมาเป็นมาตรฐานของ W3C เอกสาร XML จะถูกตรวจความถูกต้องโดยใช้ XML parser ร่วมกับ XML Schema XML Schemaทำงานเหมือนไวยากรณ์ของเอกสารที่ใช้เพื่อให้ความหมายกับเอกสารที่มีโครงสร้าง

2.2.2.1 ลักษณะของ XML Schema

การอธิบายจะอธิบายโดยการเปรียบเทียบ DTD กับ XML Schema ของเอกสารเดียวกัน ซึ่งจะทำให้เห็นความแตกต่างได้อย่างชัดเจน ตัวอย่าง DTD ของบัญชีหนังสือมีลักษณะดังนี้

```
<!ELEMENT BookCatalogue (Book)*>
<!ELEMENT Book (Title, Author, Date, ISBN, Publisher)>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Author (#PCDATA)>
<!ELEMENT Date (#PCDATA)>
<!ELEMENT ISBN (#PCDATA)>
<!ELEMENT Publisher (#PCDATA)>
```

เมื่อเปลี่ยนมาใช้ XML Schema จะมีลักษณะดังนี้

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
targetNamespace="http:///www.publishing.org"  xmlns="http://www.publishing.org"
elementFormDefault="qualified">
<xsd:element name="BookCatalogue">
        <xsd:complexType> <xsd:sequence>
                <xsd:element ref="Book" minOccurs="0" maxOccurs="unbounded"/>
        </xsd:sequence> </xsd:complexType>
</xsd:element>
<xsd:element name="Book">
        <xsd:complexType> <xsd:sequence>
                <xsd:element ref="Title" minOccurs="1" maxOccurs="1"/>
                <xsd:element ref="Author" minOccurs="1" maxOccurs="1"/>
                <xsd:element ref="Date" minOccurs="1" maxOccurs="1"/>
                <xsd:element ref="ISBN" minOccurs="1" maxOccurs="1"/>
                <xsd:element ref="Publisher" minOccurs="1" maxOccurs="1"/>
</xsd:sequence> </xsd:complexType>
</xsd:element>
<xsd:element name="Title" type="xsd:string"/>
<xsd:element name="Author" type="xsd:string"/>
<xsd:element name="Date" type="xsd:string"/>
<xsd:element name="ISBN" type="xsd:string"/>
<xsd:element name="Publisher" type="xsd:string"/>
</xsd:schema>
```

ต่อไปจะกล่าวถึงอีลิเมนต์พื้นฐานแบบ Simple Type และแบบ Complex Type

2.2.2.2 Simple Type และ Complex Type

ใน XML Schema แบ่งอีลิเมนต์ออกเป็น 2 แบบคือ แบบ Simple Type และ Complex Type

- Simple Type คือ อีลิเมนต์ที่มีข้อมูลภายในเป็นข้อมูลพื้นฐาน เช่น สตริง ตัวเลข เป็นต้น
- Complex Type คือ อีลิเมนต์ที่มีแอตทริบิวต์หรือมีข้อมูลภายในเป็นอีลิเมนต์อื่น ๆ

- **ชนิดข้อมูลแบบ Simple Type**

Simple Type คือ อีลิเมนต์ที่มีข้อมูลภายในเป็นสตริง หรือตัวเลข หรือข้อมูลพื้นฐานอื่น ๆ นอกจากนี้ยังประกอบด้วยข้อมูลพื้นฐานอื่น ๆ อีก

ชนิดข้อมูลของ Simple Type แบ่งออกเป็น 2 กลุ่มคือ

1. Primitive datatype คือ ชนิดข้อมูลพื้นฐาน เช่น string, boolean, float, double เป็นต้น
2. Derived datatype คือ ชนิดข้อมูลที่ขยายมาจาก primitive datatype อีกที

การประกาศ XML Schema สำหรับ อีลิเมนต์ที่เป็น Simple Type มีรูปแบบพื้นฐานดังนี้

```
<element name="ชื่ออีลิเมนต์" type="ชนิดของข้อมูล">
```

- **ชนิดข้อมูลแบบ Complex Type**

Complex Type คือ อีลิเมนต์ที่บรรจุอีลิเมนต์อื่นไว้ภายใน หรือเป็นอีลิเมนต์ที่มีแอตทริบิวต์ประกอบอยู่ด้วย

ตัวอย่าง : การประกาศอีลิเมนต์แบบ Complex Type ใน XML Schema

```
1:    <xsd:complexType name="USAddress"
2:          <xsd:sequence>
3:                <xsd:element name="name" type="xsd:string"/>
4:                <xsd:element name="street" type="xsd:string"/>
5:                <xsd:element name="city" type="xsd:string"/>
6:                <xsd:element name="state" type="xsd:string"/>
7:          <xsd:element name="zip" type="xsd:decimal"/>
8:    </xsd:sequence>
```

XML Schema ในตัวอย่างข้างต้นกำหนดอีลิเมนต์ประเภท USAddress ขึ้นมาเป็นแบบ Complex Type (บรรทัดที่ 1) ซึ่งประกอบไปด้วย อีลิเมนต์ name, street, city, state, zip (บรรทัดที่ 3 ถึงบรรทัดที่ 7) และมีแอตทริบิวต์ country ด้วย (บรรทัดที่ 9) นอกจากนี้ยังระบุชนิดข้อมูลของแต่ละ

อีลิเมนต์ ซึ่งส่วนใหญ่เป็นข้อมูลชนิดสตริง ยกเว้น zip ที่เป็นตัวเลขทศนิยม (decimal) และ country เป็นชนิด NMTOKEN ซึ่งก็คือสตริงประเภทที่ไม่มีช่องว่าง

2.2.2.3 เนมสเปซ

ภาษา XML อาศัยข้อกำหนดที่สำคัญอย่างหนึ่ง ซึ่งเรียกว่า เนมสเปซ (namespace) เพื่อใช้สำหรับป้องกันความสับสนในการระบุชื่อองค์ประกอบใด ๆ ภายในเอกสาร XML คนละไฟล์ แต่ใช้ชื่อเดียวกัน เนื่องจากอาจเป็นไปได้ว่าเอกสาร XML ที่ใช้ในงานที่ต่างกัน มีชื่ออีลิเมนต์เหมือนกัน ชื่อชนิดข้อมูลเหมือนกัน แล้วอาจจะถูกเรียกใช้ภายในเอกสาร XML ไฟล์เดียวกัน การประกาศ prefix และ เนมสเปซ จะอาศัยแอตทริบิวต์ xmlns (ย่อมาจาก XML namespace) ตามรูปแบบดังนี้

```
<xmlns:prefix="ชื่อเนมสเปซ">
```

2.3 การเปรียบเทียบระหว่าง XML Schema กับ DTD

XML Schema มีข้อดีกว่า DTD หลายด้านดังแสดงไว้ในตารางที่ 2.1 ดังต่อไปนี้

ตารางที่ 2.1 แสดงให้เห็นความแตกต่างระหว่างการใช้งาน DTD และ XML schema

ข้อเปรียบเทียบ	DTD	XML schema
ไวยากรณ์	EBNF ซึ่งมีโครงสร้างไม่เหมือนกับภาษา XML	XML 1.0 ทำให้มีโครงสร้างการเขียนเหมือนกับภาษา XML
เครื่องมือในการใช้งาน	มีให้เลือกใช้มาก(ใช้เครื่องมือ SGML ที่มีอยู่) แต่เครื่องมือเหล่านี้มีราคาแพง	ใช้เครื่องมือ XML ที่มีอยู่ในปัจจุบันได้เกือบทั้งหมด รวมถึง DOM, XSLT และเบราเซอร์ที่รู้จักภาษา XML
การใช้โมเดลชนิดข้อมูล	เสียเปรียบเนื่องจาก - มีเพียงลิสต์ตัวเลือก หรือลิสต์ลำดับอย่างง่าย ไม่สามารถใช้โมเดลชนิดข้อมูลผสมกันได้ - ระบุจำนวนครั้งที่ปรากฏอีลิเมนต์ได้แค่ 0,1 หรือ หลายครั้งเท่านั้น - ไม่สามาถกำหนดชื่อกลุ่มอีลิเมนต์หรือ แอตทริบิวต์ได้	ได้เปรียบเนื่องจาก - โมเดลชนิดข้อมูลมีรายละเอียดมากกว่าและสามารถใช้โมเดลชนิดข้อมูลผสมกันได้ - สามารถกำหนดจำนวนครั้งที่ปรากฏขึ้นจริงได้ - สามาถกำหนดชื่อกลุ่มอีลิเมนต์หรือ แอตทริบิวต์ได้

ตารางที่ 2.1 (ต่อ)

ข้อเปรียบเทียบ	DTD	XML schema
การกำหนดชนิดข้อมูล	เสียเปรียบเนื่องจากรองรับข้อมูลไม่กี่ชนิดเช่น สตริง, เนมโทเคน, ID และอื่น ๆ อีกเล็กน้อย	ได้เปรียบเนื่องจากรองรับชนิดข้อมูลสมัยใหม่ที่นิยมใช้ได้แก่: สตริง, ตัวเลข, วัน/เวลา และโครงสร้าง ซึ่งสามารถนำไปจัดเก็บไว้ในฐานข้อมูลได้โดยง่าย นอกจากนี้ยังสามารถสร้างชนิดข้อมูลใหม่ที่ยิงจากชนิดข้อมูลเดิมที่มีอยู่ได้
การขยายความ สามารถ	มีข้อจำกัดเนื่องจากต้องแก้ไขข้อแนะนำ XML 1.0 (และอาจรวมถึง SGML) เพื่อขยับขยายความสามารถใหม่ ๆ	ไม่จำกัดเนื่องจากขึ้นอยู่กับข้อกำหนดภาษา XML และความสามารถในการขยายขนาดของ XML Schema ยังรวมถึงคุณสมบัติเฉพาะในการทำ Internationalization ได้
ข้อจำกัดในการสืบทอด	มีข้อจำกัดเนื่องจาก DTD ต้องใช้งานย้อนหลังร่วมกับ SGML ที่ยังคงมีใช้อยู่ได้	ไม่มีข้อจำกัดนี้เนื่องจาก XML Schema จะขึ้นอยู่กับเทคโนโลยีและภาษาโปรแกรมสมัยใหม่รวมทั้งรองรับ Object Oriented ด้วย
โครงสร้างเอกสารเปลี่ยนแปลงได้	ไม่สามารถทำได้ในขณะรันไทม์	สามารถเปลี่ยนแปลงได้ XML schema สามารถถูกเลือกใช้และแก้ไขได้ขณะรันไทม์ ซึ่งบางครั้งเกิดจากการติดต่อกับผู้ใช้

การใช้ไวยากรณ์ภาษาโครงสร้างในงานวิจัยนี้ เลือกใช้ DTD เนื่องจากข้อกำหนดในการใช้งานโปรแกรมที่ใช้งานร่วมกับฐานข้อมูล [5,6] จำเป็นต้องใช้การประกาศโครงสร้างของเอกสาร XML แบบ DTD เพื่อที่จะนำโครงสร้างนั้นไปใช้ตรวจสอบความถูกต้องของเอกสาร และใช้สร้างตารางที่เกี่ยวข้องกับเอกสาร เนื่องจากโปรแกรมนี้ถูกพัฒนาขึ้นมาก่อนการประกาศใช้ XML Schema อย่างเป็นทางการของ W3C แต่ด้วยข้อดีของ XML Schema โปรแกรมประยุกต์ในรุ่นต่อไปที่ทำงานร่วมกับเอกสาร XML ควรที่จะรองรับการทำงานร่วมกับ XML Schema ได้ เพิ่มเติมจาก DTD ที่ใช้อยู่ในปัจจุบัน

บทที่ 3
โมเดลและการประเมินประสิทธิภาพระบบค้นคืนสารสนเทศ

3.1 โมเดลระบบค้นคืนสารสนเทศ

ในงานวิจัยเกี่ยวกับการค้นคืนสารสนเทศ (Information Retrieval) นั้นเอกสารต่าง ๆ ที่จะนำมาประมวลผลจะต้องจัดเก็บให้อยู่ในรูปแบบต่าง ๆ ซึ่งเรียกว่า IR Model ปัจจุบันมีการคิดค้นโมเดลต่าง ๆ มาสนับสนุนมากมาย โดยส่วนมากจะถูกจัดให้อยู่ในแบบแผนของโมเดลดังนี้ คือ บูลลีนโมเดล (Boolean model) เวคเตอร์โมเดล (Vector model) และแบบจำลองเชิงความน่าจะเป็น (Probabilistic model) ซึ่งจะกล่าวถึงในหัวข้อต่อไป

3.1.1 บูลลีนโมเดล

บูลลีนโมเดล (Boolean model) เป็นโมเดลในระบบ IR ที่ทำงานอยู่บนพื้นฐานของเซตและพีชคณิต (Boolean algebra) ซึ่งเป็นโมเดลที่ง่ายต่อการทำความเข้าใจและง่ายต่อการนำมาใช้งาน ด้วยคุณสมบัติดังกล่าวทำให้บูลลีนโมเดลได้รับความนิยมในเชิงพาณิชย์เป็นอย่างมาก แต่ประสิทธิภาพของโมเดลนี้ไม่ค่อยดีมากนักดังสรุปได้ดังนี้

1. การได้มาซึ่งข้อมูลจะอยู่บนพื้นฐานของการตัดสินใจแบบไบนารี (Binary decision) กล่าวคือ เอกสารจะถูกจัดอยู่ในกลุ่มเอกสารที่สัมพันธ์กันหรือไม่สัมพันธ์กันอย่างใดอย่างหนึ่ง โดยไม่มีการแบ่งลำดับชั้นย่อยเลย ทำให้ไม่สามารถใช้เป็นข้อมูลในการจัดอันดับเอกสารที่สืบค้นได้
2. ในบางครั้งการที่จะแปลเอาความต้องการมาของผู้ใช้ให้อยู่ในรูปแบบของบูลีนนั้นสามารถทำได้ยาก

บูลลีนโมเดลนั้นจะพิจารณาถึงตัวอินเด็กซ์เทอม (Index term) ว่าปรากฏอยู่ในเอกสารหรือไม่ นั่นคือ น้ำหนักของเทอมจะมีค่าเป็น 0,1 เท่านั้น $w_{i,j} \in \{0,1\}$ คิวรี่นั้นจะประกอบไปด้วยอินเด็กซ์เทอมที่เชื่อมด้วย and, or หรือ not ดังนั้นจึงสามารถแทนคิวรี่ได้ในรูป Disjunction of conjunction vector (Disjunction normal form : DNF) เช่น $[q = k_a \wedge (k_b \vee \neg k_c)]$ สามารถเขียนให้อยู่ในรูป DNF ได้เป็น $[\vec{q}_{dnf} = (1,1,1) \vee (1,1,0) \vee (1,0.0)]$ เมื่อแต่ละองค์ประกอบของ Binary weight vector แทนด้วยทัปเปิล (k_a, k_b, k_c)

นิยามที่ 3.1 ระบบ IR model แทนด้วย quadruple คือ $[D, Q, F, R(q_i, d_j)]$ เมื่อ

(1) D คือ เซตของกลุ่มเอกสารทั้งหมดในระบบ

(2) Q คือ เซตของคิวรี่ (Query) ที่ป้อนโดยผู้ใช้

(3) F คือ รูปแบบของโมเดลที่ใช้ในระบบ

(4) $R(q_i, d_j)$ คือ ฟังก์ชันที่ใช้จัดอันดับผลการค้นคืนเอกสาร $d_j \in D$ ที่สัมพันธ์กับคิวรี่ $q_i \in Q$ ที่เรียงตามลำดับความสัมพันธ์ที่มีต่อกัน

เมื่อเอกสาร d_j ถูกแทนด้วยอินเด็กซ์เทอม (Index terms) ดังนั้น D ก็คือเซตของทุกอินเด็กซ์เทอมที่มีอยู่ใน d_j กำหนดให้ k_i เป็นอินเด็กซ์เทอมและ d_j เป็นเอกสาร $d_j \in D$ ดังนั้น $w_{i,j} \geq 0$ คือค่าน้ำหนักของความสัมพันธ์ระหว่าง k_i ที่มีต่อ d_j

นิยามที่ 3.2 กำหนดให้ t คือจำนวนอินเด็กซ์เทอมทั้งหมดในระบบ, $t \in D$, และ $K = \{k_1, ..., k_i\}$ ถ้าหากค่าน้ำหนัก $w_{i,j} > 0$ แสดงว่าอินเด็กซ์เทอม k_i มีความสัมพันธ์ต่อเอกสาร d_j

นิยามที่ 3.3 ให้น้ำหนักของอินเด็กซ์เทอมเป็นแบบไบนารี $w_{i,j} \in \{0,1\}$ และคิวรี่ q ถูกแทนด้วยเวคเตอร์ $\vec{q}_{dnf}$ ในรูปของ Disjunctive Normal Form : DNF และให้เวคเตอร์ $\vec{q}_{cc}$ คือองค์ประกอบที่อยู่ใน $\vec{q}_{dnf}$ ดังนั้นสามารถคำนวณค่า Similarity ระหว่างเอกสาร d_j กับคิวรี่ q ได้คือ

$$sim(d_j, q) = \begin{cases} 1 \text{ if } \exists \vec{q}_{cc} \left| \left(\vec{q}_{cc} \in \vec{q}_{dnf}\right) \wedge \left(\forall k_i, g\left(\vec{d}_j\right) = g_i\left(\vec{q}_{cc}\right)\right) \right. \\ 0 \qquad \text{otherwise} \end{cases}$$

ถ้าค่า $sim(d_j, q) = 1$ แล้วบูลลีนโมเดลจะทำนายว่าเอกสาร d_j จะมีความเกี่ยวข้อง (Relevant) กับคิวรี่ q ถ้านอกเหนือไปจากนี้จะสรุปว่าเอกสาร d_j ไม่มีความเกี่ยวข้องกับคิวรี่ q

บูลลีนโมเดลมีความสามารถในการทำนายได้ค่าว่าแต่ละเอกสาร Relevant หรือ non-relevant เท่านั้น ไม่สามารถระบุความเกี่ยวข้องแบบ partial matching ได้

<u>**ตัวอย่างที่ 3.1**</u>: เอกสาร d_j มี k_b เป็นอินเด็กซ์เทอมแทนด้วย $d_j = (0,1,0)$ ดังนั้นเมื่อใช้ บูลลีนโมเดลหาความเกี่ยวข้องของเอกสาร d_j กับคิวรี่ $[q = k_a \wedge (k_b \vee \neg k_c)]$ จะได้ผลว่าไม่มีความเกี่ยวข้องกันเนื่องจาก $q = 0 \wedge (1 \vee \neg 0) = 0$

3.1.2 เวคเตอร์โมเดล

เวคเตอร์โมเดลเป็นการนำเสนอเอกสารและคำค้นโดยที่เอกสารและคำค้นถูกเปลี่ยนให้อยู่ในรูปของเวคเตอร์ ซึ่งสิ่งที่เก็บอยู่ในเวคเตอร์จะเป็นคำในเอกสารหรือคำค้น ซึ่งคำเหล่านี้ได้รับหลังจากหารากคำ และ ตัดคำ โดยที่เวคเตอร์เหล่านี้จะมีการให้น้ำหนักของเทอมแต่ละเทอมเพื่อเน้นข้อมูลในเอกสารหรือคำค้นที่มันนำเสนอ ในการค้นคืนจะนำเวคเตอร์ค้นคืนมาเปรียบเทียบกับ

เวคเตอร์เอกสารทุกเวคเตอร์ ถ้าเวคเตอร์ค้นคืนใดเหมือนกับเวคเตอร์เอกสารใดมากที่สุด เอกสารนั้นจะเป็นคำตอบของคำค้นนั้น

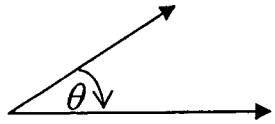
เอกสารที่พิจารณาจะถูกจัดให้อยู่ในรูปเวคเตอร์เรียกว่า Vector Model โดยคุณสมบัติของโมเดลนี้จะมีประสิทธภาพดีกว่าบูลลีนโมเดลเนื่องจากมีความสามารถทำ Partial matching ได้ เพราะใช้ตัวเลขจำนวนจริงบวกแทนค่าน้ำหนักของเทอม ซึ่งต่างจากบูลลีนโมเดล ที่ใช้ค่าน้ำหนักของเทอมด้วยเลขไบนารี่, $w_{i,j} \in \{0,1\}$, เอกสารทั้งหมดที่อยู่ในฐานข้อมูลของระบบ (Collection) จะถูกแทนด้วยเวคเตอร์น้ำหนักของเทอม ดังนั้นทำให้สามารถคำนวณหาค่าความคล้าย (Similarity) ระหว่างเอกสาร d_j กับชุดคำค้นจากผู้ใช้ q (User query) ได้โดยแทน q ให้อยู่ในรูปเวคเตอร์เดียวกับ d_j ซึ่งค่าความคล้ายนี้จะเป็นตัวบ่งบอกว่าเอกสาร d_j คล้ายกับคิวรี่ q เพียงไรและมีประโยชน์ในการเรียงลำดับ (Ranking) เอกสารในการแสดงผลต่อไป

นิยามที่ 3.4 กำหนดให้ค่าน้ำหนัก $w_{i,j}$ คือค่าความสัมพันธ์ของคู่ลำดับ (k_i, d_j) แสดงความมีอิทธิพลของเทอม k_i ที่มีต่อเอกสาร d_j มีค่าเป็นจำนวนจริงบวกแบบ non-binary และกำหนดให้ค่าน้ำหนัก $w_{i,q}$ ค่าความสัมพันธ์ของคู่ลำดับ (k_i, q) แสดงความมีอิทธิพลของเทอม k_i ที่มีคิวรี่ q มีค่าเป็นจำนวนจริงบวก จะได้เวคเตอร์แทนเอกสาร $\vec{d}_j = (w_{1,j}, w_{2,j}, ..., w_{t,j})$ และเวคเตอร์แทนคิวรี่ $\vec{q}_j = (w_{1,q}, w_{2,q}, ..., w_{t,q})$ ตามลำดับ

ดังนั้นทั้งเอกสาร d_j และคิวรี่ q ต่างถูกแทนด้วยเวคเตอร์ในระบบ t-มิติ ดังแสดงในรูป 3.1 จากทฤษฎีของเวคเตอร์ทำให้สามารถคำนวณหา Similarity ได้จากโอเปอร์เรเตอร์ที่ทำบนเวคเตอร์ เช่น ใช้ฟังก์ชัน Cosine of angle

$$sim(d_j, q) = \frac{\vec{d}_j \bullet \vec{q}}{|\vec{d}_j| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2 \times \sum_{i=1}^{t} w_{i,q}^2}} \qquad (3.1)$$

เนื่องจาก $w_{i,j} \geq 0$ และ $w_{i,q} \geq 0$ ดังนั้นค่า $sim(d_j, q)$ จึงมีค่าอยู่ในช่วง 0 ถึง +1 ซึ่งสามารถทำนายความเกี่ยวข้องกันระหว่างเอกสาร d_j กับ q ได้จากค่า $sim(d_j, q)$ หรือเรียกค่านี้ว่า Degree of similarity



รูปที่ 3.1 แสดงการแทนคิวรี่และเอกสารให้อยู่ในรูปเวคเตอร์ t-มิติ

นิยามที่ 3.5 กำหนดให้ N คือจำนวนเอกสารทั้งหมดใน Collection และ n_i คือจำนวนของเอกสารที่มี k_i ปรากฏอยู่ ดังนั้น $freq_{i,j}$ คือความถี่ (ทางสถิติ) ของเทอม k_i ที่ปรากฏในเอกสาร d_i (จำนวนครั้งที่เทอม k_i ถูกกล่าวถึงในเอกสาร d_i) ดังนั้นค่า Normalized frequency, $f_{i,j}$, ของเทอม k_i ในเอกสาร d_i แสดงได้เป็น $f_{i,j} = \frac{freq_{i,j}}{\max_l freq_{i,j}}$ เมื่อ $\max_l freq_{l,j}$ คือค่าความถี่ของเทอมที่สูดที่สุดในเอกสาร d_i

นิยามที่ 3.6 ถ้าเทอม k_i ปรากฏอยู่ในทุกเอกสารใน Collection แล้ว เทอม k_i จะไม่มีอำนาจในการจำแนกเอกสาร d_i ออกจาก Collection ได้หรือถ้าเทอม k_i ปรากฏในหลาย ๆ เอกสารใน Collection แล้ว เทอม k_i จะมีความสำคัญต่อเอกสาร d_i น้อยลง เรียกค่านี้ว่า Inverse document frequency ของเทอม k_i $idf_i = \log(N/n_i)$ ดังนั้นค่า $w_{i,j}$ ในนิยาม 3.3 คำนวณได้จาก $w_{i,j} = f_{i,j} \times idf_i$ เมื่อ $f_{i,j}$ นิยามตามนิยามที่ 3.5 เรียกวิธีการคำนวณแบบนี้ว่า *tf-idf*

<u>**ตัวอย่างที่ 3.2:**</u> กำหนดให้ใน Collection ประกอบด้วยเอกสารจำนวน 10,000 ชุด และเอกสาร d_i มีเทอม A ปรากฏ 3 ครั้ง เทอม B ปรากฏ 2 ครั้ง และเทอม C ปรากฏ 1 ครั้ง จะได้

$$tf_{A,j} = 3, tf_{B,j} = 2 \text{ และ } tf_{C,j} = 1$$

และจำนวนเอกสารใน Collection ที่มีเทอม A ปรากฏอยู่ 50 เอกสาร, เทอม B ปรากฏอยู่ 1,300 เอกสาร และเทอม C ปรากฏอยู่ 250 เอกสาร จะได้

$$df_A = 50, df_B = 1300 \text{และ } df_C = 250$$

ดังนั้นเทอม A,B และ C จะมีน้ำหนักความสำคัญต่อเอกสาร d_i เป็น

$$w_{A,j} : tf = 3/3;\ idf = \log(10000/50 = 5.3; \qquad tf\text{-}idf = 5.3$$

$$w_{B,j} : tf = 2/3;\ idf = \log(10000/1300) = 2.0; \quad tf\text{-}idf = 1.3$$

$$w_{C,j} : tf = 1/3;\ idf = \log(10000/250) = 3.7; \quad tf\text{-}idf = 1.2$$

จากตัวอย่างจะพบว่าเทอม B จะมีค่า *idf* เท่ากับ 2.0 ซึ่งน้อยกว่าเทอมอื่น ๆ เพราะเทอม B ปรากฏในหลาย ๆ เอกสารจึงทำให้อำนาจจำแนกเอกสาร d_j ออกจาก Collection นั้นมีค่าน้อยลงตามลำดับ

สรุปการทำงานของเวคเตอร์โมเดลคือ การแทนเอกสารใน Collection และ คิวรี่ ให้อยู่ในรูปเวคเตอร์ เมื่อทั้งเอกสารและคิวรี่อยู่ในรูปเวคเตอร์แล้ว ทำให้สามารถใช้โอเปอร์เรเตอร์ทางคณิตศาตร์มาใช้ประมวลผลในกระบวนการวัดความคล้ายได้ ประโยชน์ของเวคเตอร์คือ

1). ด้วยวิธีการ *tf-idf* ทำให้สามารถคำนวณ Degree of similarity ระหว่างคิวรี่ q กับเอกสาร d_i ด้วยโอเปอร์เรเตอร์ทางคณิตศาสตร์ได้

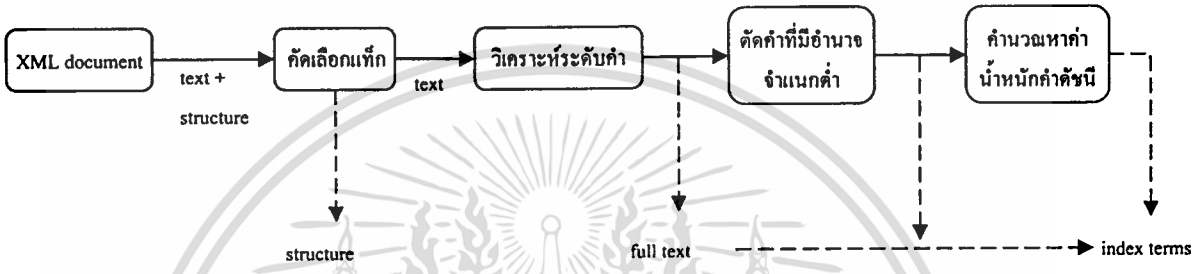
2). สามารถค้นคืนเอกสารแบบ Partial matching ได้

3). สามารถนำค่า Degree of similarity มาใช้ในกระบวนการ Ranking เพื่อเรียงลำดับเอกสารก่อนรายงานต่อผู้ใช้

อย่างไรก็ตาม มีโมเดลอื่น ๆ อีกหลายแบบได้ถูกนำเสนอขึ้นมาใช้งานใน IR system เช่น Probabilistic model เป็นต้น แต่ในวิทยานิพนธ์เล่มนี้ได้แสดงเพียงโมเดลพื้นฐานไว้เท่านั้น

บทที่ 4
การประมวลผลเอกสารและวิธีคำนวณน้ำหนักคำดัชนี

ในบทนี้กล่าวถึงการประมวลผลเอกสาร และวิธีคำนวณน้ำหนักคำดัชนี [7,8] เป็นขั้นตอนในการจัดทำดัชนีดังแสดงในรูปที่ 4.1 การประมวลผลเอกสารทำให้ได้คำหรือกลุ่มคำ ซึ่งคำหรือกลุ่มคำที่ได้นี้จะนำมาใช้ในการคำนวณน้ำหนักคำดัชนีเพื่อจัดลำดับความสำคัญของคำดัชนี



รูปที่ 4.1 แสดงขั้นตอนการทำดัชนี

จากรูปที่ 4.1 แสดงขั้นตอนการจัดทำดัชนีของเอกสาร XML โดยเริ่มจากการคัดเลือกแท็กที่มีความสำคัญในการจัดทำดัชนี ซึ่งในขั้นนี้เอกสารยังมีโครงสร้าง เมื่อผ่านการคัดเลือกแท็กแล้วจะได้ข้อมูลที่เป็นตัวอักษร จากนั้นนำไปวิเคราะห์ระดับคำเพื่อให้ได้คำหรือกลุ่มคำ และตัดคำที่มีอำนาจจำแนกต่ำ หลังจากนั้นนำคำหรือกลุ่มคำที่ได้ไปคำนวณค่าน้ำหนักคำดัชนี สุดท้ายจะได้กลุ่มคำดัชนี โดยจะได้อธิบายในรายละเอียดต่อไปในบทนี้

4.1 การประมวลผลเอกสาร XML

4.1.1 การจัดเตรียมเอกสาร XML

โครงสร้างเอกสาร XML นี้ออกแบบเพื่อจัดเก็บข้อมูลบทความ โดยออกแบบเพื่อให้เก็บข้อมูลทั้งหมดของบทความที่เป็นตัวอักษร ซึ่งแสดงโครงสร้างเอกสารและตัวอย่างเอกสารที่จัดเก็บไว้ในภาคผนวก ก หลังจากนั้นทำการเก็บรวบรวมข้อมูลจากบทความและวารสารต่าง ๆ ซึ่งข้อมูลส่วนใหญ่อยู่ในรูปไฟล์เอกสารของโปรแกรม MS Word โดยเปลี่ยนให้อยู่ในรูปแบบไฟล์ XML และให้ผู้เชี่ยวชาญคัดเลือกและจัดลำดับความสำคัญของคำดัชนีและจัดเก็บไว้ในแต่ละไฟล์ โดยไฟล์บทความเอกสาร XML ที่จัดทำขึ้นมีจำนวนทั้งหมด 300 ไฟล์ ซึ่งมีทั้งภาษาไทยและ

ภาษาอังกฤษ โดยมีเอกสาร XML ภาษาไทยร่วมกับภาษาอังกฤษจำนวน 181 ไฟล์ และเอกสาร XML ภาษาอังกฤษเพียงอย่างเดียวจำนวน 119 ไฟล์

4.1.2 การประมวลผลคำในเอกสาร XML

การประมวลผลคำในเอกสาร XML มีขั้นตอนดังต่อไปนี้

1. การคัดเลือกแท็ก คือเลือกแท็กที่มีความเกี่ยวข้องกับการทำดัชนี
2. การวิเคราะห์ระดับคำ (Lexical analysis) โดยทำการแยกข้อความออกมาว่าข้อมูลชนิดใด เช่น ตัวเลข, เครื่องหมาย หรือว่าตัวอักษร
3. ตัดคำที่ไม่ที่มีอำนาจจำแนกต่ำ (Elimination of stopwords)

ซึ่งจะได้อธิบายในรายละเอียดในหัวข้อถัดไป

4.1.2.1 การคัดเลือกแท็ก

การคัดเลือกแท็ก คือการคัดเลือกแท็กในเอกสารที่จะนำข้อมูลภายในแท็กนั้นมาคิดหาคำดัชนี โดยคัดเลือกแท็กที่มีความสำคัญของเอกสาร และตัดข้อมูลในแท็กที่ไม่สำคัญสำหรับเอกสารออก เพื่อลดจำนวนข้อมูลที่นำมาประมวลผล โดยงานวิจัยนี้ได้ตัดแท็ก ผู้แต่ง, ที่อยู่, กิตติกรรมประกาศ, บรรณานุกรม และประวัติผู้เขียนออก

4.1.2.2 การวิเคราะห์ระดับคำ

การวิเคราะห์ระดับคำคือการเปลี่ยนข้อมูลจากตัวอักษรมาสู่ข้อมูลที่อยู่ในรูปของคำ ดังนั้นส่วนสำคัญคือการระบุให้ได้ว่ากลุ่มตัวอักษรนั้นคือคำว่าอะไร นอกจากนี้ยังต้องสามารถจำแนกตัวเลข และเครื่องหมายวรรคตอนต่าง ๆ ออกจากตัวอักษรให้ได้

เครื่องหมายวรรคตอนสามารถที่จะเอาออกได้เนื่องจากจะมีผลต่อระบบค้นคืนน้อยมาก เมื่อผ่านขั้นตอนนี้แล้วจะได้คำและกลุ่มคำ ซึ่งในวิทยานิพนธ์นี้ใช้โปรแกรมตัดคำภาษาไทยช่วยในการวิเคราะห์ระดับคำภาษาไทย ส่วนภาษาอังกฤษใช้ช่องว่างแบ่งคำ และใช้การวิเคราะห์ระดับแท็กในการแบ่งคำหรือกลุ่มคำเฉพาะในแท็กที่ใช้เครื่องหมายวรรคตอนเฉพาะเช่น คำในแท็กคำสำคัญจะใช้เครื่องหมาย (,) ในการแบ่งคำหรือกลุ่มคำ

4.1.2.3 การตัดคำที่มีอำนาจจำแนกต่ำ

คำที่พบบ่อยในเอกสารเกิน 80% ของจำนวนเอกสารทั้งหมดในระบบถือว่าเป็นคำที่มีอำนาจจำแนกต่ำ (Stopword) ซึ่งจะถูกตัดทิ้งไม่นำมาคำนวณเป็นคำดัชนี เช่น คำนำหน้าชื่อ, คำบุพบท, คำสันธาน, บางคำของคำกริยา, คำกริยาวิเศษณ์และคำคุณศัพท์

หลังจากขั้นตอนนี้จะได้คำหรือกลุ่มคำที่นำไปคำนวณค่าน้ำหนักของคำดัชนี

4.2 วิธีคำนวณน้ำหนักคำดัชนีแบบไม่ใช้ค่าน้ำหนักแท็ก

วิธีคำนวณแบบนี้ไม่นำค่าน้ำหนักแท็กที่กำหนดให้สำหรับแต่ละแท็กร่วมคิดแต่คำนวณจากความถี่ของคำดัชนีที่ปรากฏในเอกสารเป็นหลักซึ่งเป็นการมองเอกสารแบบไม่มีโครงสร้างและไม่มีการให้ความสำคัญกับส่วนของข้อมูลที่มีความแตกต่างกันกัน ซึ่งเป็นการคำนวณแบบเวคเตอร์โมเดลทั่วไป [2] โดยมีวิธีคำนวณได้จากสมการด้านล่างนี้

การหาค่าความถี่มาตรฐาน $f_{i,j}$ ของเทอม k_i ในเอกสาร d_j หาได้จากสมการที่ 4.1

$$f_{i,j} = \frac{freq_{i,j}}{\max_l freq_{i,j}} \tag{4.1}$$

โดยที่ $freq_{i,j}$ คือ ค่าความถี่ของแต่ละเทอม k_i ที่พบในเอกสาร d_j, $\max_l freq_{i,j}$ คือ ค่าความถี่ของเทอม l ที่มีค่ามากที่สุดในแต่ละเอกสาร ในการหาค่าส่วนกลับความถี่เอกสารของเทอม k_i หรือค่า idf_i หาได้จากสมการที่ 4.2

$$idf_i = \log \frac{N}{n_i} \tag{4.2}$$

โดยที่ N คือ จำนวนเอกสารทั้งหมดในระบบ, n_i คือ จำนวนเอกสารที่มีเทอม k_i ปรากฏอยู่ ส่วนค่าน้ำหนักคำดัชนีตามวิธีเวกเตอร์โมเดลหาได้จากสมการที่ 4.3

$$w_{i,j} = f_{i,j} \times idf_i \tag{4.3}$$

ตัวอย่างการคำนวณ จากรูปที่ 4.2 หาค่าน้ำหนักของคำว่า "word" ในเอกสารที่ 5 ได้ดังนี้ โดยกำหนดให้ ค่าจำนวนเอกสารที่มีคำว่า "word" อยู่ (n_i) เท่ากับ 1 เพื่อไม่ให้ค่า $idf_{(word)}$ ที่คำนวณได้มีค่าเท่ากับ 0

$$w_{i,j}(word) = (9/9)*\log(5/1) = 0.69$$

ปัญหาของวิธีคำนวณแบบไม่ใช้ค่าน้ำหนักแท็กคือไม่มีการให้ความสำคัญกับส่วนของเอกสารดังที่ได้กล่าวไปในตอนต้นโดยจะคำนวณโดยใช้เพียงความถี่ของเอกสารเพียงอย่างเดียวซึ่งเหมาะกับเอกสารที่ไม่มีโครงสร้างที่ชัดเจนและไม่รู้ความสำคัญของแต่ละส่วนเอกสารซึ่งไม่

เหมือนกับเอกสาร XML ที่ทราบโครงสร้างของเอกสารและความสำคัญของแต่ละส่วนเอกสารผ่านแท็กที่ข้อมูลนั้นปรากฏอยู่

4.3 วิธีคำนวณน้ำหนักคำดัชนีแบบใช้ค่าน้ำหนักแท็กอย่างเดียวเป็นค่าน้ำหนักแท็ก

วิธีคำนวณแบบใช้ค่าน้ำหนักแท็กอย่างเดียวเป็นค่าน้ำหนักของคำดัชนี คือ การนำค่าน้ำหนักแท็กที่ได้จากการกำหนดซึ่งจะได้กล่าวถึงการกำหนดค่าน้ำหนักแท็กในบทถัดไปมาเป็นค่าน้ำหนักแท็กแล้วนำมาคำนวณเป็นค่าน้ำหนักคำดัชนี เพื่อทำให้ค่าน้ำหนักคำดัชนีที่อยู่ในแท็กที่มีความสำคัญมากกว่ามีค่าน้ำหนักคำดัชนีมากกว่าค่าน้ำหนักคำดัชนีที่อยู่ในแท็กที่มีความสำคัญน้อยกว่า ซึ่งสามารถแบ่งการนำค่าน้ำหนักแท็กมาใช้ในการคำนวณได้ 2 วิธี คือ 1. วิธีการคำนวณแบบรวมค่าน้ำหนักแท็กที่พบ และ 2. วิธีการคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบ

4.3.1 วิธีคำนวณแบบรวมค่าน้ำหนักแท็กที่พบ (A)

แนวคิดในการคำนวณแบบรวมค่าน้ำหนักแท็กที่พบ [9] คือ ถ้าพบคำที่แท็กใดให้นำค่าน้ำหนักของแท็กมาคิดเพิ่มให้กับค่า $w_{i,j}$ ที่ได้จากสมการที่ (4.3) จะได้ดังสมการที่ (4.4)

$$w'_{i,j} = \sum w_k \times freq_{i,j} \times idf_i \qquad (4.4)$$

โดยที่ $freq_{i,j}$ คือ ค่าความถี่ของแต่ละเทอม k_i ที่พบในเอกสาร d_j , w_k คือ ค่าน้ำหนักของแต่ละแท็ก t_k ในเอกสาร ส่วนการหาค่า idf_i ยังคงใช้วิธีการคำนวณเดิมจาก (4.2)

ตัวอย่างการคำนวณ จากรูปที่ 4.2 หาค่าน้ำหนักของคำว่า "word" ในเอกสารที่ 5 ได้ดังนี้ โดยกำหนดให้ ค่าจำนวนเอกสารที่มีคำว่า "word" อยู่ (n_i) เท่ากับ 1 เพื่อไม่ให้ค่า $idf_{(word)}$ ที่คำนวณได้มีค่าเท่ากับ 0

$$w'_{i,j}(word) = (0.9+0.8+0.4)*(9)*\log(5/1) = 13.21$$

ข้อสังเกตวิธีนี้จะรวมค่าน้ำหนักแท็กของคำที่พบโดยไม่ให้ความสำคัญกับความถี่ของคำที่พบในแท็กนั้นแต่ให้ความสำคัญแบบรวม

4.3.2 วิธีคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบ (B)

จากข้อเสนอแนะวิธีข้างต้นจะได้ว่าคำที่ปรากฏบ่อยครั้งในแท็กเดียวกันควรจะมีค่าน้ำหนักมากกว่าคำที่ปรากฏน้อยครั้งกว่า จึงมีการปรับปรุงสมการที่ (4.4) ทำให้ความถี่ของคำในแต่ละแท็กมีความสำคัญแตกต่างกันไป [10] ดังการคำนวณด้านล่างนี้

การหาค่าความถี่มาตรฐาน $f_{i,j,k}$ ของเทอม k_i ในเอกสาร d_j ในแท็ก t_k หาได้จากสมการที่ (4.5)

$$f_{i,j,k} = \sum_{k=1}^{t} \left(w_k \times freq_{i,j} \right) \tag{4.5}$$

โดยที่ $freq_{i,j}$ คือ ค่าความถี่ของแต่ละเทอม k_i ที่พบในเอกสาร d_j , w_k คือ ค่าน้ำหนักของแต่ละแท็ก t_k ในเอกสาร ส่วนการหาค่า idf_i ยังคงใช้วิธีการคำนวณเดิมจาก (4.2)
ดังนั้นค่าน้ำหนักคำดัชนีตามการคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบคำนวณได้ตามสมการ

$$w''_{i,j} = f_{i,j,k} \times idf_i \tag{4.6}$$

ตัวอย่างการคำนวณ จากรูปที่ 4.2 หาค่าน้ำหนักของคำว่า "word" ในเอกสารที่ 5 ได้ดังนี้ โดยกำหนดให้ ค่าจำนวนเอกสารที่มีคำว่า "word" อยู่ (n_i) เท่ากับ 1 เพื่อไม่ให้ค่า $idf_{(word)}$ ที่คำนวณได้มีค่าเท่ากับ 0

$$w''_{i,j}(word) = [(0.9*1)+(0.8*3)+(0.4*5)]*\log(5/1) = 3.70$$

จากตัวอย่างเอกสาร XML รูปที่ 4.2 สมมุติให้มีคำว่า "word" ปรากฏซ้ำ ๆ ในเอกสาร จากนั้นได้แสดงให้เห็นถึงความแตกต่างในการคำนวณค่าน้ำหนักคำดัชนีจากการคำนวณทั้ง 3 วิธีข้างต้นได้ผลการคำนวณดังตารางที่ 4.1 โดยการคำนวณหาค่าน้ำหนักด้วยวิธีเวกเตอร์โมเดลคำนวณค่า $w_{i,j}$ จากสมการที่ (4.3) วิธีการคำนวณแบบรวมค่าน้ำหนักแท็กที่พบคำนวณค่า $w'_{i,j}$ จากสมการที่ (4.4) และวิธีการคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบคำนวณค่า $w''_{i,j}$ จากสมการที่ (4.6)

ตารางที่ 4.1 แสดงผลการคำนวณค่าน้ำหนักด้วยวิธีที่ต่างกัน

เอกสาร	w_1	$f_{1,j}$	w_2	$f_{2,j}$	w_3	$f_{3,j}$	$w_{i,j}$	$w'_{i,j}$	$w''_{i,j}$
d1	0.9	1	0.8	-	0.4	8	0.69	8.18	2.87
d2	0.9	8	0.8	-	0.4	1	0.69	8.18	5.31
d3	0.9	3	0.8	3	0.4	3	0.69	13.21	4.40
d4	0.9	1	0.8	4	0.4	4	0.69	13.21	3.98
d5	0.9	1	0.8	3	0.4	5	0.69	13.21	3.70

จากตารางที่ 4.1 ในเอกสารที่ 1 และ 2 แสดงให้เห็นว่าคำที่ปรากฏในแท็กเดียวกันแต่มีความถี่ต่างกัน ค่าน้ำหนักที่คำนวณด้วยวิธีรวมค่าน้ำหนักแท็กที่พบ ($w'_{i,j}$) จะให้ค่าไม่ต่างกัน แต่ถ้าคำนวณด้วยวิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบจะทำให้ได้ค่าน้ำหนัก ($w''_{i,j}$) ที่แตกต่างกัน ดังแสดงในเอกสารที่ 3, 4 และ 5 ส่วนการคำนวณหาค่าน้ำหนักด้วยวิธีเวคเตอร์โมเดล ($w_{i,j}$) จะให้ค่าไม่แตกต่างกันในทุกเอกสาร เนื่องจากใช้ความถี่ของคำที่พบและค่าส่วนกลับความถี่เอกสารเพียงอย่างเดียว (สังเกตว่าทั้ง 5 เอกสารมีคำว่า "word" ปรากฏ 9 ครั้งเช่นเดียวกัน)

```
<d1>
<tag1>...word...</tag1>
<tag3>...word...word...word...word...word…word…word…word…</tag3>
</d1>
<d2>
<tag1>...word...word...word...word...word…word…word…word…</tag1>
<tag3>...word...</tag3>
</d2>
<d3>
<tag1>…word…word…word…</tag1>
<tag2>…word…word…word…</tag2>
<tag3>…word…word…word…</tag3>
</d3>
<d4>
<tag1>…word…</tag1>
<tag2>…word…word…word…word…</tag2>
<tag3>…word…word…word…word…</tag3>
</d4>
<d5>
<tag1>…word…</tag1>
<tag2>…word…word…word…</tag2>
<tag3>…word…word…word…word…word…</tag3>
</d5>
```

รูปที่ 4.2 แสดงตัวอย่างเอกสาร XML

4.4 วิธีคำนวณน้ำหนักคำดัชนีแบบใช้ค่าน้ำหนักแท็กและค่าจำนวนคำในแท็กเป็นค่าน้ำหนักแท็ก

วิธีนี้ได้ปรับปรุงการให้ค่าน้ำหนักแท็กด้วยการนำค่าน้ำหนักแท็กจากการกำหนดให้ในแต่ละแท็กรวมกับค่าจำนวนคำในแท็กที่คำ ๆ นั้นปรากฏอยู่แล้วใช้เป็นค่าน้ำหนักแท็กแล้วนำมาคำนวณค่าน้ำหนักคำดัชนี เพื่อทำให้ค่าน้ำหนักแท็กที่ได้มีทั้งส่วนที่มาจากการกำหนดโดยตรง คือ ค่าน้ำหนักของแต่ละแท็กที่กำหนดขึ้น และจากข้อมูลภายในของแต่ละเอกสาร คือ ค่าจำนวนคำในแท็ก ซึ่งการรวมกันของทั้งสองส่วนทำให้เกิดความยืดหยุ่นเพิ่มขึ้นในการนำมาใช้เป็นค่าน้ำหนักแท็กแทนที่การใช้แต่ค่าน้ำหนักแท็กที่ได้จากการกำหนดขึ้นเพียงอย่างเดียว ซึ่งวิธีการคำนวณแบบนี้แบ่งได้ 2 วิธี โดยมีวิธีการคำนวณเหมือนกับวิธีในหัวข้อ 4.3 แต่แตกต่างกันที่ค่าน้ำหนักแท็กที่นำมาใช้ดังแสดงในรายละเอียดของการคำนวณแต่ละวิธีต่อไป

4.4.1 วิธีคำนวณแบบรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก (AL)

การคำนวณวิธีนี้ได้ปรับปรุงการให้ค่าน้ำหนักแท็กในสมการที่ (4.4) โดยนำค่าจำนวนคำในแท็กมาร่วมคำนวณกับค่าน้ำหนักแท็กที่กำหนดขึ้นด้วยดังสมการที่ (4.10)

$$w'^{tm}_{i,j} = \sum \left(\frac{w_k}{word\_tag\_length_k} \right) \times freq_{i,j} \times idf_i \qquad (4.10)$$

โดยที่ w_k คือ ค่าน้ำหนักของแต่ละแท็ก t_k ในเอกสาร, $word\_length_k$ คือ ค่าจำนวนคำในแต่ละแท็ก t_k ในเอกสาร, $freq_{i,j}$ คือ ค่าความถี่ของแต่ละเทอม k_i ที่พบในเอกสาร d_j, idf_i ยังคงใช้วิธีการคำนวณเดิมจากสมการที่ (4.2)

ตัวอย่างการคำนวณ จากรูปที่ 4.2 หาค่าน้ำหนักของคำว่า "word" ในเอกสารที่ 5 ได้ดังนี้ โดยกำหนดให้ ค่าจำนวนเอกสารที่มีคำว่า "word" อยู่ (n_i) เท่ากับ 1 เพื่อไม่ให้ค่า $idf_{(word)}$ ที่คำนวณได้มีค่าเท่ากับ 0 และค่าจำนวนคำในแท็กนับจากจำนวนของคำว่า "word" เพียงคำเดียว

$$w'^{tm}_{i,j}(word) = \left[\left(\frac{0.9}{1} \right) + \left(\frac{0.8}{3} \right) + \left(\frac{0.4}{5} \right) \right] * 9 * \log(5/1) = 7.84$$

4.4.2 วิธีคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก (BL)

การคำนวณวิธีนี้ได้ปรับปรุงการให้ค่าน้ำหนักแท็กในสมการที่ (4.6) โดยนำค่าจำนวนคำในแท็กมาร่วมคำนวณกับค่าน้ำหนักแท็กที่กำหนดขึ้นด้วยดังสมการที่ (4.11) การหาค่าความถี่มาตรฐาน $f_{i,j,k}$ ของเทอม k_i ในเอกสาร d_j ในแท็ก t_k หาได้จากสมการที่ (4.11)

$$f_{i,j,k} = \sum_{k=1}^{i} \left(weight\_length_k \times freq_{i,j} \right) \quad (4.11)$$

โดยที่ $freq_{i,j}$ คือ ค่าความถี่ของแต่ละเทอม k_i ที่พบในเอกสาร d_j , $weigth\_length_k$ คือ ค่าน้ำหนักคูณด้วยส่วนกลับของจำนวนคำในแต่ละเท็ก t_k ในเอกสาร โดยหาได้จากสมการที่ (4.12) ส่วนการหาค่า idf_i ยังคงใช้วิธีการคำนวณเดิมจากสมการที่ (4.2)

$$weight\_length_k = \frac{w_k}{word\_tag\_length_k} \quad (4.12)$$

โดยที่ w_k คือ ค่าน้ำหนักของแท็ก t_k ในเอกสาร, $word\_tag\_length_k$ คือ จำนวนคำในแท็ก t_k ดังนั้นค่าน้ำหนักคำดัชนีตามการคำนวณค่าน้ำหนักโดยใช้ค่าจำนวนคำในแท็กเพียงอย่างเดียว คำนวณได้ตามสมการที่ (4.13)

$$w''''_{i,j} = f_{i,j,k} \times idf_i \quad (4.13)$$

ตัวอย่างการคำนวณ จากรูปที่ 4.2 หาค่าน้ำหนักของคำว่า "word" ในเอกสารที่ 5 ได้ดังนี้ โดยกำหนดให้ ค่าจำนวนเอกสารที่มีคำว่า "word" อยู่ (n_i) เท่ากับ 1 เพื่อไม่ให้ค่า $idf_{(word)}$ ที่คำนวณได้มีค่าเท่ากับ 0 และค่าจำนวนคำในแท็กนับจากจำนวนของคำว่า "word" เพียงคำเดียว

$$w''''_{i,j}(word) = \left[\left(\frac{0.9}{1} * 1 \right) + \left(\frac{0.8}{3} * 3 \right) + \left(\frac{0.4}{5} * 5 \right) \right] * \log(5/1) = 1.47$$

จากวิธีการคำนวณทั้งหมดที่นำเสนอมานี้จะนำไปทดลองคำนวณกับเอกสาร XML บทความทางวิชาการเพื่อค้นหาว่าวิธีการคำนวณแบบใดที่เหมาะสมกับเอกสาร XML บทความวิจัย

บทที่ 5
การกำหนดค่าน้ำหนักแท็ก

บทนี้กล่าวถึงการกำหนดค่าน้ำหนักแท็กวิธีต่าง ๆ เพื่อนำค่าน้ำหนักแท็กที่ได้ไปใช้คำนวณหาค่าน้ำหนักคำดัชนีด้วยวิธีคำนวณน้ำหนักคำดัชนีที่นำเสนอในบทที่แล้ว ซึ่งการกำหนดค่าน้ำหนักแท็กที่นำเสนอในงานวิจัยนี้มี 3 วิธีคือ 1. การกำหนดค่าน้ำหนักแท็กโดยผู้ใช้ 2. การกำหนดค่าน้ำหนักแท็กโดยใช้จำนวนคำในแท็ก และ 3. การกำหนดค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึม

5.1 การกำหนดค่าน้ำหนักแท็กโดยผู้ใช้

การกำหนดค่าน้ำหนักแท็กโดยผู้ใช้กำหนดขึ้นโดยดูจากโครงสร้างเอกสารที่ใช้เก็บบทความวิจัยซึ่งแสดงไว้ในภาคผนวก ก โดยการกำหนดค่าระหว่าง 1-10 เพื่อแสดงถึงความสำคัญที่แตกต่างกันของแต่ละแท็ก โดยแท็กที่กำหนดขึ้นนำมาจากการหาค่า XPath จากโครงสร้างบทความวิจัย ซึ่งมีรายละเอียดของแต่ละเส้นทางดังตารางที่ 5.1

ตารางที่ 5.1 แสดงความสัมพันธ์ระหว่างชื่อแท็กและเส้นทางข้อมูลของโครงสร้างบทความวิจัย

ชื่อแท็ก	XPath	หมายเหตุ
Keyword	/paper/keyword	เก็บคำสำคัญ
Title	/paper/title	เก็บชื่อบทความ
Abstract	/paper/abstract	เก็บบทคัดย่อ
Introduction	/paper/intro	เก็บส่วนบทนำ
Chapter – Name	/paper/chapter/name	เก็บชื่อหัวข้อ
Chapter - Content	/paper/chapter/content	เก็บเนื้อหาในหัวข้อ
Subchapter - Name	/paper/chapter/subchapter/name	เก็บชื่อหัวข้อย่อย
Subchapter - Content	/paper/chapter/subchapter/content	เก็บเนื้อหาในหัวข้อย่อย
Experiment - Content	/paper/experiment/content	เก็บเนื้อหาวิธีการทดลอง
SubExperiment - Name	/paper/experiment/subchapter/name	เก็บชื่อหัวข้อย่อยในส่วนวิธีการทดลอง
SubExperiment - Content	/paper/experiment/subchapter/content	เก็บเนื้อหาในหัวข้อย่อยในส่วนวิธีกาทดลอง
Result – Content	/paper/result/content	เก็บเนื้อหาผลการทดลอง
SubResult - Name	/paper/result/subchapter/name	เก็บชื่อหัวข้อย่อยในส่วนผลการทดลอง

ตารางที่ 5.1 (ต่อ)

ชื่อแท็ก	XPath	หมายเหตุ
SubResult - Content	/paper/result/subchapter/content	เก็บเนื้อหาในส่วนของหัวข้อย่อยผลการทดลอง
Summary - Content	/paper/summary/content	เก็บเนื้อหาส่วนสรุป
SubSummary - Name	/paper/summary/subchapter/name	เก็บชื่อหัวข้อย่อยในส่วนสรุป
SubSummary - Content	/paper/summary/subchapter/content	เก็บเนื้อหาในส่วนของหัวข้อย่อยสรุป

จากตารางที่ 5.1 ได้จำนวนแท็กที่ต้องให้น้ำหนักทั้งหมด 17 แท็ก ซึ่งเป็นการแบ่งส่วนของบทความวิจัยเป็นส่วน ๆ ซึ่งแต่ละส่วนเก็บข้อมูลที่อยู่ในกลุ่มเดียวกันไว้ ตัวอย่างการหาค่าข้อมูลในแท็ก SubResult - Content ซึ่งได้จาก XPath -> /paper/result/subchapter/content จากตัวอย่างบทความในภาคผนวก ก ได้ค่าข้อมูลในแท็กนี้คือ *"ลักษณะการกระจายอุณหภูมิ..."* เป็นต้น ในงานวิจัยนี้ได้เก็บตัวอย่างการให้ค่าน้ำหนักจากผู้ใช้จำนวน 10 คนเพื่อใช้ในการทดลองดังแสดงในตารางที่ 5.2

ตารางที่ 5.2 แสดงค่าน้ำหนักที่กำหนดโดยผู้ใช้

ชื่อแท็ก	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10
Keyword	9	9	9	8	9	9	7	6	9	9
Title	9	7	9	9	8	9	8	9	10	8
Abstract	8	8	8	9	5	8	9	7	8	7
Introduction	7	6	8	7	5	5	9	4	6	3
Chapter – Name	9	6	7	6	7	5	8	7	7	5
Chapter - Content	8	6	6	6	4	4	7	6	8	2
Subchapter - Name	8	5	6	6	6	3	7	5	8	5
Subchapter - Content	7	5	6	6	4	3	6	4	6	2
Experiment - Content	7	6	8	7	5	5	9	5	6	2
SubExperiment - Name	7	6	8	6	5	3	6	5	7	5
SubExperiment -Content	7	7	8	6	4	3	6	4	6	2
Result – Content	7	7	7	7	4	8	9	6	6	4
SubResult - Name	7	6	8	6	5	5	6	5	7	5
SubResult - Content	7	6	8	6	4	5	6	4	5	2
Summary - Content	9	8	9	8	5	8	9	7	7	7
SubSummary - Name	8	7	7	7	6	5	5	5	7	3
SubSummary - Content	8	8	7	7	5	5	5	5	5	3

จากตารางที่ 5.2 จะเห็นว่าแท็กที่ค่าน้ำหนักมากประกอบด้วยแท็ก Keyword, Title, Abstract และ Summary – Content ซึ่งเป็นส่วนสำคัญของเอกสารบทความวิชาการ ส่วนแท็กอื่น ๆ จะมีค่าน้ำหนักน้อยเนื่องจากมีความสำคัญน้อย

5.2 การกำหนดค่าน้ำหนักแท็กโดยใช้จำนวนคำในแท็ก

การกำหนดค่าน้ำหนักแท็กด้วยวิธีนี้กำหนดจากจำนวนคำในแท็กซึ่งได้จากส่วนกลับจำนวนคำในแท็กที่คำดัชนีนั้น ๆ ปรากฏอยู่นำมากำหนดเป็นค่าน้ำหนักแท็กของแท็กนั้น ซึ่งเป็นค่าน้ำหนักแท็กที่คำนวณด้วยวิธีนี้สามารถเปลี่ยนแปลงได้ขึ้นอยู่กับจำนวนของข้อมูลในแท็กนั้น ซึ่งคำนวณค่าน้ำหนักแท็กได้จากสมการ

$$w_k = \frac{1}{word\_tag\_length_k} \tag{5.1}$$

โดยที่ $word\_tag\_length_k$ คือ จำนวนคำในแท็ก t_k แต่การกำหนดค่าน้ำหนักแท็กด้วยวิธีนี้นำไปใช้คำนวณได้เฉพาะวิธีในหัวข้อ 4.3 คือวิธีคำนวณแบบใช้ค่าน้ำหนักแท็กอย่างเดียวเป็นค่าน้ำหนักแท็กของคำดัชนี แต่ไม่สามารถนำไปใช้กับวิธีในหัวข้อ 4.4 คือวิธีคำนวณแบบใช้ค่าน้ำหนักแท็กและค่าจำนวนคำในแท็กเป็นค่าน้ำหนักแท็กของคำดัชนี เนื่องจากค่า w_k ที่คำนวณได้จะทำให้ค่าน้ำหนักแท็กนั้นมีค่าเป็นผลรวมของจำนวนแท็กที่พบสำหรับวิธีคำนวณแบบรวมค่าน้ำหนักแท็กที่พบ และจะไม่มีค่าสำหรับวิธีคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบ

5.3 การกำหนดค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึม

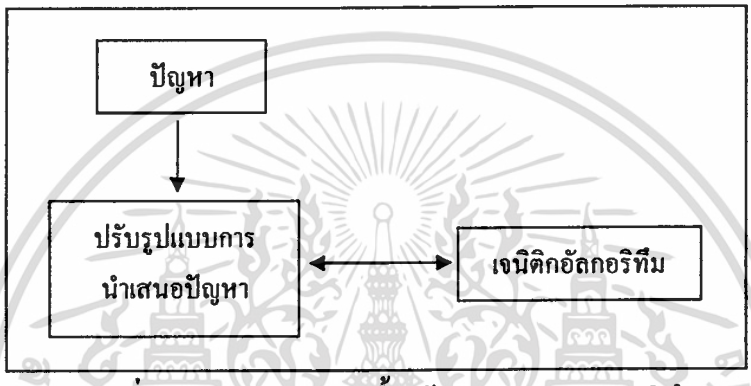
การกำหนดค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึม [11,12] เป็นการหาค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึมช่วยในการหาค่าน้ำหนักแท็กสำหรับวิธีคำนวณค่าน้ำหนักคำดัชนีแบบต่าง ๆ ในบทที่ 4 ในหัวข้อนี้นำเสนอหลักการของเจนิติกอัลกอริทึม, การประยุกต์หลักการของเจนิติกอัลกอริทึมสำหรับปรับค่าน้ำหนักแท็ก และผลการหาค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึม ซึ่งมีรายละเอียดดังต่อไปนี้

5.3.1 หลักการของเจนิติกอัลกอริทึม

ในปี ค.ศ. 1975 John Holland ได้ศึกษาเกี่ยวกับวิวัฒนาการทางธรรมชาติ (Natural Evolution) ในการให้กำเนิดประชากรสิ่งมีชีวิตในรุ่นต่อ ๆ ไปโดยกระบวนการทางชีววิทยาประกอบด้วยการคัดเลือกทางธรรมชาติ (Natural Selection) คือ สิ่งมีชีวิตที่แข็งแรงกว่าย่อมมีโอกาสอยู่รอดมากกว่าสิ่งมีชีวิตที่อ่อนแอ เปรียบเหมือนโครโมโซมที่ประกอบด้วยยีนส์ต่าง ๆ ที่มีลักษณะที่ดีสามารถอยู่รอดได้มากกว่า โครโมโซมที่อยู่รอดได้ก็จะถ่ายทอดยีนส์ที่มีลักษณะที่ดีเหล่านั้นไปยัง

ลูกหลานได้มากกว่าเช่นกัน ด้วยกระบวนการทางพันธุกรรมศาสตร์(Genetic Operation) คือการกำเนิดโครโมโซมใหม่จากการครอสโอเวอร์ (Crossover) หรือมิวเตชัน (Mutation)

จากแนวคิดดังกล่าว Holland จึงได้นำปรับมาใช้กับการแก้ปัญหาด้วยคอมพิวเตอร์เพื่อหาคำตอบที่ดีที่สุดหรือใกล้เคียงที่สุดโดยมีจุดมุ่งหมายที่จะศึกษาระบบปรับปรุงการประมวลผลเอง (Self Adaptive Process) และสร้างระบบผู้เชี่ยวชาญ (Artificial System) โดยอาศัยแนวคิดของระบบการคัดเลือกทางธรรมชาติเรียกว่า เจนิติกอัลกอริทึม (Genetic Algorithms : GA) เพื่อปรับปรุงการหาคำตอบที่ดีขึ้น หลักการเบื้องต้นในการใช้เจนิติกอัลกอริทึมแก้ปัญหา คือต้องปรับปรุงรูปแบบปัญหาให้เหมาะสมกับการนำเสนอของเจนิติกอัลกอริทึม ดังรูปที่ 5.1



รูปที่ 5.1 แสดงหลักการเบื้องต้นของเจนิติกอัลกอริทึม

เจนิติกอัลกอริทึมเป็นวิธีการค้นหาคำตอบโดยการเลียนแบบการคัดเลือกทางธรรมชาติและธรรมชาติทางพันธุกรรมซึ่งอาศัยหลักการสุ่มเพื่อปรับปรุงความสามารถในการค้นหาคำตอบที่ดีขึ้น โดยมีวิธีการคือ

1. ค้นหาคำตอบภายใต้โครงสร้างของปัญหา อันเกิดจากการกำหนดรหัส (Coding) รูปแบบโครงสร้างจากกลุ่มตัวแปรต่าง ๆ ของปัญหานั้น ไม่ใช่ค้นหาคำตอบจากค่าของกลุ่มตัวแปรนั้น
2. ค้นหาคำตอบโดยพิจารณาจากประชากรคำตอบ หรือ กลุ่มคำตอบ ไม่ใช่จากค่าของกลุ่มตัวแปร
3. ค้นหาคำตอบจากผลลัพธ์ของกลุ่มค่าตัวแปรที่เป็นฟังก์ชันเป้าหมายของปัญหา
4. ค้นหาคำตอบโดยอาศัยการถ่วงน้ำหนักความเหมาะสมของแต่ละคำตอบจากกลุ่มคำตอบนั้นๆ

5.3.1.1 ฟังกชันเป้าหมายกับฟังก์ชันความเหมาะสม

การหาคำตอบที่ดีที่สุดของเจนิติกอัลกอริทึมเป็นการนำผลลัพธ์ที่ได้จากการหาคำตอบครั้งก่อนมาปรับปรุงให้ดีขึ้น วิธีการของเจนิติกอัลกอริทึมจะพิจารณาว่าคำตอบใหม่ที่ได้รับนั้นดีขึ้นหรือไม่ หรือเป็นคำตอบที่ใกล้เคียงกับคำตอบที่ต้องการหรือไม่จากฟังก์ชันเป้าหมาย (Objective Function: f) ในแต่ละปัญหาจะสามารถกำหนดฟังก์ชันเป้าหมายได้ตามรูปแบบของปัญหา โดยฟังก์ชันเป้าหมายเป็นฟังก์ชันที่แสดงความสัมพันธ์ของตัวแปร พารามิเตอร์ เงื่อนไข หรือข้อกำหนดต่าง ๆ ของปัญหา สำหรับฟังก์ชันความเหมาะสม (Fitness Function: F) เป็นฟังก์ชันที่ใช้เป็นตัวกำหนดค่าความเหมาะของแต่ละโครโมโซมว่ามีโอกาสถูกคัดเลือกมากน้อยเพียงใด โดยส่วนใหญ่จะใช้ฟังก์ชันเป้าหมายเป็นฟังก์ชันความเหมาะสม หรืออาจใช้ฟังก์ชันเป้าหมายที่ถูกปรับให้เหมาะสมเป็นฟังก์ชันความเหมาะสมได้

5.3.1.2 รูปแบบโครโมโซม

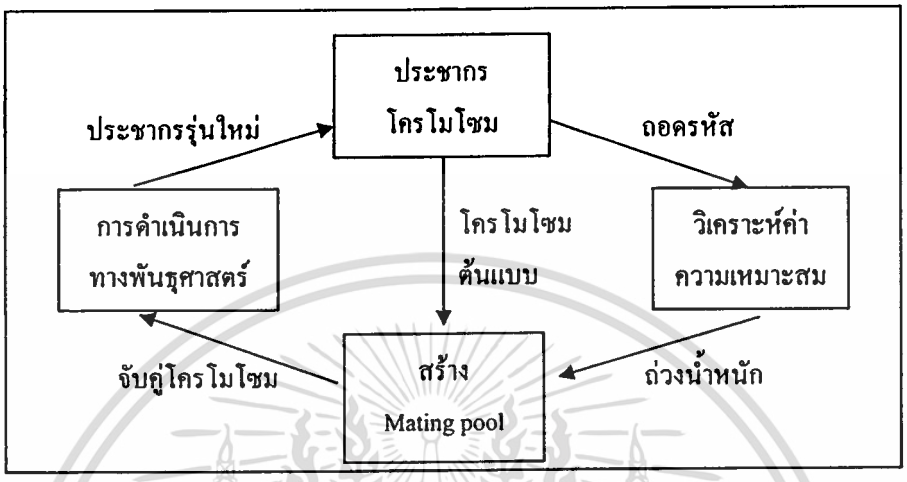
การจำลองแบบทางธรรมชาติของเจนิติกอัลกอริทึมเพื่อใช้แก้ปัญหาเริ่มจากการมองปัญหาเทียบเท่ากับโครโมโซมชนิดหนึ่ง โดยประกอบด้วยยีนลักษณะต่าง ๆ ซึ่งหมายถึงข้อมูลต่าง ๆ เมื่อแปลความหมายแล้วจะให้ค่าของคำตอบค่าหนึ่ง ในเจนิติกอัลกอริทึมยีนที่อยู่ในโครโมโซมเป็นตัวแสดงค่าคำตอบ ๆ หนึ่งของปัญหา ที่แปรผันไปตามการประยุกต์ใช้งานซึ่งโดยทั่วไปยีนหมายถึงตัวแปร พารามิเตอร์ เงื่อนไข หรือ ข้อกำหนดต่าง ๆ ที่เป็นองค์ประกอบของปัญหา การกำหนดรูปแบบของแต่ละโครโมโซมทำได้โดยการแปลงตัวแปร พารามิเตอร์ เงื่อนไข หรือข้อกำหนดต่าง ๆ ให้อยู่ในรูปลำดับของยีนบนโครโมโซมหรือเรียกว่าสตริง (String) อันประกอบด้วยบิท (Bit) หรือเรียกว่าอักขระ (Character) ซึ่งลักษณะต่าง ๆ ที่เป็นไปได้ของแต่ละยีนคือค่าของบิท (Bit Value) หรือค่าตัวแปร พารามิเตอร์ ต่าง ๆ ที่เป็นไปได้ การกำหนดรูปแบบของปัญหาให้เป็นไปตามธรรมชาติโดยกำหนดรหัสในรูปแบบตัวเลขหรือตัวอักษรในช่วงที่จำกัดตามค่าตัวแปรหรือพารามิเตอร์ และประกอบรวมกันเป็นยีนหรือโครโมโซมที่มีความยาวคงที่

5.3.1.3 วัฏจักรการทำงานของเจนิติกอัลกอริทึม

เมื่อกำหนดรูปแบบโครโมโซมและฟังก์ชันความเหมาะสมของปัญหาแล้ว เจนิติกอัลกอริทึมจะสามารถประมวลผลหาคำตอบของปัญหาได้ โดยสร้างวิวัฒนาการกลุ่มคำตอบในรุ่นต่อไปตามวัฏจักรการทำงานของเจนิติกอัลกอริทึม (Genetic Algorithm Cycle) ดังรูปที่ 5.2 ซึ่งมี 4 ขั้นตอน คือ

1. สร้างประชากรรุ่นเก่าตามรูปแบบที่กำหนดไว้ โดยประชากรต้นกำเนิด (Initial Popular) เกิดจากการสร้างชุดโครโมโซมโดยการสุ่มค่าแต่ละบิท
2. วิเคราะห์ค่าความเหมาะสมของแต่ละโครโมโซมโดยถอดรหัสค่าตัวแปร พารามิเตอร์ต่างๆ ของแต่ละบิทและคำนวณค่าความเหมาะสม

3. สร้าง mating pool คือชุดโครโมโซมต้นแบบหรือโครโมโซม พ่อ-แม่ ที่สามารถอยู่รอดเป็นต้นแบบ โดยพิจารณาถ่วงน้ำหนักจากค่าความเหมาะสมของแต่ละโครโมโซม
4. ดำเนินการทางพันธุศาสตร์ โดยสุ่มจับคู่โครโมโซมต้นแบบใน mating pool เพื่อสร้างประชากรรุ่นใหม่ ซึ่งตัวดำเนินการทางพันธุศาสตร์ประกอบด้วยครอสโอเวอร์หรือมิวเตชัน



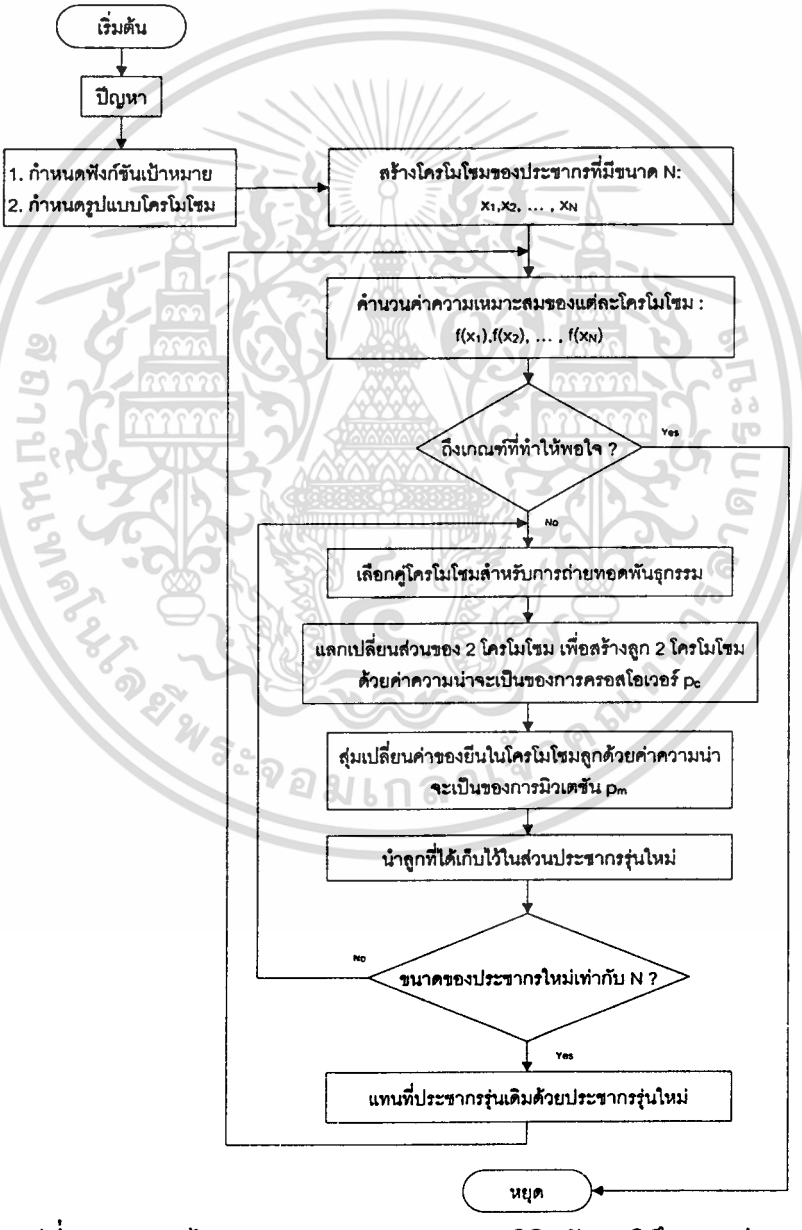
รูปที่ 5.2 วัฏจักรการทำงานของเจนิติกอัลกอริทึม
ที่มา : [10]

การค้นหาคำตอบของเจนิติกอัลกอริทึม จะประมวลผลซ้ำตามวัฏจักรจนกว่าจะได้รับคำตอบที่พอใจตามเกณฑ์ที่ตั้งไว้ หรือในระยะเวลาตามจำนวนรุ่นที่ดำเนินการตามต้องการซึ่งแสดงอัลกอริทึมการทำงานของเจนิติกอัลกอริทึมดังนี้

```
{
    gen = 0;
    Initpopulation P(gen);      // สร้างประชากรโครโมโซมต้นกำเนิดโดยการสุ่ม
    Evaluate P(gen);            // วิเคราะห์ค่าความเหมาะสมแต่ละโครโมโซมประชากรต้นกำเนิด
    While (termination criterion not reached) {  // ตรวจสอบเงื่อนไขความพอใจ
        gen = gen+1;
        if ( P'(gen).size < popsize) {
            P'(gen) = Selectparents P(gen-1); // คัดเลือกโครโมโซมต้นแบบจากประชากรรุ่นก่อน
            Recombine P'(gen);          // แลกเปลี่ยนส่วนยีนส์ภายในโครโมโซมต้นแบบ
            Mutate P'(gen);             // มิวเตชันโครโมโซมต้นแบบ
        }
    P(gen) = P'(gen);           // ประชากรรุ่นใหม่กลายเป็นประชากรรุ่นเก่าต่อไป
```

```
    Evaluate P(gen);              // วิเคราะห์ค่าความเหมาะสมของประชากรรุ่นใหม่
    }
}
```

ยุคแรก ๆ ของการเริ่มใช้งานเจเนติอัลกอริทึมจะเป็นเจนิติกอัลกอริทึมแบบง่าย (Simple Genetic Algorithm : SGA) ซึ่งมีพื้นฐานและมีกระบวนการไม่มากนัก ง่ายในการศึกษาทำความเข้าใจในแต่ละขั้นตอน การทำงานของเจนิติกอัลกอริทึมแบ่งเป็น 2 ส่วน คือส่วนการเตรียมการและส่วนการทำงาน ซึ่งแสดงไดอะแกรมการทำงานของเจนิติกอัลกอริทึมแบบง่ายดังรูปที่ 5.3



รูปที่ 5.3 แสดงไดอะแกรมการทำงานของเจนิติกอัลกอริทึมแบบง่าย

ที่มา : ดัดแปลงมาจาก [11]

จากรูปที่ 5.3 อธิบายการทำงานของเจนิติกอัลกอริทึมแบบง่ายได้ดังนี้ ส่วนแรกคือขั้นตอนการเตรียมการ ส่วนที่สองคือขั้นตอนการทำงานซึ่งในแต่ละขั้นตอนมีรายละเอียดดังนี้

<u>ส่วนการเตรียมการ</u> เป็นส่วนของการปรับรูปแบบของปัญหาให้เหมาะสมสำหรับการใช้เจนิติกอัลกอริทึม ประกอบด้วย 2 ส่วนดังนี้

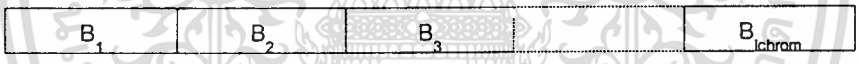
<u>1. กำหนดฟังก์ชันเป้าหมาย</u> เพื่อความสะดวกและง่ายต่อการเข้าใจ จะกำหนดตัวอย่างการหาคำตอบของปัญหาค่าสูงสุดของฟังก์ชัน $y = 15x - x^2$ [11] โดยที่ x เป็นจำนวนเต็มที่อยู่ในช่วง [0,15]

<u>ตัวอย่าง</u> : ฟังก์ชันเป้าหมายคือ $f(x) = 15x - x^2$

กำหนดให้ฟังก์ชันความเหมาะสมคือ $F(x) = 15x - x^2$

ซึ่งคำตอบที่ดีที่สุดคือค่า x ที่มีค่าความเหมาะสมสูงสุด MAX(F(x))

<u>2. กำหนดรูปแบบโครโมโซม</u> รูปแบบของโครโมโซมที่จะใช้กับปัญหานี้ เป็นแบบไบนารี โดยค่าตัวแปรหรือพารามิเตอร์ของปัญหาจะถูกแปลงให้อยู่ในรูปของไบนารีโครโมโซม คือ ประกอบด้วย บิทที่มีค่าเป็น 0 หรือ 1 ซึ่งเป็นค่าในเลขฐานสอง และมีความยาว (Chromosome Length : lchrom) ตามแต่จะกำหนด ซึ่งแสดงด้วยสัญลักษณ์ได้ดังรูปที่ 5.4



รูปที่ 5.4 ตัวอย่างรูปแบบของโครโมโซมซึ่ง $B_i \in [0,1]$

<u>ตัวอย่าง</u> : วิธีการเข้ารหัสแบบไบนารีโดยแปลงค่าพารามิเตอร์ x ให้อยู่ในรูปไบนารี 4 บิท (lchrom = 4) ดังนั้น โครโมโซมของปัญหาจะมีค่าอยู่ระหว่าง 0000 ถึง 1111 ซึ่งเมื่อถอดรหัสแล้วจะมีค่าอยู่ในช่วง 0 ถึง 15

<u>ส่วนการทำงาน</u> รายละเอียดขั้นตอนการทำงานของ เจนิติกอัลกอริทึมแบบง่ายจะเป็นขั้นตอนพื้นฐานแบบง่ายประกอบด้วย 5 ส่วนดังนี้

<u>1. ประชากรรุ่นเก่า</u> (Old Population) เป็นชุดโครโมโซมที่จะถูกคัดเลือกไปเป็นต้นแบบสำหรับสร้างประชากรรุ่นใหม่ (New Population) ในวิวัฒนาการ (Generation : gen) รุ่นต่อไป โดยประชากรเริ่มต้นที่ gen = 0 จะถูกสร้างขึ้นโดยการสุ่มตามจำนวนโครโมโซมในแต่ละรุ่น (Population Size : popsize) ที่กำหนด

ตัวอย่าง :

ลำดับ	โครโมโซม
1	1100
2	0100
3	0001
4	1110
5	0111
6	1001

ชุดโครโมโซมรุ่นนี้เป็นชุดโครโมโซมรุ่นเริ่มต้นที่กำหนดให้ในแต่ละรุ่นประกอบด้วยจำนวนโครโมโซม จำนวน 6 โครโมโซม โดยแต่ละโครโมโซมประกอบด้วย ค่าไบนารี 0 หรือ 1 ที่เกิดจากการสุ่มจำนวน 4 ครั้ง

2. วิเคราะห์ค่าความเหมาะสม เป็นขั้นตอนการถอดรหัสจากรูปแบบโครโมโซมที่กำหนดไว้ เพื่อคำนวณค่าความเหมาะสมตามฟังก์ชันความเหมาะสมของปัญหา ในที่นี้ฟังก์ชันเป้าหมายหรือฟังก์ชันความเหมาะสมคือ $F = 15x - x^2$ ดังนั้น การวิเคราะห์ค่าความเหมาะสมจึงเป็นการถอดรหัสเลขฐานสองของแต่ละโครโมโซมเป็นค่าตัวแปร x และคำนวณค่าความเหมาะสม คือ ค่า $15x - x^2$ ซึ่งจะเห็นได้ว่าชุดโครโมโซมรุ่นเริ่มต้นมีค่าความเหมาะสมเป็น 36, 44, 14, 14, 56 และ 54 ตามลำดับ

ตัวอย่าง :

ลำดับ	โครโมโซม	x	ค่าความเหมาะสม
1	1100	12	36
2	0100	4	44
3	0001	1	14
4	1110	14	14
5	0111	7	56
6	1001	9	54

3. การคัดเลือก เป็นขั้นตอนที่จำลองการคัดเลือกทางธรรมชาติเพื่อสร้าง Mating pool โดยคัดเลือกชุดโครโมโซมรุ่นเก่าให้เป็นโครโมโซมต้นแบบ หรือ โครโมโซมพ่อ-แม่ เพื่อใช้สร้างโครโมโซมรุ่นลูกเป็นรุ่นต่อไป การคัดเลือกของเจเนติอัลกอริทึมแบบง่าย เป็นแบบอ้างอิงค่าความเหมาะสม (Fitness-based Selection) โดยใช้ค่าความเหมาะสมเป็นตัวตัดสินว่า โครโมโซมใดในรุ่นเก่ามีโอกาสจะถูกเลือกเป็นโครโมโซมพ่อ-แม่มากน้อยเพียงใด โครโมโซมที่มีค่าความเหมาะสมที่ดีจะถูกกำหนดน้ำหนักค่าความน่าจะเป็นที่จะถูกเลือกแต่ละครั้งสูง การกำหนดค่าความน่าจะเป็นที่จะถูกเลือกต่อการสุ่มเลือกแต่ละครั้ง (Probability of Selected Value : pselect) ของแต่ละโครโมโซมกำหนดจากค่าความเหมาะสมเทียบกับผลรวมของค่าความเหมาะสมทั้งหมดดังสมการที่ 5.1

$$pselect_i = \frac{F_i}{\sum F} \tag{5.1}$$

ซึ่งสามารถคำนวณค่าที่คาดหวังว่าจะสุ่มได้ (Expected Value : E) ของแต่ละโครโมโซมในแต่ละรุ่น ได้ดังสมการที่ 5.2

$$E_i = pselect_i * popsize = \frac{F_i}{F} \tag{5.2}$$

สำหรับวิธีการสุ่มโครโมโซมต้นแบบของเจนิติกอัลกอริทึมแบบง่ายนั้นเป็นแบบจำลองการหมุนวงล้อถ่วงน้ำหนัก (Roulette Wheel : RW) ซึ่งกำหนดขนาดแต่ละช่องของวงล้อนั้นตามความน่าจะเป็นที่จะสุ่มได้ในแต่ละครั้งของแต่ละโครโมโซมซึ่งมีวิธีการดังนี้

1) หาค่าความเหมาะสมของแต่ละโครโมโซม
2) หาค่าความน่าจะเป็นที่จะสุ่มได้ในแต่ละครั้งของแต่ละโครโมโซม
3) หาค่าความถี่สะสม (q) ของค่าความน่าจะเป็นของแต่ละโครโมโซม ดังสมการที่ 5.3

$$q_i = \sum_{j=1}^{i} pselect_i \tag{5.3}$$

4) สร้างเลขสุ่มจำนวนจริง (r) มีค่าอยู่ในช่วง [0.0,1.0]
5) เลือกโครโมโซมลำดับที่ r ซึ่ง r มีค่าอยู่ระหว่าง q_{i-1} และ q_i

<u>ตัวอย่าง</u> : การคำนวณจากตัวอย่างข้างต้นแสดงดังตารางที่ 5.3

ตารางที่ 5.3 ตัวอย่างการคำนวณค่าของสมการ

ลำดับ	โครโมโซม	x	ค่าความเหมาะสม (F)	ค่าความน่าจะเป็น ($pselect_i$), (%)	จำนวนที่คาดหวัง (E_i)	จำนวนที่สุ่มได้จาก RW
1	1100	12	36	0.165, (16.5)	0.99	1
2	0100	4	44	0.202, (20.2)	1.212	2
3	0001	1	14	0.064, (6.4)	0.384	0
4	1110	14	14	0.064, (6.4)	0.384	0
5	0111	7	56	0.257, (25.7)	1.542	2
6	1001	9	54	0.248, (24.8)	1.488	1
รวม			218	1.000, (100)	6.000	
ค่าเฉลี่ย			36.33	0.167, (16.7)	1.000	
ค่าสูงสุด			56	0.257, (25.7)	1.542	

ที่มา : ดัดแปลงมาจาก [12]

จากตัวอย่างการกำหนดค่าความน่าจะเป็นโดยกำหนดจากค่าความเหมาะสมเทียบกับผลรวมของค่าความเหมาะสมทั้งหมด จะเห็นได้ว่าในการคัดเลือกโครโมโซมต้นแบบจาก 6 โครโมโซมนี้ โอกาสที่จะสุ่มได้โครโมโซมลำดับที่ 1 ต่อการสุ่มแต่ละครั้งเท่ากับ 0.165 หรือ 16.5% และโอกาสที่จะสุ่มได้โครโมโซมลำดับที่ 2, 3, 4, 5 และ 6 ต่อการสุ่มแต่ละครั้งเท่ากับ 0.202, 0.064, 0.064, 0.257 และ 0.248 ตามลำดับ จำนวนโครโมโซมต้นแบบที่สุ่มได้โดยจำลองการหมุนวงล้อแสดงดังตารางที่ 5.4

ตารางที่ 5.4 แสดงโครโมโซมต้นแบบที่สุ่มได้โดยจำลองการหมุนวงล้อ

ลำดับโครโมโซม	1	2	3	4	5	6
ค่าความเหมาะสม (F)	36	44	14	14	56	54
ค่าความน่าจะเป็นที่สุ่มได้แต่ละครั้ง ($pselect_i$)	0.165	0.202	0.064	0.064	0.257	0.248
ความถี่สะสมค่าความน่าจะเป็น (q_i)	0.165	0.367	0.431	0.495	0.752	1.000
สร้างเลขสุ่มในการหมุนวงล้อแต่ละครั้ง (r)	0.888	0.185	0.156	0.532	0.228	0.678
ลำดับโครโมโซมที่ถูกเลือก ($q_{i-1} \leq r \leq q_i$)	6	2	1	5	2	5

ที่มา : ดัดแปลงมาจาก [12]

ซึ่งจำนวนที่สุ่มได้เป็นโครโมโซมต้นแบบใน Mating pool ของแต่ละโครโมโซมเป็น 1, 2, 0, 0, 2 และ 1 ตามลำดับ จะเห็นได้ว่าโครโมโซมลำดับที่ 5 และ 2 มีค่าความเหมาะสมสูง จะมีโอกาสถูกคัดเลือกในจำนวนที่มากที่สุด ส่วนโครโมโซมลำดับที่ 3 และ 4 มีค่าความเหมาะสมต่ำมากจึงมีโอกาสที่จะไม่ถูกเลือก

4. ดำเนินการทางพันธุศาสตร์ เป็นขั้นตอนที่จำลองแบบธรรมชาติทางพันธุกรรม ซึ่งตัวดำเนินการทางพันธุศาสตร์ของเจนิติกอัลกอริทึมแบบง่าย คือ ครอสโอเวอร์และมิวเตชันซึ่งมีรายละเอียดดังนี้

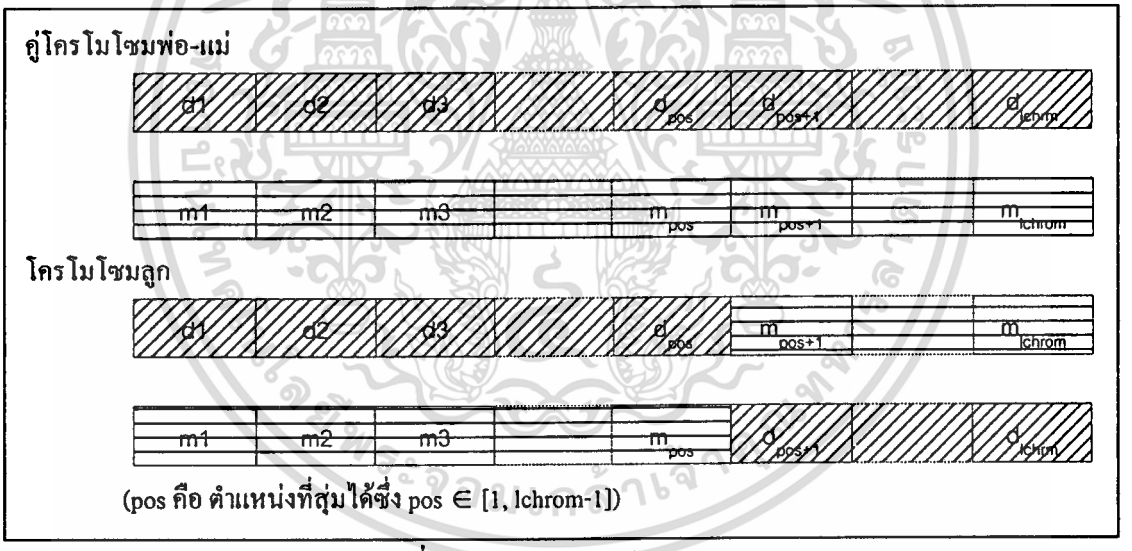
ครอสโอเวอร์

เป็นตัวดำเนินการในการแลกเปลี่ยนส่วนของโครโมโซมพ่อ-แม่ ตามการกำหนดอัตราความน่าจะเป็นของการครอสโอเวอร์ (Probability of Crossover : P_c) เพื่อสร้างชุดโครโมโซมรุ่นใหม่หรือโครโมโซมลูก มีขั้นตอนการทำงานคือ

ขั้นตอนแรก : สุ่มจับคู่โครโมโซมพ่อ-แม่ ใน Mating pool ที่สร้างไว้จากการคัดเลือก

ขั้นตอนที่สอง : สร้างเลขสุ่มจำนวนจริง (r) มีค่าอยู่ในช่วง [0.0, 1,0] โดยถ้า $r \leq P_c$ แล้วโครโมโซมพ่อ-แม่นั้นจึงจะมีการครอสโอเวอร์

ขั้นตอนที่สาม : ครอสโอเวอร์โดยการแลกเปลี่ยนส่วนของคู่โครโมโซมพ่อ-แม่นั้น ซึ่งการครอสโอเวอร์ของเจนิติกอัลกอริทึมแบบง่าย นั้นเป็นการครอสโอเวอร์แบบ 1 จุด (One-point Crossover) แสดงดังรูปที่ 5.5 ดังนี้



รูปที่ 5.5 ครอสโอเวอร์แบบ 1 จุด

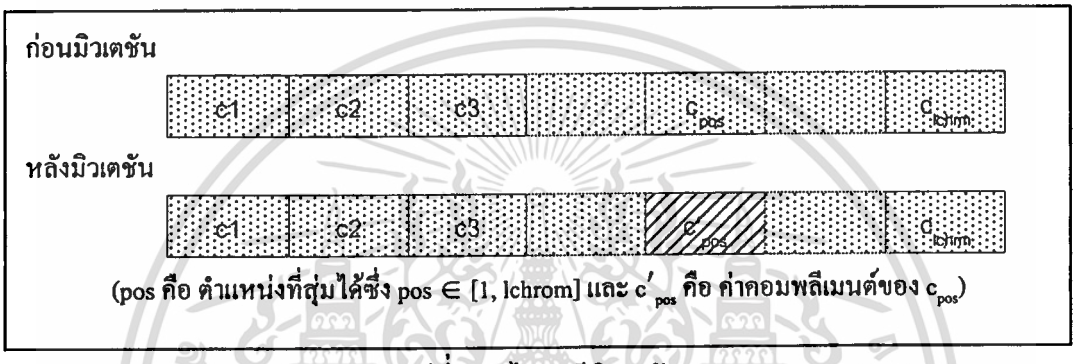
- สุ่มเลือกตำแหน่ง pos ซึ่งเป็นตำแหน่งที่จะครอสโอเวอร์ โดย pos มีค่าอยู่ในช่วง [1, lchrom-1]
- แลกเปลี่ยนค่าในแต่ละบิทของคู่โครโมโซมพ่อ-แม่ตั้งแต่ตำแหน่งที่ pos+1 ถึง lchorm ซึ่งจะทำให้เกิดโครโมโซมลูกใหม่ 2 โครโมโซม

จำนวนการครอสโอเวอร์ในแต่ละรุ่นดำเนินการขึ้นอยู่กับการกำหนดค่า P_c ซึ่งแตกต่างกันในแต่ละปัญหา เช่น ถ้าจำนวนประชากรแต่ละรุ่น popsize เท่ากับ 30 โครโมโซม และกำหนดให้ P_c

= 0.6 แล้วจำนวนการครอสโอเวอร์ในแต่ละรุ่นเท่ากับ P_c*(popsize / 2) = 0.6*(30/2) = 9 ครั้ง (การครอสโอเวอร์หนึ่งครั้งเกิดจากโครโมโซมสองโครโมโซม)

<u>มิวเตชัน</u>

เป็นตัวดำเนินการผ่าเหล่าตัวหนึ่งที่อาจช่วยให้โครโมโซม มีค่าความหมาะสมดีขึ้น หลังจากครอสโอเวอร์ โดยกลับค่าบิทเป็นค่าใหม่ในตำแหน่งบิทที่สุ่มได้ ตามอัตราความน่าจะเป็นของการมิวเตชันในแต่ละบิท (Probability of Mutation : P_m) ที่กำหนด สำหรับการมิวเตชันของเจนิติกอัลกอริทึมแบบง่ายนั้นเป็นแบบไบนารีมิวเตชัน (Binary Mutation) โดยกลับค่าบิทเป็นค่าคอมพลีเมนต์คือจาก 0 เป็น 1 หรือ จาก 1 เป็น 0 ดังรูปที่ 5.6

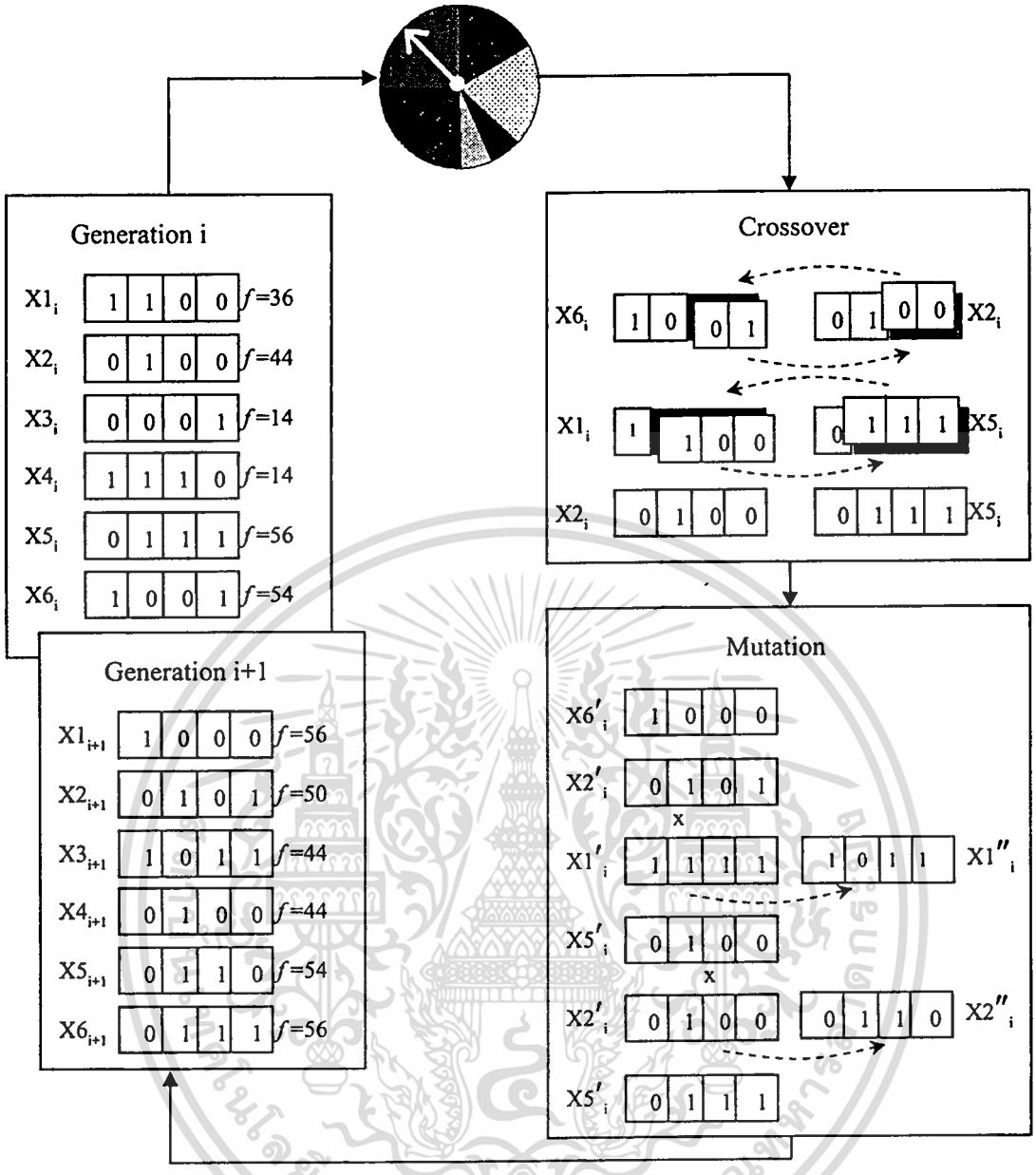


รูปที่ 5.6 ไบนารีมิวเตชัน

จำนวนการมิวเตชันในแต่ละรุ่นขึ้นอยู่กับการกำหนดค่า P_m ซึ่งแตกต่างกันในแต่ละปัญหา เช่น ถ้าจำนวนประชากรแต่ละรุ่น popsize เท่ากับ 30 โครโมโซม ซึ่งแต่ละโครโมโซมประกอบด้วย 4 บิท และกำหนดให้ P_m = 0.02 แล้วจำนวนการมิวเตชันในแต่ละรุ่นเท่ากัน P_m*popsize*lchrom = 0.02*30*4 = 2.4 บิท

<u>5. ประชากรรุ่นใหม่</u> เป็นชุดโครโมโซมลูกที่เกิดจากขั้นตอนของการวิวัฒนาการต่าง ๆ ทั้งหมด ซึ่งประชากรรุ่นใหม่ทั้งหมดที่เกิดขึ้นจะถูกถ่ายทอดกลายเป็นประชากรรุ่นเก่าสำหรับวิวัฒนาการในรุ่นถัดไป ซึ่งเรียกวิวัฒนาการแบบนี้ว่า การถ่ายทอดแบบทั่วไปหรือรีโพรดักชันแบบทั่วไป (General Reproduction) กระบวนการต่าง ๆ จะถูกปฏิบัติซ้ำ ๆ จนกระทั่งถึงรุ่นที่มากที่สุด (Max generation) ที่ต้องการ

วัฏจักรการหาคำตอบตามตัวอย่างที่นำเสนอแสดงได้ดังรูปที่ 5.7



รูปที่ 5.7 วัฏจักรเจนิติกอัลกอริทึมแสดงการหาคำตอบที่ต้องการ

ที่มา : [11]

รูปที่ 5.7 แสดงวัฏจักรการทำงานของเจนิติกอัลกอริทึมจากตัวอย่างการหาคำตอบของปัญหาค่าสูงสุดของฟังก์ชัน y = 15x – x2 โดยที่ x เป็นจำนวนเต็มที่อยู่ในช่วง [0,15] ซึ่งเริ่มตั้งแต่การกำหนดประชากร Generation i แล้วหาค่าฟังก์ชันความเหมาะสมของแต่ละโครโมโซมแล้วนำไปทำการสุ่มเลือกด้วยวิธี Roulette Wheel หลังจากนั้นนำโครโมโซมที่เลือกมาได้มาทำการ Crossover และ Mutation แล้วนำประชากรที่ได้รุ่นใหม่ Generation i+1 ไปแทนที่ประชากรเดิมแล้วทำกระบวนการเดิมต่อไปจนถึงเกณฑ์ที่ต้องการ

5.3.2 การประยุกต์หลักการของเจนิติกอัลกอริทึมสำหรับปรับค่าน้ำหนักแท็ก

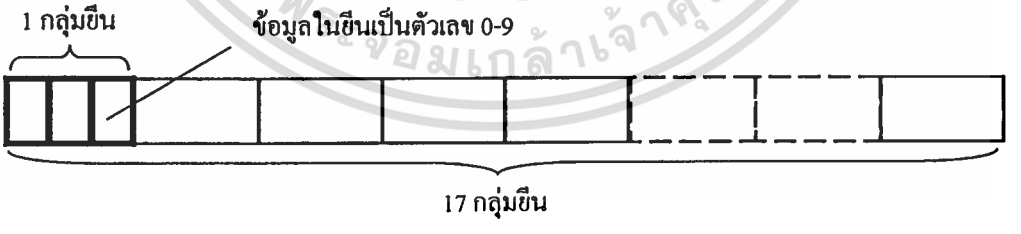
งานวิจัยนี้ใช้ส่วนของโปรแกรมเจนิติกอัลกอริทึมจาก [13] เพื่อนำมาประยุกต์ใช้ในงานวิจัย และมีการประยุกต์เจนิติกอัลกอริทึมเพื่อนำมาปรับค่าน้ำหนักแท็กดังต่อไปนี้

5.3.2.1 ฟังก์ชันความเหมาะสม

ฟังก์ชันความเหมาะสมคำนวณจากการหาค่าเปอร์เซ็นต์ที่ตรงกันของการจัดเรียงลำดับคำดัชนีที่คำนวณจากวิธีคำนวณน้ำหนักคำดัชนีแบบต่าง ๆ กับการจัดเรียงลำดับคำดัชนีที่จัดทำโดยผู้เชี่ยวชาญ โดยวัดเทียบการเรียงลำดับคำดัชนีจากวิธีคำนวณน้ำหนักแบบต่าง ๆ ลำดับที่ 1-5 กับลำดับที่ 1-5 ของการเรียงลำดับคำดัชนีที่จัดทำโดยผู้เชียวชาญแล้วนับจำนวนคำดัชนีว่าตรงกันทั้งหมดกี่คำแล้วนำมาคำนวณเป็นเปอร์เซ็นต์ เช่น มีการจัดเรียงลำดับคำดัชนีจากการคำนวณน้ำหนักคำดัชนีแบบต่าง ๆ ตรงกับคำดัชนีที่จัดเรียงลำดับโดยผู้เชี่ยวชาญจำนวน 4 คำ ซึ่งคิดเป็น (4/5)*100 = 80 เปอร์เซ็นต์ และทำการเปรียบเทียบลักษณนี้ในลำดับที่ 1-10 และ 1-20 หลังจากนั้นนำมาคำนวณค่าเฉลี่ยเปอร์เซ็นต์ความตรงกันและคิดเป็นค่าความเหมาะสมสำหรับโครโมโซมนั้น

5.3.2.2 รูปแบบโครโมโซม

จากตารางที่ 5.1 มีแท็กทั้งหมด 17 แท็กที่ต้องกำหนดค่าน้ำหนักแท็กโดยแทนด้วย 17 กลุ่มยีน และจากการคำนวณค่าเฉลี่ยจำนวนคำในแต่ละเอกสารดังแสดงในภาคผนวก ง ซึ่งมีค่าประมาณ 1356 คำต่อเอกสาร จึงกำหนดให้แต่ละแท็กมีค่าน้ำหนักระหว่าง 0-999 เพื่อให้ใกล้เคียงกับจำนวนคำเฉลี่ยในเอกสารเพื่อให้ค่าน้ำหนักแท็กที่กำหนดขึ้นสามารถครอบคลุมจำนวนคำในแต่ละเอกสารโดยในแต่ละกลุ่มยีนมียีน 3 ยีนโดยยีนแต่ละยีนแทนค่าด้วยเลข 0-9 เพื่อให้ถอดรหัสออกมาแล้วมีค่าระหว่าง 0-999 ดังนั้นโครโมโซมจะมีขนาดทั้งหมด 17*3 = 51 ยีน ดังแสดงในรูปที่ 5.8



รูปที่ 5.8 แสดงรูปแบบโครโมโซม

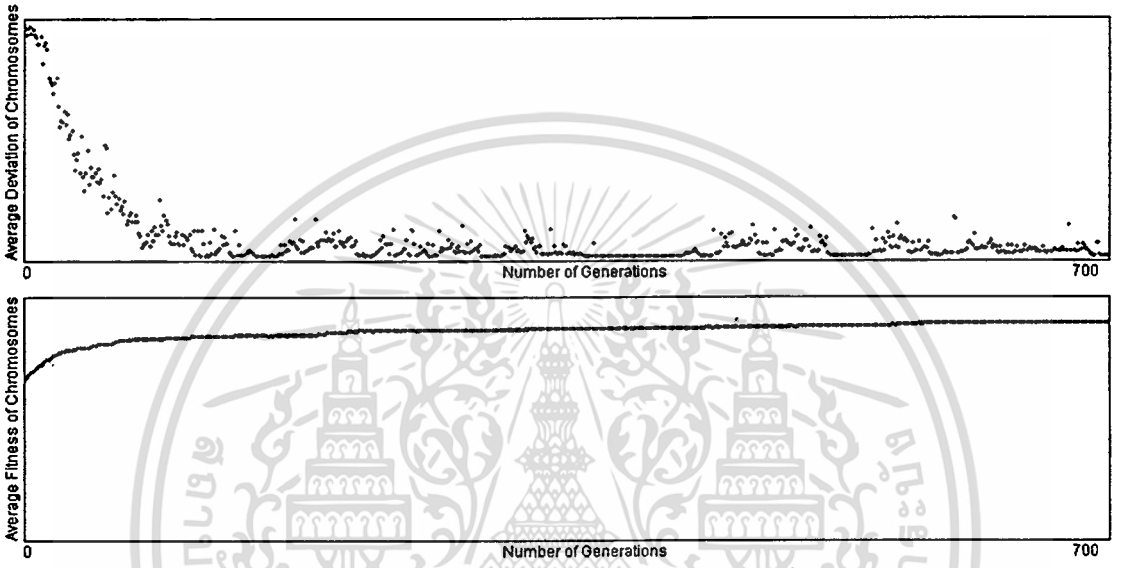
5.3.3 ผลการหาค่าน้ำหนักแท็กโดยใช้เจนิติอัลกอริทึม

หลังจากใช้เจนิติกอัลกอริทึมหาค่าน้ำหนักแท็กที่เหมาะสมสำหรับแต่ละวิธีคำนวณค่าน้ำหนักคำดัชนีที่นำเสนอในบทที่ 4 ได้ค่าน้ำหนักแท็กดังนี้

5.3.3.1 ผลการหาค่าน้ำหนักแท็กโดยใช้เจนิติอัลกอริทึมสำหรับวิธีคำนวณแบบใช้ค่าน้ำหนักแท็กอย่างเดียวเป็นค่าน้ำหนักแท็กของคำดัชนี

5.3.3.1.1 สำหรับวิธีรวมค่าน้ำหนักแท็กที่พบ

จากการทดลองหาค่าน้ำหนักโดยการเรียนรู้ด้วยเจนิติกอัลกอริทึมที่จำนวนประชากรเท่ากับ 100 จำนวนเจนเนอร์เรชันเท่ากับ 700 และจำนวนบทความเท่ากับ 300 แสดงผลการปรับค่าด้วยเจนิติกได้ดังรูปที่ 5.9



รูปที่ 5.9 แสดงผลการหาค่าน้ำหนักแท็กที่ใช้วิธีรวมค่าน้ำหนักแท็กที่พบ

จากรูปที่ 5.9 กราฟแสดงค่าเฉลี่ยส่วนเบี่ยงเบนของโครโมโซมเมื่อทดลองด้วยวิธีรวมค่าน้ำหนักแท็กที่พบจะเห็นได้ว่า กราฟในช่วงแรกมีค่าลดลงจนคงที่ ส่วนกราฟด้านล่างแสดงค่าเฉลี่ยจากฟังก์ชันความเหมาะสมของโครโมโซมเมื่อทดลองด้วยวิธีรวมค่าน้ำหนักแท็กที่พบจะเห็นได้ว่า กราฟในช่วงแรกมีค่าเพิ่มขึ้นจนคงที่ จากกราฟทั้งสองแสดงว่า โครโมโซมที่ได้เป็นโครโมโซมที่ดีที่สุด ซึ่งแสดงค่าน้ำหนักแต่ละแท็กได้ดังตารางที่ 5.5

ตารางที่ 5.5 แสดงค่าน้ำหนักแท็กสำหรับวิธีรวมค่าน้ำหนักแท็กที่พบ

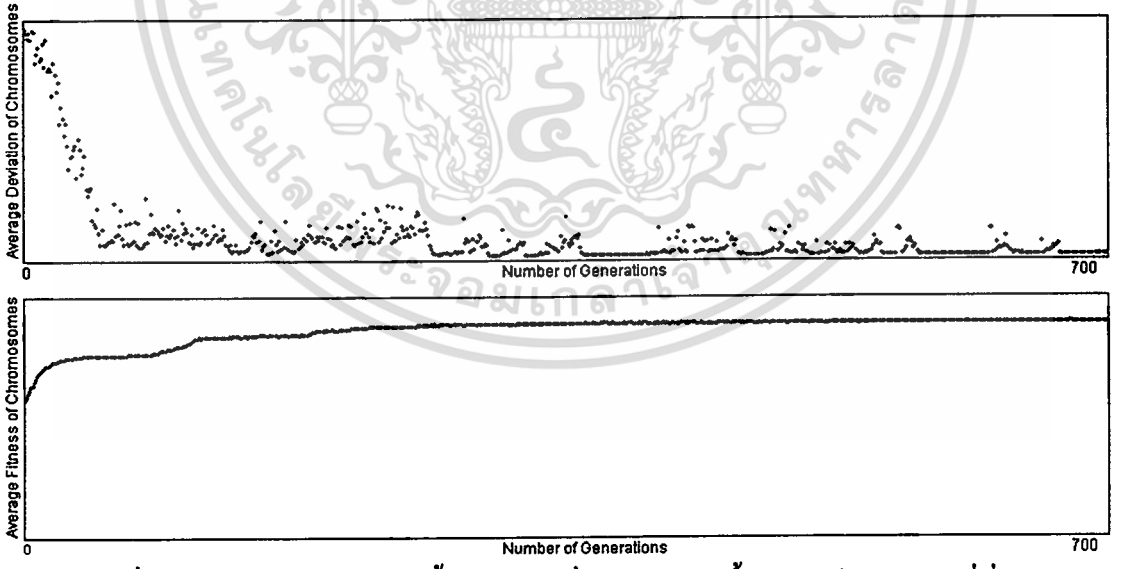
ชื่อแท็ก	ค่าน้ำหนัก
Keyword	997
Title	100
Abstract	1
Introduction	1
Chapter - Name	10

ตารางที่ 5.5 (ต่อ)

ชื่อแท็ก	ค่าน้ำหนัก
Chapter - Content	1
Subchapter - Name	4
Subchapter - Content	2
Experiment - Content	7
SubExperiment - Name	14
SubExperiment - Content	4
Result - Content	2
SubResult - Name	1
SubResult - Content	1
Summary - Content	1
SubSummary - Name	42
SubSummary - Content	4

5.3.3.1.2 สำหรับวิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบ

จากการทดลองหาค่าน้ำหนักโดยการเรียนรู้ด้วยเจนิติกอัลกอริทึมที่จำนวนประชากรเท่ากับ 100 จำนวนเจนเนอร์เรชันเท่ากับ 700 และจำนวนบทความเท่ากับ 300 แสดงผลการปรับค่าด้วยเจนิติกได้ดังรูปที่ 5.10



รูปที่ 5.10 แสดงผลการหาค่าน้ำหนักแท็กที่ใช้วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบ

จากรูปที่ 5.10 กราฟแสดงค่าเฉลี่ยส่วนเบี่ยงเบนของโครโมโซมเมื่อทดลองด้วยวิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบจะเห็นได้ว่า กราฟในช่วงแรกมีค่าลดลงจนคงที่ ส่วนกราฟด้านล่างแสดงค่าเฉลี่ยจากฟังก์ชันความเหมาะสมของโครโมโซมเมื่อทดลองด้วยวิธีรวมค่าน้ำหนักแท็กกับ

ความถี่ที่พบจะเห็นได้ว่า กราฟในช่วงแรกมีค่าเพิ่มขึ้นจนคงที่ จากกราฟทั้งสองแสดงว่าโครโมโซมที่ได้เป็นโครโมโซมที่ดีที่สุด ซึ่งแสดงค่าน้ำหนักแต่ละแท็กได้ดังตารางที่ 5.6

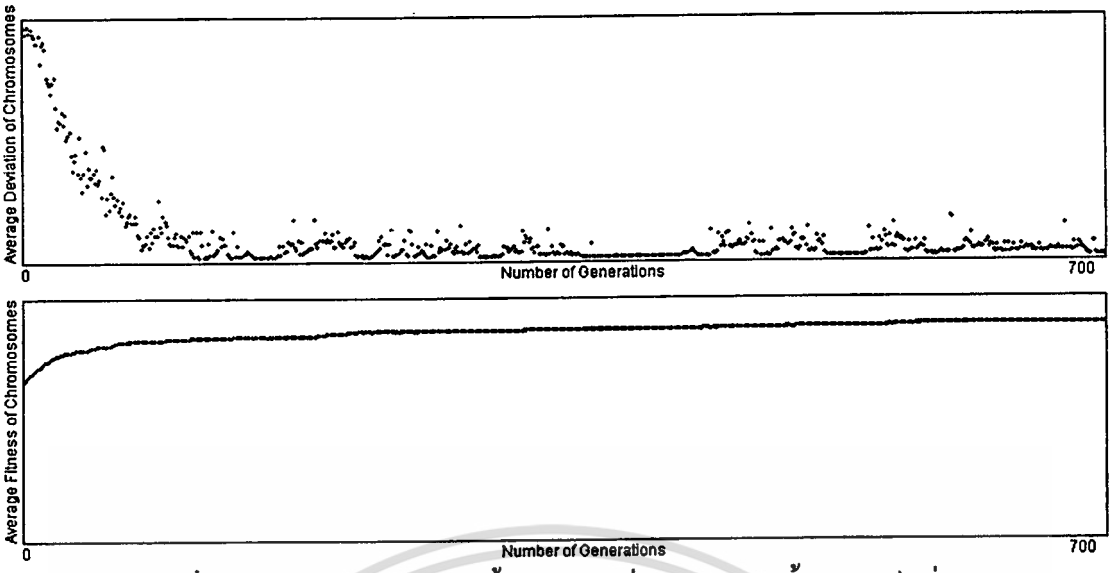
ตารางที่ 5.6 แสดงค่าน้ำหนักแท็กที่ใช้วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบ

ชื่อแท็ก	ค่าน้ำหนัก
Keyword	999
Title	632
Abstract	103
Introduction	1
Chapter - Name	92
Chapter - Content	1
Subchapter - Name	115
Subchapter - Content	6
Experiment - Content	2
SubExperiment - Name	112
SubExperiment - Content	2
Result - Content	6
SubResult - Name	21
SubResult - Content	3
Summary - Content	15
SubSummary - Name	497
SubSummary - Content	228

5.3.3.2 ผลการหาค่าน้ำหนักแท็กโดยใช้เจนิติอัลกอริทึมสำหรับวิธีคำนวณแบบใช้ค่าน้ำหนักแท็กและค่าจำนวนคำในแท็กเป็นค่าน้ำหนักแท็กของคำดัชนี

5.3.3.2.1 สำหรับวิธีรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก

จากการทดลองหาค่าน้ำหนักโดยการเรียนรู้ด้วยเจนิติกอัลกอริทึมที่จำนวนประชากรเท่ากับ 100 จำนวนเจนเนอร์เรชันเท่ากับ 700 และจำนวนบทความเท่ากับ 300 แสดงผลการปรับค่าด้วยเจนิติกได้ดังรูปที่ 5.11



รูปที่ 5.11 แสดงผลการหาค่าน้ำหนักแท็กที่ใช้วิธีรวมค่าน้ำหนักแท็กที่พบและจำนวนของคำในแท็ก

จากรูปที่ 5.11 กราฟแสดงค่าเฉลี่ยส่วนเบี่ยงเบนของโครโมโซมเมื่อทดลองด้วยวิธีรวมค่าน้ำหนักแท็กที่พบและจำนวนของคำในแท็กจะเห็นได้ว่า กราฟในช่วงแรกมีค่าลดลงจนคงที่ ส่วนกราฟด้านล่างแสดงค่าเฉลี่ยจากฟังก์ชันความเหมาะสมของโครโมโซมเมื่อทดลองด้วยวิธีรวมค่าน้ำหนักแท็กที่พบและจำนวนของคำในแท็กจะเห็นได้ว่า กราฟในช่วงแรกมีค่าเพิ่มขึ้นจนคงที่ จากกราฟทั้งสองแสดงว่า โครโมโซมที่ได้เป็นโครโมโซมที่ดีที่สุด ซึ่งแสดงค่าน้ำหนักแต่ละแท็กได้ดังตารางที่ 5.7

ตารางที่ 5.7 แสดงค่าน้ำหนักแท็กที่ใช้วิธีรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก

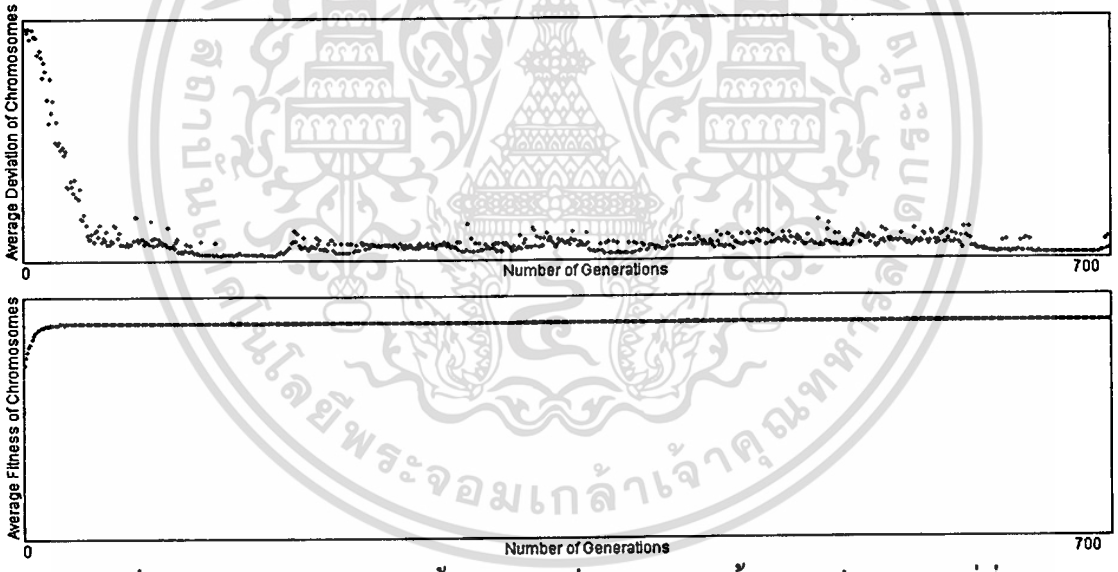
ชื่อแท็ก	ค่าน้ำหนัก
Keyword	998
Title	236
Abstract	13
Introduction	112
Chapter - Name	6
Chapter - Content	11
Subchapter - Name	8
Subchapter - Content	166
Experiment - Content	22
SubExperiment - Name	36
SubExperiment - Content	65

ตารางที่ 5.7 (ต่อ)

ชื่อแท็ก	ค่าน้ำหนัก
Result – Content	21
SubResult - Name	10
SubResult - Content	125
Summary - Content	12
SubSummary - Name	125
SubSummary - Content	105

5.3.3.2.1 สำหรับวิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก

จากการทดลองหาค่าน้ำหนักโดยการเรียนรู้ด้วยเจนิติกอัลกอริทึมที่จำนวนประชากรเท่ากับ 100 จำนวนเจนเนอร์เรชันเท่ากับ 700 และจำนวนบทความเท่ากับ 300 แสดงผลการปรับค่าด้วยเจนิติกได้ดังรูปที่ 5.12



รูปที่ 5.12 แสดงผลการหาค่าน้ำหนักแท็กที่ใช้วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและจำนวนของคำในแท็ก

จากรูปที่ 5.12 กราฟแสดงค่าเฉลี่ยส่วนเบี่ยงเบนของโครโมโซมเมื่อทดลองด้วยวิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและจำนวนของคำในแท็กจะเห็นได้ว่า กราฟในช่วงแรกมีค่าลดลงจนคงที่ ส่วนกราฟด้านล่างแสดงค่าเฉลี่ยจากฟังก์ชันความเหมาะสมของโครโมโซมเมื่อทดลองด้วยวิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและจำนวนของคำในแท็กจะเห็นได้ว่า กราฟในช่วงแรกมีค่า

เพิ่มขึ้นจนคงที่ จากกราฟทั้งสองแสดงว่า โครโมโซมที่ได้เป็นโครโมโซมที่ดีที่สุด ซึ่งแสดงค่าน้ำหนักแต่ละแท็กได้ดังตารางที่ 5.8

ตารางที่ 5.8 แสดงค่าน้ำหนักแท็กสำหรับวิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและจำนวนของคำในแท็ก

ชื่อแท็ก	ค่าน้ำหนัก
Keyword	970
Title	891
Abstract	987
Introduction	153
Chapter - Name	74
Chapter - Content	311
Subchapter - Name	19
Subchapter - Content	369
Experiment - Content	17
SubExperiment - Name	60
SubExperiment - Content	119
Result - Content	8
SubResult - Name	95
SubResult - Content	33
Summary - Content	107
SubSummary - Name	14
SubSummary - Content	861

หลังจากได้ค่าน้ำหนักแท็กสำหรับแต่ละวิธีคำนวณน้ำหนักคำดัชนีแล้วจะได้นำค่าน้ำหนักแท็กที่ได้ไปทดลองหาว่าการให้น้ำหนักแท็กวิธีใดเหมาะสมที่สุด

บทที่ 6
การวัดประสิทธิภาพระบบค้นคืนสารสนเทศและผลการเปรียบเทียบ

6.1 การประเมินประสิทธิภาพของระบบค้นคืนสารสนเทศ

การประเมินประสิทธิภาพของระบบคอมพิวเตอร์ใด ๆ ต้องพิจารณาจาก 2 เรื่อง คือ เวลาที่ใช้ (Time) และหน่วยความจำที่ต้องการในการประมวลผล (Memory space) ระบบที่ใช้เวลาน้อยและใช้หน่วยความจำในการประมวลผลน้อยจะถือว่าระบบนั้นมีประสิทธิภาพดีกว่าอีกระบบหนึ่ง แต่ในงานวิจัยทางด้าน Infromation Retrieval (IR) นั้นนิยมพิจารณาประสิทธิภาพของระบบในอีกรูปแบบหนึ่งคือ นิยมใช้ค่า Recall และ Precision เป็นตัววัดประสิทธิภาพ [2] ของวิธีการค้นคืนเอกสาร

6.1.1 การวัดประสิทธิภาพด้วยค่า Recall

การวัดประสิทธิภาพของระบบ IR จากค่า Recall กระทำโดยป้อนคิวรี่ q จากผู้ใช้ให้กับระบบ S แล้วพิจารณาว่าระบบ S สามารถดึงข้อมูลที่เกี่ยวข้องกับคิวรี่ออกมาจากฐานข้อมูลได้มากน้อยเพียงไร

นิยาม 6.1 : กำหนดให้คอลเลคชัน C เป็นฐานข้อมูลที่เก็บเอกสารทั้งหมดของระบบ กำหนดให้เซต R เป็นเซตของเอกสารทั้งหมดใน C ที่มีความเกี่ยวข้องกับคิวรี่ q (ได้จาก special list) และกำหนดให้เซต A เป็นเซตของเอกสารที่ค้นคืนได้จากระบบ S

การทดสอบระบบด้วยค่า Recall กระทำโดยป้อนคิวรี่ q ให้กับระบบ S แล้วตรวจสอบความสามารถของระบบ S ในการดึงเอกสารจากเซต R มาใส่เซต A ได้มากน้อยเพียงไรดังแสดงในรูปที่ 6.1 และสมการ (6.1)

$$\text{Recall} = \frac{|Ra|}{|R|} \tag{6.1}$$

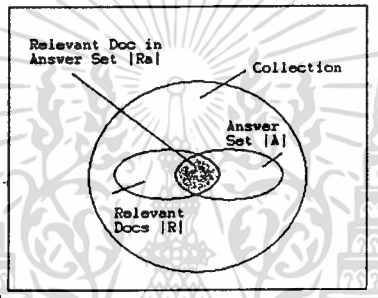
ถ้าหากค่า Recall มีค่ามาก (เข้าใกล้ 1) แสดงว่าระบบ S มีประสิทธิภาพในเชิง Recall ดี คือสามารถค้นคืนเอกสารที่เกี่ยวข้องกับคิวรี่ได้เกือบทั้งหมด

6.1.2 การวัดประสิทธิภาพด้วยค่า Precision

การวัดประสิทธิภาพของระบบ IR จากค่า Precision กระทำโดยป้อนคิวรี่ q จากผู้ใช้ให้กับระบบ S แล้วตรวจสอบว่าเซตคำตอบ A ที่ระบบค้นคืนมาได้นั้นมีเอกสารที่เกี่ยวข้องกับคิวรี่ q อยู่ในเป็นอัตราส่วนเป็นเท่าไร ดังแสดงตามสมการ (6.2)

$$\text{Precision} = \frac{|Ra|}{|A|} \tag{6.2}$$

ถ้าหากค่า Precision มีค่ามาก (เข้าใกล้ 1) แสดงว่าระบบ S มีประสิทธิภาพในเชิง Precision ดี คือ สามารถค้นคืนเอกสารที่เกี่ยวข้องกับคิวรี่ได้ถูกต้องเกือบทั้งหมด



รูปที่ 6.1 แสดงค่า Recall และค่า Precision
ที่มา : [2]

6.1.3 ความสัมพันธ์ระหว่าง Recall กับ Precision

พิจารณาความสัมพันธ์ของค่า Recall และ Precision แล้วจะพบว่าความสัมพันธ์ของค่าทั้งสองจะเป็นในลักษณะแปรผกผันดังแสดงในรูปที่ 6.2 นั่นคือถ้าต้องการให้ระบบมีค่า Recall สูงแล้วระบบจะให้ค่า Precision ต่ำ ในทางกลับกันถ้าต้องการให้ระบบมีค่า Precision สูงแล้วระบบจะให้ค่า Recall ต่ำ ลองพิจารณาความสัมพันธ์ระหว่างค่า Recall และค่า Precision จากตัวอย่างต่อไปนี้

ตัวอย่างที่ 6.1: มีระบบ S ระบบหนึ่งเมื่อป้อนคิวรี่ q ให้กับระบบแล้วระบบจะส่งคืนเซตคำตอบ A ออกมาโดยมี ranking ดังนี้

Ranking for query q:

1.d_{123} •	4. d_6	7. d_{11}	10. d_{25} •	13. d_{250}
2.d_{84}	5. d_8	8. d_{129}	11. d_{38}	14. d_{113}
3.d_{56} •	6. d_9 •	9. d_{187}	12. d_{48}	15. d_3 •

ในขณะที่เซตของเอกสารใน R ที่เกี่ยวข้องกับคิวรี่ q: $R_q=\{d_3, d_5, d_9, d_{25}, d_{39}, d_{44}, d_{56}, d_{71}, d_{89}, d_{123}\}$ สังเกตเอกสารในเซตคำตอบ A ที่มีจุดอยู่ข้างหลัง คือเอกสารที่เกี่ยวข้องกับคิวรี่ q หรือเอกสารนั้นอยู่ใน R_q นั่นเอง

เริ่มพิจารณาเอกสารในเซตคำตอบ A ทีละเอกสาร จะได้

1. เอกสาร d_{123} เป็นเอกสารที่อยู่ใน R_q ถูกจัดลำดับให้อยู่ในลำดับที่ 1 ดังนั้น ณ. จุดนี้ระบบจะให้

   Recall = 1/10 = 10% และ Precision = 1/1 = 100%

2. เอกสาร d_{56} เป็นเอกสารที่อยู่ใน R_q ถูกจัดลำดับให้อยู่ในลำดับที่ 3 ดังนั้น ณ. จุดนี้ระบบจะให้

   Recall = 2/10 = 20% และ Precision = 2/3 = 66%

3. เอกสาร d_9 เป็นเอกสารที่อยู่ใน R_q ถูกจัดลำดับให้อยู่ในลำดับที่ 6 ดังนั้น ณ. จุดนี้ระบบจะให้

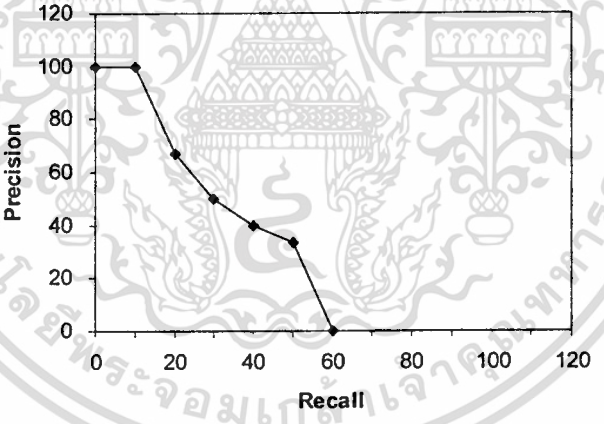
   Recall = 3/10 = 30% และ Precision = 3/6 = 50%

4. เอกสาร d_{25} เป็นเอกสารที่อยู่ใน R_q ถูกจัดลำดับให้อยู่ในลำดับที่ 10 ดังนั้น ณ. จุดนี้ระบบจะให้

   Recall = 4/10 = 40% และ Precision = 4/10 = 40%

5. เอกสาร d_3 เป็นเอกสารที่อยู่ใน R_q ถูกจัดลำดับให้อยู่ในลำดับที่ 15 ดังนั้น ณ. จุดนี้ระบบจะให้

   Recall = 5/10 = 50% และ Precision = 5/15 = 33.33%



รูปที่ 6.2 ความสัมพันธ์ระหว่างค่า Precision ที่ระดับ Recall 11 ค่ามาตรฐาน

ที่มา : [2]

เมื่อนำความสัมพันธ์ดังกล่าวมาเขียนเป็นกราฟ จะได้ความสัมพันธ์ดังแสดงในรูปที่ 6.2 ค่า Precision ในระดับที่ค่า Recall มากกว่า 50% นั้นจะมีค่าเข้าสู่ค่า 0 เพราะไม่มีเอกสารใด ๆ ที่เกี่ยวข้องกับคิวรี่ q ได้ถูกค้นคืนออกมาอีกแล้ว ความสัมพันธ์ระหว่าง Precision และ Recall ปกติจะมี 11 ค่า (แทนที่ 10 ค่า) เรียกว่าค่า Standard Recall (ค่า Recall มาตรฐาน) ซึ่งประกอบด้วยค่า

Recall ที่ระดับ 0%, 10%, 20%,...,100% ที่ Recall ระดับ 0% ค่า Precision หาได้ด้วยวิธี Interpolation (การเพิ่มเติมข้อมูลที่หายไป) ดังที่จะได้อธิบายต่อไปนี้

ในตัวอย่างด้านบน รูปที่ 6.2 ซึ่งแสดงค่า Precision กับ Recall ของคิวรี่ 1 คิวรี่ แต่โดยทั่วไปการวัดประสิทธิภาพของอัลกอริทึมการค้นคืนเอกสาร นั้นวัดจากผลการคิวรี่หลาย ๆ คิวรี่ ที่แตกต่างกัน ทำให้ได้กราฟแสดงค่า Precision กับ Recall ที่แตกต่างกันของแต่ละคิวรี่เพื่อวัดประสิทธิภาพของแต่ละอัลกอริทึมการค้นคืนเอกสารด้วยผลการทดสอบด้วยคิวรี่หลาย ๆ คิวรี่ โดยใช้ค่าเฉลี่ย Precision ที่แต่ละระดับของค่า Recall ดังสมการที่ 6.3

$$\overline{P}(r) = \sum_{i=1}^{N_q} \frac{P_i(r)}{N_q} \tag{6.3}$$

โดยที่ $\overline{P}(r)$ เป็นค่าเฉลี่ย Precision ที่ระดับค่า Recall เท่ากับ r, N_q คือจำนวนของคิวรี่ที่ใช้ และ $P_i(r)$ เป็นค่า Precision ที่ระดับค่า Recall เท่ากับ r สำหรับคิวรี่ลำดับที่ i

เนื่องจากค่า Recall ของแต่ละคิวรี่อาจจะมีค่า Recall ที่แตกต่างกัน ดังนั้นจึงนำวิธี Interpolation มาใช้

<u>ตัวอย่างที่ 6.2</u>: ให้พิจารณาตัวอย่างที่ผ่านมาอีกครั้งหนึ่ง สมมติว่า เซตของเอกสารที่เกี่ยวข้องสำหรับคิวรี่ q ถูกเปลี่ยนใหม่เป็น

$$R_q=\{d_3, d_{56}, d_{129}\}$$

ในกรณีนี้ คำนวนหาค่า Precision กับ Recall ได้ดังนี้

1. เอกสาร d_{56} เป็นเอกสารที่อยู่ใน R_q ถูกจัดลำดับให้อยู่ในลำดับที่ 3 ดังนั้น ณ. จุดนี้ระบบจะให้
   Recall = 1/3 = 33.3% และ Precision = 1/3 = 33.3%
2. เอกสาร d_{129} เป็นเอกสารที่อยู่ใน Rq ถูกจัดลำดับให้อยู่ในลำดับที่ 8 ดังนั้น ณ. จุดนี้ระบบจะให้
   Recall = 2/3 = 66.6% และ Precision = 2/8 = 25%
3. เอกสาร d_3 เป็นเอกสารที่อยู่ใน Rq ถูกจัดลำดับให้อยู่ในลำดับที่ 15 ดังนั้น ณ. จุดนี้ระบบจะให้
   Recall = 3/3 = 100% และ Precision = 3/15 = 20%

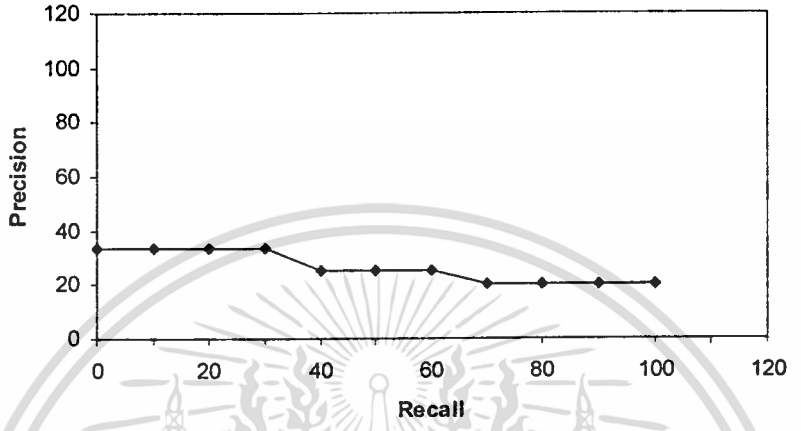
เมื่อนำค่า Precision มาเขียนเป็นกราฟความสัมพันธ์โดยใช้ค่า Standard Recall ด้วยวิธี Interpolate มีวิธีคำนวณดังนี้

ให้ $r_j, j \in \{0, 1, 2,...,10\}$ เป็นค่าอ้างอิงของระดับ Standard Recall ที่ระดับ j (ตัวอย่าง r_5 อ้างอิงระดับ Recall ที่ 50%) ดังนั้น

$$P(r_j) = \max_{r_j \le r \le r_{j+1}} P(r) \tag{6.4}$$

ซึ่งค่า Precision ที่ระดับ j ของ Standard Recall คือ ค่า Precision ที่สูงที่สุดระหว่างระดับ Recall ที่ j และ j+1

จากตัวอย่างที่ผ่านมา สามารถเขียนกราฟ Precision กับ Recall ได้ดังรูปที่ 6.3

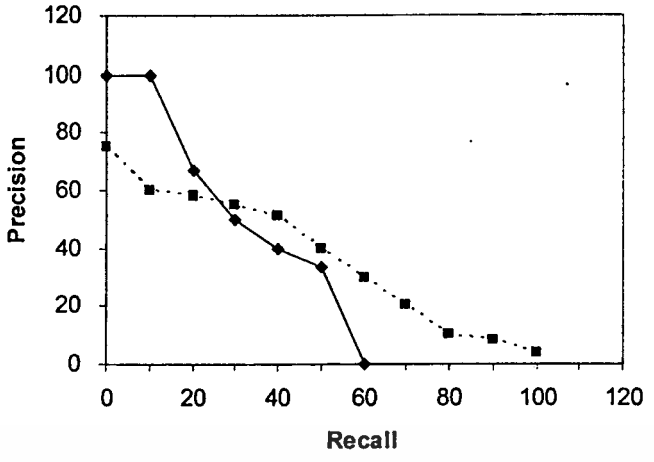


รูปที่ 6.3 แสดงค่า Precision โดยใช้วิธี Interpolate ด้วยระดับ Recall มาตรฐาน 11 ค่า ซึ่งสัมพันธ์กับ $R_q=\{d_3, d_{56}, d_{129}\}$

ที่มา : [2]

จากตัวอย่างที่ผ่านมาสามารถเขียนกราฟ Precision กับ Recall ได้ดังรูปที่ 6.3 ซึ่งอธิบายได้ดังนี้ ที่ระดับ Recall 0%, 10%, 20% และ 30% ค่า Precision เท่ากับ 33.3% (ซึ่งมาจากค่า Precision ที่ระดับ Recall 33.3%) ที่ระดับ Recall 40%, 50% และ 60% ค่า Precision เท่ากับ 25% (ซึ่งมาจากค่า Precision ที่ระดับ Recall 66.6%) ที่ระดับ Recall 70%, 80%, 90% และ 100% ค่า Precision เท่ากับ 20% (ซึ่งมาจากค่า Precision ที่ระดับ Recall 100%)

เส้นกราฟความสัมพันธ์ของ Precision กับ Recall สามารถหาค่าเฉลี่ยที่ได้จากหลาย ๆ คิวรี่ ทำให้ได้รูปกราฟความสัมพันธ์เฉลี่ยของ Precision กับ Recall ซึ่งใช้เปรียบเทียบประสิทธิภาพของอัลกอริทึมการค้นคืนที่ต่างกัน ตัวอย่างเช่น การเปรียบเทียบประสิทธิภาพอัลกอริทึมการค้นคืนวิธีใหม่กับประสิทธิภาพของวิธีเวคเตอร์โมเดล รูปที่ 6.4 แสดงความสัมพันธ์ของค่าเฉลี่ย Precision กับ Recall ของอัลกอริทึมการค้นคืน 2 วิธีที่แตกต่างกัน จากการเปรียบเทียบโดยดูจากกราฟความสัมพันธ์สามารถอธิบายได้ดังนี้ อัลกอริทึมที่หนึ่งมีค่า Precision สูงกว่า ที่ระดับ Recall ที่ต่ำกว่า ขณะที่ อัลกอริทึมที่สองจะดีกว่าที่ระดับ Recall ที่สูงกว่า



รูปที่ 6.4 แสดงค่าเฉลี่ยความสัมพันธ์ Precision กับ Recall ของสองอัลกอริทึมที่แตกต่างกัน ที่มา : [2]

กราฟความสัมพันธ์ระหว่างค่าเฉลี่ย Precision กับ Recall ได้ถูกนำมาใช้ในการวัดประสิทธิภาพของระบบค้นคืนสารสนเทศอย่างกว้างขวาง และมีประโยชน์เนื่องจากค่าทั้งสองวัดได้ทั้งคุณภาพของคำตอบที่ได้และความครอบคลุมของอัลกอริทึมในแนวกว้าง เครื่องมือวัดอีกแบบหนึ่งจะนำค่า Precision และ Recall มารวมกันเป็นค่าเดียวเพื่อให้ง่ายในการวิเคราะห์หาค่าประสิทธิภาพของระบบค้นคืนสารสนเทศ ซึ่งจะได้อธิบายวิธีการคำนวณต่อไป

6.1.4 การวัดประสิทธิภาพระบบค้นคืนสารสนเทศด้วยค่าเพียงค่าเดียว

กราฟค่าเฉลี่ย Precision กับ Recall ใช้สำหรับเปรียบเทียบประสิทธิภาพระบบค้นคืนสารสนเทศ สำหรับอัลกอริทึมการค้นคืน ที่แตกต่างกันที่ทดสอบด้วยตัวอย่างคิวรี่หลายคิวรี่ อย่างไรก็ตามมีบางสถานการณ์ ที่ต้องการเปรียบเทียบประสิทธิภาพการค้นคืนสำหรับแต่ละคิวรี่ เหตุผลมี 2 ข้อ ข้อแรก คือ ค่าเฉลี่ย Precision ของหลาย ๆ คิวรี่ อาจจะบิดเบือนความผิดปกติที่สำคัญในอัลกอริทึมการค้นคืนที่กำลังศึกษา ข้อที่สอง เมื่อเปรียบเทียบระหว่าง 2 อัลกอริทึมซึ่งอาจจะสนใจในการปรับปรุงวิธีการส่วนหนึ่งของทั้งหมดเพื่อให้ประสิทธิภาพดีกว่าอีกวิธีหนึ่งสำหรับแต่ละคิวรี่ที่ใช้ ในสถานการณ์นี้การใช้ค่าเพียงค่าเดียวที่ใช้ Precision ที่ระดับ Recall ที่กำหนดขึ้นมาเพื่อใช้อธิบายประสิทธิภาพจึงไม่เหมาะสม ซึ่งต่อไปจะนำเสนอวิธีที่ใช้หาค่าสรุปเพียงค่าเดียว ดังที่จะได้กล่าวต่อไป

R-Precision

แนวคิดนี้ [2] คือการหาค่าสรุปจากการจัดลำดับในการคำนวณค่า Precision ภายในลำดับที่จำนวนของ R โดยที่ R คือ จำนวนทั้งหมดของเอกสารที่เกี่ยวข้องในเซตคิวรี่ปัจจุบัน (ตัวอย่างเช่น จำนวนเอกสารที่อยู่ในเซต R_q) และ A_R คือ จำนวนเอกสารที่เกี่ยวข้องภายในลำดับที่จำนวนของ R ดังสมการที่ 6.5

$$RP = \frac{A_R}{R} \tag{6.5}$$

ตัวอย่างที่ 6.3: จากตัวอย่างที่ 6.1 ค่า R-precision เท่ากับ 0.4 (เนื่องจาก R = 10 และมีเอกสารที่เกี่ยวข้อง 4 เอกสารใน 10 ลำดับแรกของเอกสารที่ค้นคืนได้)
และตัวอย่างที่ 6.2 ค่า R-precision เท่ากับ 0.33 (เนื่องจาก R = 3 และมีเอกสารที่เกี่ยวข้อง 1 เอกสารใน 3 ลำดับแรกของเอกสารที่ค้นคืนได้)

ค่า R-precision เป็นค่าที่มีประโยชน์สำหรับดูพฤติกรรมของแต่ละอัลกอริทึมสำหรับแต่ละคิวรี่ในการทดลอง การเพิ่มเติมสามารถทำได้โดยการคำนวณค่าเฉลี่ยของ R-precision สำหรับทุกคิวรี่ อย่างไรก็ตาม การใช้ค่าสรุปเพียงค่าเดียวเพื่ออธิบายพฤติกรรมทั้งหมดของอัลกอริทึมการค้นคืนกับหลาย ๆ คิวรี่อาจไม่แม่นยำ

6.2 การหาค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยในงานวิจัย

ในการวัดประสิทธิภาพวิธีการคำนวณน้ำหนักคำดัชนีในงานวิจัยนี้ใช้การวัดค่า Precision เฉลี่ยโดยหาค่า Precision สำหรับแต่ละคิวรี่ที่ระดับ Recall เท่ากับ 25%, 50% และ 75% ด้วยการคำนวณค่า Precision กับ Recall มาตรฐาน 11 ค่าดังที่นำเสนอไว้ด้านบน แล้วคำนวณค่า Precision เฉลี่ยตามสมการที่ 6.3 โดยทดลองกับเอกสาร 300 เอกสารซึ่งเป็นเอกสารชุดเดียวกับที่ใช้ในการหาค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมและใช้ชุดคิวรี่ทั้งหมด 5 ชุด นอกจากค่า Precision เฉลี่ยที่ใช้ในการวัดประสิทธิภาพแล้วยังใช้ค่า R-Precision เฉลี่ยในการวัดประสิทธิภาพร่วมด้วย ซึ่งสามารถคำนวณค่า R-Precision ของแต่ละคิวรี่ได้จากสมการที่ 6.5 หลังจากนั้นจึงนำมาหาค่าเฉลี่ยสำหรับแต่ละวิธี เพื่อใช้ในการเปรียบเทียบต่อไป

6.3 ผลการวัดประสิทธภาพเพื่อหาวิธีให้ค่าน้ำหนักแท็กที่ดีที่สุด

ในงานวิจัยนี้มีวิธีการคำนวณที่ใช้น้ำหนักแท็กทั้งหมด 4 วิธีคือ 1. วิธีการคำนวณแบบรวมค่าน้ำหนักแท็กที่พบ, 2. วิธีการคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบ, 3. วิธีการคำนวณแบบรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก และ 4. วิธีการคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก ผลการทดลองที่นำเสนอได้จากการแทนค่าชุดน้ำหนักต่าง ๆ ที่นำเสนอในบทที่ 5 แล้ววัดประสิทธิภาพด้วยค่า Precision เฉลี่ยและค่า R-Precision กับวิธีทั้ง 4 โดยนำเสนอผลการแทนค่าน้ำหนักแท็กแต่ละวิธีดังผลการทดลองด้านล่างนี้

6.3.1 วิธีคำนวณแบบรวมค่าน้ำหนักแท็กที่พบ (A)

การแทนค่าน้ำหนักแท็กสำหรับวิธีนี้มี 3 ชุดค่าน้ำหนักคือ 1. จากการกำหนดด้วยผู้ใช้, 2. จากการกำหนดด้วยจำนวนคำในแท็ก และ 3. จากการกำหนดด้วยเจนิติกอัลกอริทึม ซึ่งมีผลการทดลองดังนี้

- จากการกำหนดด้วยผู้ใช้

เมื่อแทนค่าน้ำหนักแท็กด้วยค่าน้ำหนักแท็กจากผู้ใช้จำนวน 10 คน ดังที่แสดงไว้ในตารางที่ 5.2 แล้วนำไปหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ของผู้ใช้แต่ละคนได้ผลดังตารางที่ 6.1 – 6.10

ตารางที่ 6.1 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 1

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	30.77
40	33.33	100.00	100.00	100.00	33.33
50	40.00	100.00	100.00	100.00	31.58
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	42.86	100.00	80.00	61.54	39.13
90	41.18	71.43	80.00	64.29	27.03
100	41.18	71.43	80.00	47.62	18.97
Precision ที่ Recall 25%,50%,75%	49.64	100.00	92.50	93.59	33.53
R-Precision	0.286	0.800	0.750	0.700	0.273
Precision เฉลี่ย = 73.85 ; R-Precision เฉลี่ย = 0.562					

ตารางที่ 6.2 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 2

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	30.77
40	33.33	100.00	100.00	100.00	33.33
50	40.00	100.00	100.00	100.00	31.58
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	42.86	100.00	80.00	61.54	39.13
90	41.18	71.43	80.00	64.29	27.78
100	41.18	71.43	80.00	47.62	21.15
Precision ที่ Recall 25%,50%,75%	49.64	100	92.50	93.59	33.53
R-Precision	0.286	0.800	0.750	0.700	0.273
Precision เฉลี่ย = 73.85 ; R-Precision เฉลี่ย = 0.562					

ตารางที่ 6.3 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 3

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	30.77
40	33.33	100.00	100.00	100.00	33.33
50	40.00	100.00	100.00	100.00	31.58
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	42.86	100.00	80.00	61.54	39.13
90	38.89	71.43	80.00	64.29	26.32
100	38.89	71.43	80.00	47.62	19.30
Precision ที่ Recall 25%,50%,75%	49.64	100.00	92.50	93.59	33.53
R-Precision	0.286	0.800	0.750	0.700	0.273
Precision เฉลี่ย = 73.85 ; R-Precision เฉลี่ย = 0.562					

ตารางที่ 6.4 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 4

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	30.77
40	33.33	100.00	100.00	100.00	33.33
50	40.00	100.00	100.00	100.00	31.58
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	42.86	100.00	80.00	61.54	39.13
90	41.18	100.00	80.00	64.29	27.78
100	41.18	100.00	80.00	47.62	20.00
Precision ที่ Recall 25%,50%,75%	49.64	100.00	92.50	93.59	33.53
R-Precision	0.286	1.000	0.750	0.700	0.091
Precision เฉลี่ย = 73.85 ; R-Precision เฉลี่ย = 0.565					

ตารางที่ 6.5 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 5

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	33.33
40	33.33	100.00	100.00	100.00	35.71
50	36.36	100.00	100.00	100.00	31.58
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	42.86	100.00	80.00	61.54	39.13
90	41.18	100.00	80.00	64.29	27.78
100	41.18	100.00	80.00	47.62	19.64
Precision ที่ Recall 25%,50%,75%	48.43	100.00	92.50	93.59	33.95
R-Precision	0.286	1.000	0.750	0.700	0.273
Precision เฉลี่ย = 73.69 ; R-Precision เฉลี่ย = 0.602					

ตารางที่ 6.6 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 6

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	30.77
40	33.33	100.00	100.00	100.00	33.33
50	40.00	100.00	100.00	100.00	31.58
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	46.15	100.00	80.00	57.14	39.13
90	38.89	71.43	80.00	60.00	27.78
100	38.89	71.43	80.00	55.56	22.00
Precision ที่ Recall 25%,50%,75%	50.19	100.00	92.50	92.86	33.53
R-Precision	0.286	0.800	0.750	0.700	0.273
Precision เฉลี่ย = 73.81 ; R-Precision เฉลี่ย = 0.562					

ตารางที่ 6.7 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 7

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	28.57
40	33.33	100.00	100.00	100.00	33.33
50	40.00	100.00	100.00	100.00	31.58
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	42.86	100.00	66.67	61.54	39.13
90	41.18	71.43	66.67	64.29	27.03
100	41.18	71.43	66.67	47.62	19.64
Precision ที่ Recall 25%,50%,75%	49.64	100.00	90.28	93.59	33.16
R-Precision	0.286	0.800	0.750	0.700	0.273
Precision เฉลี่ย = 73.33 ; R-Precision เฉลี่ย = 0.562					

ตารางที่ 6.8 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 8

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	33.33
40	33.33	100.00	100.00	100.00	33.33
50	40.00	100.00	100.00	100.00	33.33
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	42.86	100.00	80.00	61.54	39.13
90	41.18	100.00	80.00	64.29	27.78
100	41.18	100.00	80.00	52.63	21.15
Precision ที่ Recall 25%,50%,75%	49.64	100.00	92.50	93.59	34.54
R-Precision	0.286	1.000	0.750	0.700	0.273
Precision เฉลี่ย = 74.05 ; R-Precision เฉลี่ย = 0.602					

ตารางที่ 6.9 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 9

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	30.00
30	33.33	100.00	100.00	100.00	33.33
40	33.33	100.00	100.00	100.00	33.33
50	40.00	100.00	100.00	100.00	31.58
60	41.67	100.00	75.00	100.00	35.00
70	41.67	100.00	75.00	100.00	38.10
80	42.86	100.00	80.00	61.54	39.13
90	41.18	100.00	80.00	64.29	27.78
100	41.18	100.00	80.00	47.62	20.00
Precision ที่ Recall 25%,50%,75%	49.64	100.00	92.50	93.59	33.95
R-Precision	0.286	1.000	0.750	0.700	0.273
Precision เฉลี่ย = 73.94 ; R-Precision เฉลี่ย = 0.602					

ตารางที่ 6.10 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 10

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	12.50
10	100.00	100.00	100.00	100.00	13.33
20	100.00	100.00	100.00	100.00	18.75
30	33.33	100.00	100.00	100.00	22.22
40	33.33	100.00	100.00	100.00	26.32
50	36.36	100.00	100.00	100.00	28.57
60	41.67	100.00	75.00	100.00	29.17
70	41.67	100.00	75.00	100.00	28.57
80	46.15	100.00	80.00	61.54	30.00
90	38.89	100.00	80.00	64.29	31.25
100	38.89	100.00	80.00	66.67	20.37
Precision ที่ Recall 25%,50%,75%	48.98	100.00	92.50	93.59	26.11
R-Precision	0.286	1.000	0.750	0.700	0.091
Precision เฉลี่ย = 72.24 ; R-Precision เฉลี่ย = 0.565					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยผู้ใช้ได้เท่ากับ 73.65 และค่า R-Precision เฉลี่ยเท่ากับ 0.574

- **จากการกำหนดด้วยจำนวนคำในแท็ก**

เมื่อแทนค่าน้ำหนักแท็กด้วยจำนวนคำในแท็กซึ่งคำนวณจากสมการที่ 5.1 แล้วหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ได้ผลดังตารางที่ 6.11

ตารางที่ 6.11 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักแท็กจำนวนคำในแท็ก

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	36.36
40	37.50	100.00	100.00	100.00	41.67
50	40.00	100.00	100.00	100.00	42.86
60	38.46	100.00	100.00	100.00	46.67
70	38.46	100.00	100.00	100.00	47.06
80	42.86	100.00	80.00	72.73	42.86
90	33.33	100.00	80.00	81.82	35.71
100	33.33	100.00	80.00	71.43	20.37
Precision ที่ Recall 25%,50%,75%	49.80	100.00	96.67	95.46	40.89
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.56 ; R-Precision เฉลี่ย = 0.620					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยจำนวนคำในแท็กได้เท่ากับ 76.56 และค่า R-Precision เฉลี่ยเท่ากับ 0.620

- **จากการกำหนดด้วยเจนิติกอัลกอริทึม**

เมื่อแทนค่าน้ำหนักแท็กด้วยค่าน้ำหนักแท็กที่ได้จากการเรียนรู้เจนิติกด้วยวิธีคำนวณแบบรวมค่าน้ำหนักแท็ก (A) ในตารางที่ 5.6 แล้วหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ได้ผลดังตารางที่ 6.12

ตารางที่ 6.12 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักแท็กเจนิติกอัลกอริทึม

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	50.00	100.00	100.00	100.00	33.33
10	50.00	100.00	100.00	100.00	33.33
20	66.67	100.00	100.00	100.00	37.50
30	37.50	100.00	100.00	100.00	44.44
40	37.50	100.00	100.00	100.00	50.00
50	36.36	100.00	100.00	100.00	54.55
60	35.71	100.00	100.00	100.00	58.33
70	35.71	100.00	100.00	100.00	57.14
80	37.50	100.00	100.00	100.00	60.00
90	38.89	100.00	100.00	39.13	47.62
100	38.89	100.00	100.00	41.67	20.75
Precision ที่ Recall 25%,50%,75%	41.68	100.00	100.00	100.00	51.36
R-Precision	0.286	1.000	1.000	0.800	0.545
Precision เฉลี่ย = 78.61 ; R-Precision เฉลี่ย = 0.726					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมได้เท่ากับ 78.61 และค่า R-Precision เฉลี่ยเท่ากับ 0.726 จากนั้นนำผลการวัดประสิทธิภาพด้วยค่า Precision เฉลี่ย และค่า R-Precision เฉลี่ย ทั้งหมดมาเปรียบเทียบดังแสดงในตารางที่ 6.13

ตารางที่ 6.13 แสดงค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสำหรับวิธี A ด้วยการแทนค่าน้ำหนักวิธีต่าง ๆ

การกำหนดค่าน้ำหนักแท็ก	ค่า Precision เฉลี่ย	ค่า R-Precision เฉลี่ย
ด้วยผู้ใช้	73.65	0.574
ด้วยจำนวนคำในแท็ก	76.56	0.620
ด้วยเจนิติกอัลกอริทึม	78.61	0.726

จากตารางที่ 6.13 จะเห็นว่าการกำหนดค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมให้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสูงสุด

6.3.2 วิธีคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบ (B)

การแทนค่าน้ำหนักแท็กสำหรับวิธีนี้มี 3 ชุดค่าน้ำหนักคือ 1. จากการกำหนดด้วยผู้ใช้, 2. จากการกำหนดด้วยจำนวนคำในแท็ก และ 3. จากการกำหนดด้วยเจนิติกอัลกอริทึม ซึ่งมีผลการทดลองดังนี้

- **จากการกำหนดด้วยผู้ใช้**

เมื่อแทนค่าน้ำหนักแท็กด้วยค่าน้ำหนักแท็กจากผู้ใช้จำนวน 10 คน ดังที่แสดงไว้ในตารางที่ 5.2 แล้วนำไปหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ของผู้ใช้แต่ละคน ได้ผลดังตารางที่ 6.14 – 6.23

ตารางที่ 6.14 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 1

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	25.00
10	100.00	100.00	100.00	100.00	33.33
20	100.00	100.00	100.00	100.00	33.33
30	50.00	100.00	100.00	100.00	33.33
40	50.00	100.00	100.00	100.00	35.71
50	44.44	60.00	100.00	100.00	40.00
60	38.46	60.00	75.00	75.00	38.89
70	38.46	66.67	75.00	77.78	38.10
80	40.00	66.67	80.00	53.33	37.50
90	38.89	62.50	80.00	56.25	35.71
100	38.89	62.50	80.00	47.62	20.00
Precision ที่ Recall 25%,50%,75%	52.89	75.56	92.50	88.52	37.04
R-Precision	0.429	0.600	0.750	0.700	0.273
Precision เฉลี่ย = 69.30 ; R-Precision เฉลี่ย = 0.550					

ตารางที่ 6.15 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 2

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	25.00
10	100.00	100.00	100.00	100.00	28.57
20	100.00	100.00	100.00	100.00	37.50
30	50.00	100.00	100.00	100.00	33.33
40	50.00	100.00	100.00	100.00	38.46
50	44.44	60.00	100.00	100.00	37.50
60	38.46	60.00	75.00	100.00	38.89
70	38.46	66.67	75.00	77.78	38.10
80	40.00	66.67	80.00	53.33	37.50
90	38.89	62.50	80.00	52.94	38.46
100	38.89	62.50	80.00	50.00	21.57
Precision ที่ Recall 25%,50%,75%	52.89	75.56	92.50	88.52	36.90
R-Precision	0.429	0.600	0.750	0.700	0.273
Precision เฉลี่ย = 69.27 ; R-Precision เฉลี่ย = 0.550					

ตารางที่ 6.16 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 3

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	25.00
10	100.00	100.00	100.00	100.00	33.33
20	100.00	100.00	100.00	100.00	33.33
30	50.00	100.00	100.00	100.00	33.33
40	50.00	100.00	100.00	100.00	38.46
50	44.44	60.00	100.00	100.00	40.00
60	41.67	60.00	75.00	100.00	38.89
70	41.67	66.67	75.00	77.78	36.36
80	40.00	66.67	80.00	50.00	37.50
90	38.89	62.50	80.00	52.94	35.71
100	38.89	62.50	80.00	50.00	20.75
Precision ที่ Recall 25%,50%,75%	53.43	75.56	92.50	87.96	36.76
R-Precision	0.429	0.600	0.750	0.700	0.364
Precision เฉลี่ย = 69.24 ; R-Precision เฉลี่ย = 0.568					

ตารางที่ 6.17 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 4

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	33.33
10	100.00	100.00	100.00	100.00	33.33
20	100.00	100.00	100.00	100.00	37.50
30	50.00	100.00	100.00	100.00	33.33
40	50.00	100.00	100.00	100.00	38.46
50	44.44	60.00	100.00	100.00	40.00
60	41.67	60.00	75.00	100.00	41.18
70	41.67	66.67	75.00	77.78	34.78
80	40.00	66.67	80.00	50.00	37.50
90	38.89	62.50	80.00	52.94	38.46
100	38.89	62.50	80.00	50.00	22.45
Precision ที่ Recall 25%,50%,75%	53.43	75.56	92.50	87.96	37.19
R-Precision	0.429	0.600	0.750	0.700	0.273
Precision เฉลี่ย = 69.33 ; R-Precision เฉลี่ย = 0.550					

ตารางที่ 6.18 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 5

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	33.33
10	100.00	100.00	100.00	100.00	28.57
20	100.00	100.00	100.00	100.00	37.50
30	50.00	100.00	100.00	100.00	33.33
40	50.00	100.00	100.00	100.00	38.46
50	44.44	100.00	100.00	100.00	42.86
60	45.45	100.00	75.00	100.00	41.18
70	45.45	80.00	75.00	77.78	38.10
80	40.00	80.00	80.00	50.00	37.50
90	38.89	62.50	80.00	52.94	35.71
100	38.89	62.50	80.00	50.00	20.00
Precision ที่ Recall 25%,50%,75%	54.06	93.33	92.50	87.96	38.69
R-Precision	0.429	0.800	0.750	0.700	0.364
Precision เฉลี่ย = 73.31 ; R-Precision เฉลี่ย = 0.608					

ตารางที่ 6.19 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 6

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	33.33
10	100.00	100.00	100.00	100.00	40.00
20	100.00	100.00	100.00	100.00	37.50
30	50.00	100.00	100.00	100.00	33.33
40	50.00	100.00	100.00	100.00	38.46
50	44.44	60.00	100.00	100.00	40.00
60	41.67	60.00	75.00	100.00	43.75
70	41.67	66.67	75.00	87.50	42.11
80	42.86	66.67	80.00	53.33	37.50
90	36.84	62.50	80.00	50.00	38.46
100	36.84	62.50	80.00	52.63	25.00
Precision ที่ Recall 25%,50%,75%	53.90	75.56	92.50	90.14	38.41
R-Precision	0.429	0.600	0.750	0.700	0.273
Precision เฉลี่ย = 70.10 ; R-Precision เฉลี่ย = 0.550					

ตารางที่ 6.20 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 7

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	25.00
10	100.00	100.00	100.00	100.00	33.33
20	100.00	100.00	100.00	100.00	37.50
30	50.00	100.00	100.00	100.00	33.33
40	50.00	100.00	100.00	100.00	38.46
50	44.44	60.00	100.00	100.00	35.29
60	38.46	60.00	75.00	100.00	35.00
70	38.46	66.67	75.00	77.78	34.78
80	40.00	66.67	80.00	53.33	37.50
90	38.89	62.50	80.00	52.94	38.46
100	38.89	62.50	80.00	50.00	20.75
Precision ที่ Recall 25%,50%,75%	52.89	75.56	92.50	88.52	35.62
R-Precision	0.429	0.600	0.750	0.700	0.273
Precision เฉลี่ย = 69.02 ; R-Precision เฉลี่ย = 0.550					

ตารางที่ 6.21 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 8

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	25.00
10	100.00	100.00	100.00	100.00	33.33
20	100.00	100.00	100.00	100.00	37.50
30	50.00	100.00	100.00	100.00	36.36
40	50.00	100.00	100.00	100.00	41.67
50	57.14	60.00	100.00	100.00	40.00
60	38.46	60.00	75.00	85.71	43.75
70	38.46	66.67	75.00	77.78	42.11
80	40.00	66.67	80.00	53.33	39.13
90	38.89	62.50	80.00	56.25	40.00
100	38.89	62.50	80.00	52.63	22.45
Precision ที่ Recall 25%,50%,75%	57.12	75.56	92.50	88.52	39.18
R-Precision	0.571	0.600	0.750	0.700	0.364
Precision เฉลี่ย = 70.58 ; R-Precision เฉลี่ย = 0.597					

ตารางที่ 6.22 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 9

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	33.33
10	100.00	100.00	100.00	100.00	33.33
20	100.00	100.00	100.00	100.00	37.50
30	50.00	100.00	100.00	100.00	36.36
40	50.00	100.00	100.00	100.00	38.46
50	44.44	100.00	100.00	100.00	40.00
60	41.67	100.00	75.00	75.00	41.18
70	41.67	80.00	75.00	77.78	40.00
80	40.00	80.00	80.00	57.14	39.13
90	38.89	62.50	80.00	60.00	38.46
100	38.89	62.50	80.00	47.62	20.75
Precision ที่ Recall 25%,50%,75%	53.43	93.33	92.50	89.15	38.83
R-Precision	0.429	0.800	0.750	0.700	0.364
Precision เฉลี่ย = 73.45 ; R-Precision เฉลี่ย = 0.608					

ตารางที่ 6.23 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 10

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	33.33
10	100.00	100.00	100.00	100.00	50.00
20	100.00	100.00	100.00	100.00	33.33
30	60.00	100.00	100.00	100.00	36.36
40	60.00	100.00	100.00	100.00	41.67
50	50.00	100.00	100.00	100.00	42.86
60	45.45	100.00	100.00	100.00	46.67
70	45.45	66.67	100.00	87.50	50.00
80	46.15	66.67	80.00	53.33	39.13
90	35.00	71.43	80.00	47.37	41.67
100	35.00	71.43	80.00	50.00	24.44
Precision ที่ Recall 25%,50%,75%	58.60	88.89	96.67	90.14	40.76
R-Precision	0.429	0.600	0.750	0.700	0.364
Precision เฉลี่ย = 75.01 ; R-Precision เฉลี่ย = 0.568					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยผู้ใช้ได้เท่ากับ 70.86 และค่า R-Precision เฉลี่ยเท่ากับ 0.570

- **จากการกำหนดด้วยจำนวนคำในแท็ก**

เมื่อแทนค่าน้ำหนักแท็กด้วยจำนวนคำในแท็กซึ่งคำนวณจากสมการที่ 5.1 แล้วหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ได้ผลดังตารางที่ 6.24

ตารางที่ 6.24 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักแท็กจำนวนคำในแท็ก

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	16.67
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	36.36
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	100.00	100.00	56.25
90	33.33	100.00	100.00	60.00	38.46
100	33.33	100.00	100.00	62.50	22.92
Precision ที่ Recall 25%,50%,75%	51.28	100.00	100.00	100.00	45.26
R-Precision	0.286	1.000	1.000	0.800	0.364
Precision เฉลี่ย = 79.31 ; R-Precision เฉลี่ย = 0.690					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยจำนวนคำในแท็กได้เท่ากับ 79.31 และค่า R-Precision เฉลี่ยเท่ากับ 0.690

- **จากการกำหนดด้วยเจนิติกอัลกอริทึม**

เมื่อแทนค่าน้ำหนักแท็กด้วยค่าน้ำหนักแท็กที่ได้จากการเรียนรู้โดยเจนิติกด้วยวิธีคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบ (B)ในตารางที่ 5.7 แล้วหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ได้ผลดังตารางที่ 6.25

ตารางที่ 6.25 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักแท็กเจนิติกอัลกอริทึม

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	100.00
10	100.00	100.00	100.00	100.00	100.00
20	100.00	100.00	100.00	100.00	100.00
30	60.00	100.00	100.00	100.00	100.00
40	60.00	100.00	100.00	100.00	100.00
50	40.00	100.00	100.00	100.00	100.00
60	38.46	100.00	100.00	100.00	100.00
70	38.46	100.00	100.00	100.00	100.00
80	37.50	100.00	100.00	100.00	69.23
90	35.00	100.00	100.00	37.50	50.00
100	35.00	100.00	100.00	40.00	37.93
Precision ที่ Recall 25%,50%,75%	52.66	100.00	100.00	100.00	94.87
R-Precision	0.429	1.000	1.000	0.800	0.727
Precision เฉลี่ย = 89.51 ; R-Precision เฉลี่ย = 0.791					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมได้เท่ากับ 89.51 และค่า R-Precision เฉลี่ยเท่ากับ 0.791 จากนั้นนำผลการวัดประสิทธิภาพด้วยค่า Precision เฉลี่ย และค่า R-Precision เฉลี่ย ทั้งหมดมาเปรียบเทียบดังแสดงในตารางที่ 6.26

ตารางที่ 6.26 แสดงค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสำหรับวิธี B ด้วยการแทนค่าน้ำหนักวิธีต่าง ๆ

การกำหนดค่าน้ำหนักแท็ก	ค่า Precision เฉลี่ย	ค่า R-Precision เฉลี่ย
ด้วยผู้ใช้	70.86	0.570
ด้วยจำนวนคำในแท็ก	79.31	0.689
ด้วยเจนิติกอัลกอริทึม	89.51	0.791

จากตารางที่ 6.26 จะเห็นว่าการกำหนดค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมให้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสูงสุด

6.3.3 วิธีคำนวณแบบรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก (AL)

การแทนค่าน้ำหนักแท็กสำหรับวิธีนี้มี 2 ชุดค่าน้ำหนักคือ 1. จากการกำหนดด้วยผู้ใช้ และ 2. จากการกำหนดด้วยเจนิติกอัลกอริทึม ซึ่งมีผลการทดลองดังนี้

- **จากการกำหนดด้วยผู้ใช้**

เมื่อแทนค่าน้ำหนักแท็กด้วยค่าน้ำหนักแท็กจากผู้ใช้จำนวน 10 คน ดังที่แสดงไว้ในตารางที่ 5.2 แล้วนำไปหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ของผู้ใช้แต่ละคน ได้ผลดังตารางที่ 6.27– 6.36

ตารางที่ 6.27 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 1

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	36.36
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	46.67
70	38.46	100.00	100.00	100.00	47.06
80	42.86	100.00	80.00	72.73	47.37
90	33.33	100.00	80.00	69.23	37.04
100	33.33	100.00	80.00	71.43	20.75
Precision ที่ Recall 25%,50%,75%	51.28	100.00	96.67	95.45	42.74
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 77.23 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.28 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 2

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	66.67	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	36.36
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	46.67
70	38.46	100.00	100.00	100.00	47.06
80	42.86	100.00	80.00	72.73	45.00
90	33.33	100.00	80.00	69.23	35.71
100	33.33	100.00	80.00	71.43	22.45
Precision ที่ Recall 25%,50%,75%	45.73	100.00	96.67	95.45	42.34
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.04 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.29 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 3

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	66.67	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	47.06
80	42.86	100.00	80.00	72.73	45.00
90	33.33	100.00	80.00	69.23	37.04
100	33.33	100.00	80.00	71.43	21.57
Precision ที่ Recall 25%,50%,75%	45.73	100.00	96.67	95.45	42.95
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.16 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.30 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 4

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	66.67	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	50.00
80	42.86	100.00	80.00	72.73	47.37
90	33.33	100.00	80.00	69.23	37.04
100	33.33	100.00	80.00	71.43	21.57
Precision ที่ Recall 25%,50%,75%	45.73	100.00	96.67	95.45	43.83
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.34 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.31 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 5

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	66.67	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	80.00	72.73	52.94
90	35.00	100.00	80.00	69.23	38.46
100	35.00	100.00	80.00	71.43	21.57
Precision ที่ Recall 25%,50%,75%	45.73	100.00	96.67	95.45	45.32
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.63 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.32 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 6

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	66.67	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	80.00	72.73	52.94
90	35.00	100.00	80.00	64.29	37.04
100	35.00	100.00	80.00	66.67	22.92
Precision ที่ Recall 25%,50%,75%	45.73	100.00	96.67	95.45	45.32
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.63 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.33 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 7

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	36.36
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	42.86
60	38.46	100.00	100.00	100.00	46.67
70	38.46	100.00	100.00	100.00	47.06
80	42.86	100.00	80.00	72.73	42.86
90	33.33	100.00	80.00	69.23	35.71
100	33.33	100.00	80.00	71.43	21.57
Precision ที่ Recall 25%,50%,75%	51.28	100.00	96.67	95.45	40.89
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.86 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.34 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 8

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	66.67	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	36.36
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	80.00	72.73	52.94
90	33.33	100.00	80.00	69.23	38.46
100	33.33	100.00	80.00	71.43	22.45
Precision ที่ Recall 25%,50%,75%	45.73	100.00	96.67	95.45	44.71
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.51 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.35 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 9

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	66.67	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	36.36
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	47.06
80	42.86	100.00	80.00	72.73	50.00
90	33.33	100.00	80.00	69.23	37.04
100	33.33	100.00	80.00	71.43	20.75
Precision ที่ Recall 25%,50%,75%	45.73	100.00	96.67	95.45	43.18
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.21 ; R-Precision เฉลี่ย = 0.620					

ตารางที่ 6.36 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 10

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	66.67	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	44.44	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	47.06
80	42.86	100.00	80.00	72.73	50.00
90	35.00	100.00	80.00	64.29	38.46
100	35.00	100.00	80.00	66.67	22.92
Precision ที่ Recall 25%,50%,75%	45.73	100.00	96.67	95.45	43.78
R-Precision	0.286	1.000	0.750	0.700	0.364
Precision เฉลี่ย = 76.33 ; R-Precision เฉลี่ย = 0.620					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยผู้ใช้ได้เท่ากับ 76.49 และค่า R-Precision เฉลี่ยเท่ากับ 0.620

- **จากการกำหนดด้วยเจนิติกอัลกอริทึม**

เมื่อแทนค่าน้ำหนักแท็กด้วยค่าน้ำหนักแท็กที่ได้จากการเรียนรู้โดยเจนิติกด้วยวิธีคำนวณแบบรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก (AL) ในตารางที่ 5.8 แล้วหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ได้ผลดังตารางที่ 6.37

ตารางที่ 6.37 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักแท็กเจนิติกอัลกอริทึม

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	50.00	100.00	100.00	100.00	20.00
10	50.00	100.00	100.00	100.00	28.57
20	66.67	100.00	100.00	100.00	37.50
30	37.50	100.00	100.00	100.00	44.44
40	37.50	100.00	100.00	100.00	50.00
50	33.33	100.00	100.00	100.00	54.55
60	33.33	100.00	100.00	100.00	58.33
70	33.33	100.00	100.00	100.00	57.14
80	37.50	100.00	100.00	100.00	60.00
90	38.89	100.00	100.00	40.91	47.62
100	38.89	100.00	100.00	43.48	19.30
Precision ที่ Recall 25%,50%,75%	40.28	100.00	100.00	100.00	51.36
R-Precision	0.286	1.000	1.000	0.800	0.545
Precision เฉลี่ย = 78.33 ; R-Precision เฉลี่ย = 0.726					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมได้เท่ากับ 78.33 และค่า R-Precision เฉลี่ยเท่ากับ 0.726 จากนั้นนำผลการวัดประสิทธิภาพด้วยค่า Precision เฉลี่ย และค่า R-Precision เฉลี่ย ทั้งหมดมาเปรียบเทียบดังแสดงในตารางที่ 6.38

ตารางที่ 6.38 แสดงค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสำหรับวิธี AL ด้วยการแทนค่าน้ำหนักวิธีต่าง ๆ

การกำหนดค่าน้ำหนักแท็ก	ค่า Precision เฉลี่ย	ค่า R-Precision เฉลี่ย
ด้วยผู้ใช้	76.49	0.620
ด้วยเจนิติกอัลกอริทึม	78.33	0.726

จากตารางที่ 6.3 จะเห็นว่าด้วยการคำนวณวิธีนี้การกำหนดค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมให้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสูงสุด

6.3.4 วิธีคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก (BL)

การแทนค่าน้ำหนักแท็กสำหรับวิธีนี้มี 2 ชุดค่าน้ำหนักคือ 1. จากการกำหนดด้วยผู้ใช้ และ 2. จากการกำหนดด้วยเจนิติกอัลกอริทึม ซึ่งมีผลการทดลองดังนี้

- **จากการกำหนดด้วยผู้ใช้**

เมื่อแทนค่าน้ำหนักแท็กด้วยค่าน้ำหนักแท็กจากผู้ใช้จำนวน 10 คน ดังที่แสดงไว้ในตารางที่ 5.2 แล้วนำไปหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ของผู้ใช้แต่ละคน ได้ผลดังตารางที่ 6.39– 6.48

ตารางที่ 6.39 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 1

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	40.00	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	75.00	50.00
70	38.46	100.00	100.00	77.78	53.33
80	42.86	100.00	100.00	53.33	56.25
90	33.33	100.00	100.00	56.25	38.46
100	33.33	100.00	100.00	47.62	23.91
Precision ที่ Recall 25%,50%,75%	49.80	100.00	100.00	88.52	45.87
R-Precision	0.286	1.000	1.000	0.700	0.545
Precision เฉลี่ย = 79.13 ; R-Precision เฉลี่ย = 0.726					

ตารางที่ 6.40 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 2

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	16.67
10	100.00	100.00	100.00	100.00	28.57
20	100.00	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	40.00	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	100.00	100.00	56.25
90	33.33	100.00	100.00	60.00	38.46
100	33.33	100.00	100.00	62.50	25.00
Precision ที่ Recall 25%,50%,75%	49.80	100.00	100.00	100.00	45.87
R-Precision	0.286	1.000	1.000	0.800	0.364
Precision เฉลี่ย = 79.13 ; R-Precision เฉลี่ย = 0.690					

ตารางที่ 6.41 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 3

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	28.57
20	100.00	100.00	100.00	100.00	37.50
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	40.00	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	100.00	47.06	56.25
90	33.33	100.00	100.00	47.37	38.46
100	33.33	100.00	100.00	40.00	23.91
Precision ที่ Recall 25%,50%,75%	49.80	100.00	100.00	91.18	46.57
R-Precision	0.286	1.000	1.000	0.700	0.364
Precision เฉลี่ย = 77.51 ; R-Precision เฉลี่ย = 0.670					

ตารางที่ 6.42 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 4

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	28.57
20	100.00	100.00	100.00	100.00	37.50
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	40.00	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	100.00	100.00	56.25
90	33.33	100.00	100.00	56.25	38.46
100	33.33	100.00	100.00	55.56	25.58
Precision ที่ Recall 25%,50%,75%	49.80	100.00	100.00	100.00	46.57
R-Precision	0.286	1.000	1.000	0.800	0.364
Precision เฉลี่ย = 79.27 ; R-Precision เฉลี่ย = 0.690					

ตารางที่ 6.43 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 5

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	33.33
10	100.00	100.00	100.00	100.00	28.57
20	100.00	100.00	100.00	100.00	37.50
30	50.00	100.00	100.00	100.00	44.44
40	50.00	100.00	100.00	100.00	50.00
50	40.00	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	100.00	100.00	52.94
90	33.33	100.00	100.00	50.00	40.00
100	33.33	100.00	100.00	52.63	24.44
Precision ที่ Recall 25%,50%,75%	51.89	100.00	100.00	100.00	46.75
R-Precision	0.429	1.000	1.000	0.800	0.545
Precision เฉลี่ย = 79.73 ; R-Precision เฉลี่ย = 0.755					

ตารางที่ 6.44 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 6

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	20.00
10	100.00	100.00	100.00	100.00	33.33
20	100.00	100.00	100.00	100.00	42.86
30	37.50	100.00	100.00	100.00	44.44
40	37.50	100.00	100.00	100.00	50.00
50	40.00	100.00	100.00	100.00	54.55
60	38.46	100.00	100.00	100.00	53.85
70	38.46	100.00	100.00	100.00	57.14
80	42.86	100.00	100.00	100.00	52.94
90	33.33	100.00	100.00	56.25	43.48
100	33.33	100.00	100.00	58.82	26.19
Precision ที่ Recall 25%,50%,75%	49.80	100.00	100.00	100.00	51.08
R-Precision	0.286	1.000	1.000	0.800	0.545
Precision เฉลี่ย = 80.18 ; R-Precision เฉลี่ย = 0.726					

ตารางที่ 6.45 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 7

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	16.67
10	100.00	100.00	100.00	100.00	25.00
20	100.00	100.00	100.00	100.00	33.33
30	37.50	100.00	100.00	100.00	36.36
40	37.50	100.00	100.00	100.00	41.67
50	40.00	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	100.00	100.00	56.25
90	33.33	100.00	100.00	60.00	38.46
100	33.33	100.00	100.00	62.50	23.91
Precision ที่ Recall 25%,50%,75%	49.80	100.00	100.00	100.00	45.26
R-Precision	0.286	1.000	1.000	0.800	0.364
Precision เฉลี่ย = 79.01 ; R-Precision เฉลี่ย = 0.690					

ตารางที่ 6.46 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 8

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	16.67
10	100.00	100.00	100.00	100.00	28.57
20	100.00	100.00	100.00	100.00	37.50
30	33.33	100.00	100.00	100.00	44.44
40	33.33	100.00	100.00	100.00	41.67
50	40.00	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	100.00	100.00	56.25
90	33.33	100.00	100.00	60.00	40.00
100	33.33	100.00	100.00	62.50	25.00
Precision ที่ Recall 25%,50%,75%	49.11	100.00	100.00	100.00	47.31
R-Precision	0.286	1.000	1.000	0.800	0.364
Precision เฉลี่ย = 79.28 ; R-Precision เฉลี่ย = 0.690					

ตารางที่ 6.47 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 9

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	25.00
10	100.00	100.00	100.00	100.00	28.57
20	100.00	100.00	100.00	100.00	37.50
30	37.50	100.00	100.00	100.00	40.00
40	37.50	100.00	100.00	100.00	41.67
50	40.00	100.00	100.00	100.00	46.15
60	38.46	100.00	100.00	100.00	50.00
70	38.46	100.00	100.00	100.00	53.33
80	42.86	100.00	100.00	100.00	56.25
90	33.33	100.00	100.00	56.25	40.00
100	33.33	100.00	100.00	58.82	25.00
Precision ที่ Recall 25%,50%,75%	49.80	100.00	100.00	100.00	46.57
R-Precision	0.286	1.000	1.000	0.800	0.364
Precision เฉลี่ย = 79.27 ; R-Precision เฉลี่ย = 0.690					

ตารางที่ 6.48 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักผู้ใช้คนที่ 10

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	33.33
10	100.00	100.00	100.00	100.00	33.33
20	100.00	100.00	100.00	100.00	42.86
30	50.00	100.00	100.00	100.00	44.44
40	50.00	100.00	100.00	100.00	50.00
50	40.00	100.00	100.00	100.00	54.55
60	38.46	100.00	100.00	100.00	53.85
70	38.46	100.00	100.00	100.00	57.14
80	40.00	100.00	100.00	100.00	52.94
90	33.33	100.00	100.00	47.37	45.45
100	33.33	100.00	100.00	50.00	26.83
Precision ที่ Recall 25%,50%,75%	51.41	100.00	100.00	100.00	51.08
R-Precision	0.429	1.000	1.000	0.800	0.545
Precision เฉลี่ย = 80.50 ; R-Precision เฉลี่ย = 0.755					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยผู้ใช้ได้เท่ากับ 79.07 และค่า R-Precision เฉลี่ยเท่ากับ 0.706

- **จากการกำหนดด้วยเจนิติกอัลกอริทึม**

เมื่อแทนค่าน้ำหนักแท็กด้วยค่าน้ำหนักแท็กที่ได้จากการเรียนรู้โดยเจนิติกด้วยวิธีคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก (BL) ในตารางที่ 5.9 แล้วหาค่า Precision-Recall แบบ 11 ค่ามาตรฐานสำหรับแต่ละคิวรี่ได้ผลดังตารางที่ 6.49

ตารางที่ 6.49 แสดงค่า Precision-Recall สำหรับแต่ละคิวรี่จากค่าน้ำหนักแท็กเจนิติกอัลกอริทึม

Precision (%) / Recall (%)	Query 1	Query 2	Query 3	Query 4	Query 5
0	100.00	100.00	100.00	100.00	100.00
10	100.00	100.00	100.00	100.00	100.00
20	100.00	100.00	100.00	100.00	100.00
30	42.86	100.00	100.00	100.00	100.00
40	42.86	100.00	100.00	100.00	100.00
50	40.00	100.00	100.00	100.00	85.71
60	35.71	100.00	100.00	100.00	77.78
70	35.71	100.00	100.00	100.00	66.67
80	40.00	100.00	100.00	88.89	52.94
90	33.33	100.00	100.00	45.00	47.62
100	33.33	100.00	100.00	47.62	32.35
Precision ที่ Recall 25%,50%,75%	49.76	100.00	100.00	98.15	81.84
R-Precision	0.429	1.000	1.000	0.800	0.636
Precision เฉลี่ย = 85.95 ; R-Precision เฉลี่ย = 0.773					

หลังจากนั้นนำมาหาค่า Precision เฉลี่ยของการกำหนดค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมได้เท่ากับ 85.95 และค่า R-Precision เฉลี่ยเท่ากับ 0.773 จากนั้นนำผลการวัดประสิทธิภาพด้วยค่า Precision เฉลี่ย และค่า R-Precision เฉลี่ย ทั้งหมดมาเปรียบเทียบดังแสดงในตารางที่ 6.50

ตารางที่ 6.50 แสดงค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสำหรับวิธี BL ด้วยการแทนค่าน้ำหนักวิธีต่าง ๆ

การกำหนดค่าน้ำหนักแท็ก	ค่า Precision เฉลี่ย	ค่า R-Precision เฉลี่ย
ด้วยผู้ใช้	79.07	0.706
ด้วยเจนิติกอัลกอริทึม	85.95	0.773

จากตารางที่ 6.50 จะเห็นว่าด้วยการคำนวณวิธีนี้การกำหนดค่าน้ำหนักแท็กด้วยเจนิติกอัลกอริทึมให้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสูงสุด

6.4 การเปรียบเทียบวิธีการคำนวณน้ำหนักคำดัชนี

ในการเปรียบเทียบวิธีการคำนวณน้ำหนักคำดัชนีแต่ละวิธีเพื่อหาว่าวิธีการคำนวณน้ำหนักคำดัชนีวิธีใดดีที่สุดโดยใช้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยเป็นเครื่องมือวัดประสิทธิภาพ การเปรียบเทียบวิธีการคำนวณน้ำหนักคำดัชนีแต่ละวิธีจะใช้ค่าน้ำหนักแท็กจากการกำหนดค่าน้ำหนักแท็กโดยผู้ใช้แทนการใช้ค่าน้ำหนักแท็กที่ได้จากการกำหนดค่าน้ำหนักแท็กโดยใช้เจนิติกซึ่งเป็นการกำหนดค่าน้ำหนักแท็กที่ให้ประสิทธิภาพดีที่สุด เพราะว่า ผลที่ได้จากการเรียนรู้ด้วยเจนิติกไม่สามารถสรุปได้ว่าค่าน้ำหนักแท็กที่ได้รับเป็นค่าน้ำหนักที่ดีที่สุดในแต่ละวิธีหรือไม่ ตารางที่ 6.51 แสดงผลการเปรียบเทียบแต่ละวิธี

ตารางที่ 6.51 แสดงการเปรียบเทียบค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยของทุกวิธี

วิธีการคำนวณ	ค่า Precision เฉลี่ย	Rank P	ค่า R-Precision เฉลี่ย	Rank RP
A	73.65	3	0.574	3
B	70.86	4	0.570	4
AL	76.49	2	0.620	2
BL	79.07	1	0.706	1

จากผลการทดลองที่ได้จะเห็นว่าวิธีการคำนวณแบบรวมค่าน้ำหนักกับความถี่ที่พบและค่าจำนวนคำในแท็กมีประสิทธภาพดีกว่าวิธีการคำนวณอื่น ๆ รองลงมาคือวิธีการคำนวณแบบรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก ซึ่งทั้งสองวิธีนี้นำค่าจำนวนคำในแท็กร่วมคำนวณกับค่าน้ำหนักแท็กที่กำหนดขึ้น ทำให้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยดีกว่าวิธีที่ไม่ได้นำค่าจำนวนคำในแท็กร่วมคำนวณด้วยทั้งสองวิธีคือ วิธีคำนวณแบบรวมค่าน้ำหนักแท็กที่พบและวิธีคำนวณแบบรวมค่าน้ำหนักแท็กกับความถี่ที่พบ

บทที่ 7
สรุปงานวิจัยและข้อเสนอแนะ

7.1 สรุปงานวิจัย

จากการศึกษาระบบค้นคืนเอกสารที่ใช้สำหรับการค้นคืนข้อมูลในเอกสาร XML พบว่าการคำนวณค่าดัชนีด้วยวิธีเดิมยังไม่เหมาะสมสำหรับการคำนวณค่าคำดัชนีให้กับเอกสาร XML เนื่องจากวิธีเดิมอาศัยเฉพาะความถี่ของคำภายในเอกสารและมองเอกสารที่จัดทำดัชนีเป็นเอกสารที่ไม่มีการให้ความสำคัญกับข้อมูลภายในเอกสาร ไม่เหมือนกับเอกสาร XML ที่สามารถให้ความสำคัญกับข้อมูลภายในเอกสารได้ ในงานวิจัยนี้จึงปรับปรุงการคำนวณค่าดัชนีจากวิธีเวคเตอร์โมเดล ซึ่งเป็นวิธีเดิม โดยใช้ค่าน้ำหนักแท็ก ซึ่งมีวิธีการคำนวณค่าน้ำหนักคำดัชนี 2 วิธี ที่ต่างกันวิธีแรกจะนำค่าน้ำหนักแท็กที่พบในแต่ละแท็กรวมกันก่อนแล้วจึงนำมารวมกับความถี่ของคำที่พบในเอกสารภายหลัง กับวิธีที่ 2 ที่จะคำนวณค่าน้ำหนักแท็กกับค่าความถี่ของคำที่พบในแท็กนั้นแล้วจึงนำค่าที่ได้มารวมกันภายหลัง แล้วปรับปรุง 2 วิธีแรกโดยนำค่าจำนวนคำในแท็กร่วมคิดด้วย ทำให้มีวิธีการคำนวณน้ำหนักคำดัชนีที่ใช้ในการทดลองทั้งหมด 4 วิธี คือ 1. วิธีรวมค่าน้ำหนักแท็กที่พบ 2. วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบ 3. วิธีรวมค่าน้ำหนักแท็กที่พบและค่าจำนวนคำในแท็ก 4. วิธีรวมค่าน้ำหนักแท็กกับความถี่ที่พบและค่าจำนวนคำในแท็ก จากผลการทดลองหาวิธีการกำหนดค่าน้ำหนักแท็กสำหรับแต่ละวิธีคำนวณค่าน้ำหนักคำดัชนี โดยใช้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยเป็นเครื่องมือวัดประสิทธิภาพพบว่าการให้ค่าน้ำหนักแท็กโดยใช้เจนิติกอัลกอริทึมให้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยสูงสุดสำหรับทุก ๆ วิธีคำนวณค่าน้ำหนักคำดัชนีจึงสรุปว่าการให้ค่าน้ำหนักแท็กโดยใช้เจนิติอัลกอริทึมเป็นการให้ค่าน้ำหนักแท็กที่ดีที่สุด และผลการทดลองหาวิธีคำนวณน้ำหนักคำดัชนีที่ดีที่สุดสำหรับ 4 วิธีข้างต้น โดยใช้ค่า Precision เฉลี่ยและค่า R-Precision เฉลี่ยเป็นเครื่องมือวัดประสิทธิภาพ และใช้การให้ค่าน้ำหนักแท็กโดยผู้ใช้เป็นค่าน้ำหนักแท็กที่นำมาทดลองแทนการใช้ค่าน้ำหนักแท็กที่ได้จากการกำหนดค่าน้ำหนักแท็กโดยใช้เจนิติกซึ่งเป็นการกำหนดค่าน้ำหนักแท็กที่ให้ประสิทธิภาพดีที่สุด เพราะว่า ผลที่ได้จากการเรียนรู้ด้วยเจนิติกไม่สามารถสรุปได้ว่าค่าน้ำหนักแท็กที่ได้รับเป็นค่าน้ำหนักที่ดีที่สุด พบว่าวิธีที่ 4 มีประสิทธภาพดีกว่าวิธีการคำนวณอื่น ๆ รองลงมาคือวิธีที่ 3 ซึ่งทั้งสองวิธีนี้นำค่าจำนวนคำในแท็กร่วมคำนวณกับค่าน้ำหนักแท็กที่กำหนดขึ้น ซึ่งดีกว่าวิธีที่ไม่ได้นำค่าจำนวนคำในแท็กร่วมคำนวณคือวิธีที่ 1 และวิธีที่ 2 ตามลำดับ

7.2 ข้อเสนอแนะ

1. เอกสารที่ใช้ในการทดสอบเป็นบทความทางวิชาการที่ผู้วิจัยได้จัดเก็บรวมรวมขึ้นเอง และได้จัดเรียงลำดับคำสำคัญสำหรับแต่ละเอกสารขึ้น รวมถึงคำค้นที่ใช้ในการทดสอบหาค่า Precision และ Recall ขึ้นเอง ซึ่งถือว่ายังไม่เป็นมาตรฐานพอ และจำนวนชุดเอกสารที่ใช้ทดสอบน้อย ในโอกาสต่อไปผู้วิจัยจะทดสอบโมเดลต่าง ๆ กับฐานข้อมูลมาตรฐานที่ได้มีการจัดเก็บและคัดเลือกชุดคำค้นไว้โดยผู้เชี่ยวชาญ และเป็นฐานข้อมูลมาตรฐานที่นิยมใช้ในงานระบบค้นคืนสารสนเทศ
2. การตัดคำในงานวิจัยนี้ใช้โปรแกรมตัดคำที่มีประสิทธิภาพที่ไม่ดีเท่าที่ควรทำให้คำภาษาไทยที่ได้จากการตัดคำมีข้อผิดพลาดอยู่มาก เนื่องจากคำภาษาอังกฤษที่แปลเป็นภาษาไทยทำให้โปรแกรมไม่สามารถตัดคำได้ถูกต้อง
3. จำนวนชุดค่าน้ำหนักแท็กจากผู้ใช้ที่นำมาทดลองมีจำนวนน้อย ในโอกาสต่อไปควรเพิ่มจำนวนชุดค่าน้ำหนักแท็กจากผู้ใช้ให้มากขึ้น

7.3 งานวิจัยในอนาคต

1. การวิจัยในอนาคตควรพัฒนาการตัดคำภาษาไทยให้สามารถตัดคำได้ถูกต้องและรองรับกับคำหรือกลุ่มคำใหม่ ๆ เพื่อเพิ่มประสิทธิภาพของการหาคำดัชนีให้ดีขึ้น เนื่องจากการประมวลผลคำได้ผลการทำงานไม่ดีเท่าที่ควรจากกลุ่มคำที่เป็นคำแปลจากภาษาอังกฤษทำให้การตัดคำประเภทนี้ได้ผลไม่ดีเท่าที่ควร
2. การพัฒนาด้านความเร็วในการค้นคืนข้อมูลเพื่อให้ได้คำตอบในการค้นคืนเอกสารที่รวดเร็วขึ้น
3. การวิจัยการหาค่าน้ำหนักคำนวณดัชนีด้วยวิธีอื่น ๆ เช่น หาค่าน้ำหนักคำดัชนีจากค่าความห่างของคำดัชนีที่พบกับคำสำคัญอื่น ๆ หรือ การหาค่าน้ำหนักคำดัชนีจากการวิเคราะห์ความหมายของคำ

เอกสารอ้างอิง

[1] Tim Bray, Jean Paoli and C. M. Sperberg-McQueen. "Extensible Markup Language (XML) 1.0 W3C Recommendation." [Online] .Available : http://www.w3.org/TR/1998/ REC-xml-19980 210.

[2] Ricardo Baeza-Yates, Berthier Ribeiro-Neto. Modern Information Retrieval. 1st ED. Addison Wesley. 1999

[3] Gerad Salton., Christopher Buckley. "Term-weighting Approaches in Automatic Text Retrieval" Information Processing & Management, Vol. 24, No. 5, 1988. pp. 513-523.

[4] สราวุธ อ้อยศรีสกุล. เริ่มคิด-เริ่มสร้าง-เริ่มใช้ XML. พิมพ์ครั้งที่ 1. กรุงเทพ:H.N. Group. 2544

[5] IBM. "DB2 UDB." [Online]. Available: http://www-306.ibm.com/software/data/ 2005

[6] IBM. "DB2 XML extender." [Online]. Available: http://www-306.ibm.com/software/data/db2/extenders/xmlext/. 2005

[7] วุฒิชัย ปิยะพันธวงศ์. "การสืบค้นข้อมูลอิงแบบจำลองออปเจคสำหรับการทำดัชนีเอกสาร XML." วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์. 2545.

[8] ธนุสศักดิ์ ธัญญสิริ. "การออกแบบและพัฒนาระบบต้นแบบสำหรับการสืบค้นข้อมูลสารสนเทศภาษาไทยด้วยดัชนีหลายระดับ." วิศวกรรศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย มหาวิทยาลัย เกษตรศาสตร์. 2543.

[9] Evangelos Kotsakis. "Structured Information Retrieval in XML documents." SAC 2002: 9th Annual Workshop on Selected Areas in Cryptography, Madrid, Spain, 2002. pp 663-667.

[10] ไพฑูรย์ ศรีนิล. "การใช้จีนีติกอัลกอริทึมและยูสเซอร์โปรไฟล์เพื่อการสืบค้นสารสนเทศจาก WWW." วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2545

[11] Michael Negnevitsky. Artificial Intelligence A Guide to Intelligent Systems. 1st ED. Addison Wesley. 2002.

[12] กริช สมกันธา. "การแก้ไขข้อผิดพลาดของตัวอักษรที่ได้จาก OCR ภาษาไทยด้วยเจเนติกอัลกอริทึม." วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2546

[13] DNJ Online. "Genetic algorithms." [Online]. Available : http://www.dnjonline.com/default.aspx. 2005

[14] Luis Ennser, Christophe Delporte, Mikio Oba and K. M. Sunil. Integrating XML with DB2 XML Extender and DB2 Text Extender. 1st ED. International Business Machines Corporaton. 2000.

[15] IBM. XML Extender Administration and Programming. International Business Machines Corporaton. 2000.

[16] เกียรติณรงค์ ทองประเสริฐ , วิศิษฏ์ หิรัญกิตติ. "การจัดเรียงลำดับคำดัชนีในเอกสารเอ็กซ์เอ็มแอลโดยอาศัยค่าน้ำหนักของแท็ก" วารสารพระจอมเกล้าลาดกระบัง, ปีที่ 12, ฉบับที่ 1, เมษายน 2547. หน้า 9-18.

ภาคผนวก ก
โครงสร้างบทความที่ใช้ในการทดลอง

โครงสร้างบทความประกอบด้วยส่วนต่าง ๆ แสดงด้วย DTD ได้ดังนี้

1 : <!ELEMENT paper (title*, creator*, address*, abstract*, keyword*, intro*, chapter*, experiment*, chapter*, result*, chapter*, summary*, chapter*, acknowledgements*, chapter*, biography*, historycreator*, paperall*, top5*, top10*, top20*)>

2 : <!ELEMENT title (#PCDATA)>

3 : <!ELEMENT creator (#PCDATA)>

4 : <!ELEMENT address (#PCDATA)>

5 : <!ELEMENT abstract (#PCDATA)>

6 : <!ELEMENT keyword (#PCDATA)>

7 : <!ELEMENT intro (#PCDATA)>

8 : <!ELEMENT chapter (name, content*, subchapter*)>

9 : <!ELEMENT name (#PCDATA)>

10 : <!ELEMENT content (#PCDATA | subchapter)*>

11 : <!ELEMENT subchapter (name, content)>

12 : <!ELEMENT result (subchapter | content)*>

13 : <!ELEMENT experiment (subchapter | content)*>

14 : <!ELEMENT summary (subchapter | content)*>

15 : <!ELEMENT acknowledgements (#PCDATA)>

16 : <!ELEMENT biography (#PCDATA)>

17 : <!ELEMENT historycreator (#PCDATA)>

18** : <!ELEMENT paperall (#PCDATA)>

19** : <!ELEMENT top5 (#PCDATA)>

20** : <!ELEMENT top10 (#PCDATA)>

21** : <!ELEMENT top20 (#PCDATA)>

หมายเหตุ ** เป็นแท็กที่กำหนดขึ้นเพื่อใช้ในการทดลองโปรแกรม

ซึ่งสามารถอธิบายได้ดังนี้

แท็ก paper ใช้เก็บข้อมูลของบทความทั้งหมด ซึ่งประกอบด้วยแท็ก title, creator, address, abstract, keyword, intro, chapter, experiment, chapter, result, chapter, summary, chapter, acknowledgements, chapter, biography, historycreator, paperall, top5, top10 และ top20

แท็ก title ใช้เก็บชื่อของบทความทั้งภาษาไทยและภาษาอังกฤษ

แท็ก creator ใช้เก็บชื่อและนามสกุลผู้แต่ง

แท็ก address ใช้เก็บที่อยู่ของผู้แต่ง

แท็ก abstract ใช้เก็บบทคัดย่อทั้งภาษาไทยและภาษาอังกฤษ

แท็ก keyword ใช้เก็บคำสำคัญของบทความ

แท็ก intro ใช้เก็บส่วนของบทนำ

แท็ก chapter ใช้เก็บส่วนเนื้อหาแต่ละหัวข้อที่ไม่ใช่เนื้อหาในหัวข้อ ผลการทดลอง การทดลอง และสรุปผลการทดลอง ซึ่งประกอบด้วยแท็ก name, content และ subchapter

แท็ก name ใช้เก็บชื่อของหัวข้อ

แท็ก content ใช้เก็บเนื้อหาของหัวข้อ รวมทั้งเนื้อหาหัวข้อย่อย ซึ่งประกอบด้วย subchapter

แท็ก subchapter ใช้เก็บส่วนหัวข้อย่อยของหัวข้อใหญ่ ซึ่งประกอบด้วย name และ content

แท็ก result ใช้เก็บส่วนของผลการทดลอง ซึ่งประกอบด้วย subchapter และ content

แท็ก experiment ใช้เก็บส่วนการทดลองซึ่งอธิบายวิธีการทดลองต่าง ๆ ซึ่งประกอบด้วย subchapter และ content

แท็ก summary ใช้เก็บส่วนสรุปผล หรือการวิเคราะห์วิจารณ์ ซึ่งประกอบด้วย subchapter และ content

แท็ก acknowledgements ใช้เก็บส่วนของกิตติกรรมประกาศ

แท็ก biography ใช้เก็บส่วนเอกสารอ้างอิง

แท็ก historycreator ใช้เก็บส่วนของประวัติผู้แต่ง

แท็ก paperall ใช้เก็บส่วนของบทความทั้งหมด ซึ่งแท็กนี้ใช้ในการทดลอง

แท็ก top5 ใช้เก็บคำสำคัญ 5 คำแรกที่ผู้เขียนคิดว่าเป็นคำสำคัญของบทความนี้เรียงตามลำดับ 5 คำ ซึ่งแท็กนี้ใช้ในการทดลอง

แท็ก top10 ใช้เก็บคำสำคัญ 10 คำแรกที่ผู้เขียนคิดว่าเป็นคำสำคัญของบทความนี้เรียงตามลำดับ 10 คำ ซึ่งแท็กนี้ใช้ในการทดลอง

แท็ก top20 ใช้เก็บคำสำคัญ 20 คำแรกที่ผู้เขียนคิดว่าเป็นคำสำคัญของบทความนี้เรียงตามลำดับ 20 คำ ซึ่งแท็กนี้ใช้ในการทดลอง

ตัวอย่างรูปแบบเอกสาร XML ที่ใช้ในการทดลอง

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE paper SYSTEM "D:\Paper.dtd">

<paper>

<title>อิทธิพลของขนาดของแกลบต่อคุณลักษณะการเผาไหม้</title>

<title>Effect of Rice Husk Sizes on Combustion Characteristics</title>

<creator>สมศักดิ์ โพธิ์ถวิลเกียรติ์ นิวัติ พิริยะรุ่งโรจน์ พงษ์เจต พรหมวงศ์</creator>

<address>ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีมหานคร</address>

<abstract>บทความนี้นำเสนอการศึกษาทดลองถึงพฤติกรรมการเผาไหม้ของขนาดเชื้อเพลิงแกลบในเตาเผาแบบวอร์เทค ขนาดของเชื้อเพลิงแกลบที่ใช้... The paper presents the experimental study of rice husk particle size influence on combustion characteristics in a vortex combustor. The fuel particles used in this experiment have two sizes in the ranges of </abstract>

<keyword>เตาวอร์เทค,ขนาดเชื้อเพลิงแกลบ,การเผาไหม้,Vortex combustor, rice husk fuel sizes,combustion</keyword>

<intro>ปัจจุบันความต้องการใช้พลังงานมีอัตราส่วนที่เพิ่มขึ้นโดยเฉพาะในภาคโรงงานอุตสาหกรรม พลังงานส่วนใหญ่นั้นได้จากการเผาไหม้เชื้อเพลิงฟอสซิลง...</intro>

<chapter>

<name>2. เครื่องมือและอุปกรณ์การทดลอง</name>

<content>การทดลองการเผาไหม้ในเตาเผาแบบวอร์เทคใช้เชื้อเพลิงแกลบในการทดสอบอยู่สองขนาดคือ 0.84-1.00 และ 1.19-1.41 มม.ความชื้นที่ 9.2% ... </content>

</chapter>

<experiment>

<content>ก่อนการทดลองต้องทำการอุ่นเตาเผาวอร์เทคโดยใช้เชื้อเพลิงก๊าซ LPG จนเตามีอุณหภูมิประมาณ 400 จึงเริ่มป้อนแกลบ ... </content>

</experiment>

<result>

<content>จากการทดลองการวัดการกระจายอุณหภูมิภายในเตาทั้งสิ้น 7 ตำแหน่งตามแนวแกนที่ตำแหน่ง x=0.1, 0.2, 0.3, 0.4, 0.5, 0.6, และ0.7 เมตร</content>

<subchapter>

<name>4.1 อิทธิพลของค่า Equivalence Ratios</name>

<content>ลักษณะการกระจายอุณหภูมิภายในเตาเผาแบบวอร์เทค สำหรับ และ โดยใช้เชื้อเพลิงแกลบขนาด 0.84-1.00 มม. ...</content>

</subchapter>

<subchapter>

<name>4.2 อิทธิพลของปริมาณอัตราการไหลของอากาศทุติยภูมิต่ออากาศทั้งหมด</name>

<content>อัตราการไหลของอากาศทุติยภูมิต่ออากาศทั้งหมด เป็นค่าที่บ่งชี้ถึงปริมาณการเกิดการไหลวนภายในเตาเผามากขึ้น... </content>

</subchapter>

<subchapter>

<name>4.3 อิทธิพลของขนาดเชื้อเพลิงแกลบ</name>

<content>ขนาดของเชื้อเพลิงแกลบเป็นปัจจัยหนึ่งที่สำคัญคือ ขนาดของเชื้อเพลิงที่แตกต่างกันพื้นที่ในการคลุกเคล้าระหว่างอากาศกับ ... </content>

</subchapter>

</result>

<summary>

<content>จากการทดลองพบว่าเมื่อปริมาณอากาศที่ใช้เท่ากับปริมาณอากาศทางทฤษฎี ผลของการกระจายอุณหภูมิภายในเตาเผาวอร์เทค... </content>

</summary>

<acknowledgements>บทความนี้สำเร็จไปได้ด้วยดีต้องขอขอบคุณ สำนักงานกองทุนสนับสนุนงานวิจัย (สกว.) และภาควิชาวิศวกรรม เครื่องกล </acknowledgements>

<biography> [1] สุพจน์ น้ำนำโชค "การเผาไหม้ขี้เลื่อยในห้องเผาไหม้แบบไซโคลนชนิดอากาศเข้าหลายช่องทาง" วิทยานิพนธ์ ... </biography>

<paperall>อิทธิพลของขนาดของแกลบต่อคุณลักษณะการเผาไหม้

Effect of Rice Husk Sizes on Combustion Characteristics

สมศักดิ์ โพธิ์ถวิลเกียรติ์ นิวัติ พิริยะรุ่งโรจน์* พงษ์เจต พรหมวงศ์

นักศึกษาปริญญาโท อาจารย์ รองศาสตราจารย์

ภาควิชาวิศวกรรมเครื่องกล...

บทคัดย่อ

บทความนี้นำเสนอการศึกษาทดลองถึงพฤติกรรมการเผาไหม้ของขนาด...

Abstract

The paper presents the experimental study of rice husk particle sizet…

คำสำคัญ: เตาวอร์เทค, ขนาดเชื้อเพลิงแกลบ, การเผาไหม้

Key words: Vortex combustor, rice husk fuel sizes, combustion

1. บทนำ

ปัจจุบันความต้องการใช้พลังงานมีอัตราส่วนที่เพิ่มขึ้นโดยเฉพาะ...

2. เครื่องมือและอุปกรณ์การทดลอง

การทดลองการเผาไหม้ในเตาเผาแบบวอร์เทคใช้เชื้อเพลิงแกลบในการทดสอบอยู่...

3. การทดลอง

ก่อนการทดลองต้องทำการอุ่นเตาเผาวอร์เทคโดยใช้เชื้อเพลิงก๊าซ...

3.1 ทำการปรับอัตราการไหลของอากาศให้ค่า Equivalence ratio เท่ากับ 0.8...

3.2 ป้อนเชื้อเพลิงแกลบขนาด 0.84-1.00 มม. โดยให้มีอัตราการไหลเท่ากับ 0.3 kg/min...

3.3 บันทึกผลของอุณหภูมิการเผาไหม้ 15 นาที เก็บเถ้าจากปล่องไอเสีย...

3.4 ทำการปรับอัตราการไหลของอากาศให้ค่าEquivalence ratio 1.0 ....

3.5 ทำทดลองซ้ำตามข้อที่ 3.1) ถึง 3.4) โดยเปลี่ยนขนาดของเชื้อเพลิงแกลบ...

4. ผลการทดลองและวิจารณ์

จากการทดลองการวัดการกระจายอุณหภูมิภายในเตา...

4.1 อิทธิพลของค่า Equivalence Ratios

ลักษณะการกระจายอุณหภูมิภายในเตาเผาแบบวอร์เทค สำหรับ ...

4.2 อิทธิพลของปริมาณอัตราการไหลของอากาศทุติยภูมิต่ออากาศทั้งหมด

อัตราการไหลของอากาศทุติยภูมิต่ออากาศทั้งหมด เป็นค่าที่บ่งชี้ถึง ...

4.3 อิทธิพลของขนาดเชื้อเพลิงแกลบ

ขนาดของเชื้อเพลิงแกลบเป็นปัจจัยหนึ่งที่สำคัญคือ ขนาดของเชื้อเพลิง...

5. สรุปผลการทดลอง

จากการทดลองพบว่าเมื่อปริมาณอากาศที่ใช้เท่ากับปริมาณอากาศทางทฤษฎี ...

กิตติกรรมประกาศ

บทความนี้สำเร็จไปได้ด้วยดีต้องขอขอบคุณ สำนักงาน...

เอกสารอ้างอิง

[1] สุพจน์ น้านำโชค "การเผาไหม้ขี้เลื่อยในห้องเผาไหม้แบบไซโคลน...</paperall>

<top5>เตาวอร์เทค,Vortex combustor,ขนาดเชื้อเพลิงแกลบ,rice husk fuel sizes,การเผาไหม้</top5>

<top10>เตาวอร์เทค,Vortex combustor,ขนาดเชื้อเพลิงแกลบ...</top10>

<top20>เตาวอร์เทค,Vortex combustor,ขนาดเชื้อเพลิงแกลบ...</top20>

</paper>

ภาคผนวก ข

การใช้งาน XML Extender

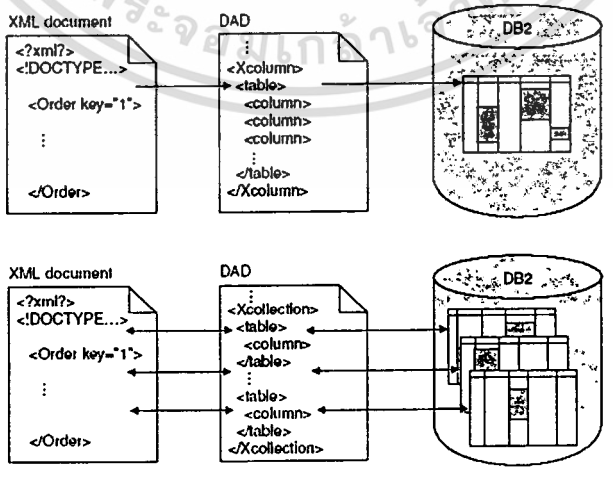
งานวิจัยนี้มีจุดมุ่งหมายเพื่อศึกษาการจัดเก็บข้อมูลที่อยู่ในรูปแบบเอกสาร XML ในโปรแกรมฐานข้อมูล IBM DB2 [5] ร่วมกับ XML Extender [6,14,15] และการจัดทำดัชนีเพื่อใช้ในการค้นคืนเอกสาร XML ด้วยการใช้ค่าน้ำหนักแท็กที่เหมาะสมช่วยทำให้การจัดลำดับความสำคัญของคำดัชนีที่ได้จากระบบมีความใกล้เคียงกับที่จัดโดยผู้เชี่ยวชาญ ซึ่งมีขั้นตอนดังนี้

ข.1 การจัดเก็บเอกสาร XML ในฐานข้อมูล

งานวิจัยนี้ใช้โปแกรมฐานข้อมูลของ IBM DB2 ซึ่งเป็นโปรแกรมฐานข้อมูลทางธุรกิจโปรแกรมหนึ่งที่เป็นที่นิยมในองค์กรต่าง ๆ และใช้ร่วมกับ XML Extender ซึ่งทำให้ DB2 สามารถเก็บและเข้าถึงข้อมูล XML ได้ง่ายขึ้น

ข.2 XML Extender

XML Extender ได้จัดเตรียมความสามารถในการเก็บและเข้าถึงเอกสาร XML และประกอบเอกสาร XML จากตารางที่เกี่ยวข้องหรือแยกส่วนเอกสาร XML เข้าไปเก็บในตารางที่เกี่ยวข้อง ความสามารถนี้ทำได้เนื่องจากมีการเตรียมเซตของข้อมูลชนิดใหม่, ฟังก์ชัน, และ stored procedure เพื่อใช้งานกับเอกสาร XML รูปที่ ข.1 แสดงการนำ XML Extender ไปใช้งานในฐานข้อมูล



รูปที่ ข.1 แสดงการนำ XML Extender ไปใช้งานในฐานข้อมูล

ที่มา : [15]

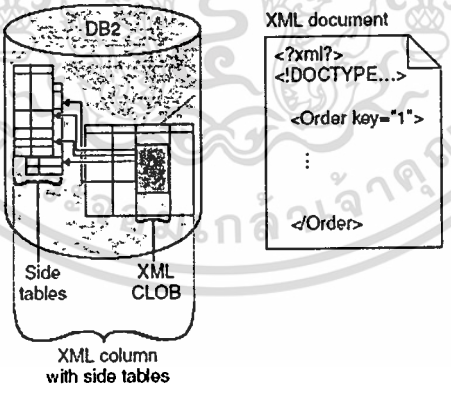
รูปที่ ข.1 แสดงการนำ XML Extender ไปใช้จัดเก็บในฐานข้อมูลโดย XML Extender เป็นตัวจัดการกับเอกสาร XML ที่ต้องการจัดเก็บ ซึ่งการจัดเก็บเอกสาร XML นี้สามารถจัดเก็บได้ 2 แบบคือแบบ XML Column โดยก่อนจัดเก็บต้องสร้างไฟล์ DAD แบบ Xcolumn แล้วจึงจัดเก็บลงฐานข้อมูลและแบบ XML Collection โดยก่อนจัดเก็บต้องสร้างไฟล์ DAD แบบ Xcollection แล้วจึงจัดเก็บข้อมูลลงฐานข้อมูล ซึ่งการจัดเก็บทั้ง 2 แบบนี้จะได้โครงสร้างข้อมูลในฐานข้อมูลที่ต่างกันแล้วแต่การนำไปใช้งาน

ข.2.1 วิธีการจัดเก็บและการเข้าถึง

XML Extender สามารถเก็บเอกสาร XML ทั้งหมดในฐานข้อมูลหรือเก็บเฉพาะเนื้อหาของเอกสาร XML ไว้ในคอลัมน์ของตารางในฐานข้อมูล นอกจากนี้ยังสามารถเก็บเอกสาร XML เป็นไฟล์อยู่ในระบบปฏิบัติการได้

XML Extender มีวิธีจัดการกับเอกสาร XML 2 แบบคือ

- XML Columns : วิธีการนี้อนุญาตให้เก็บเอกสาร XML ในฐานข้อมูล เอกสาร XML ที่ถูกเก็บเข้าไปในฐานข้อมูลจะถูกเก็บไว้ภายในคอลัมน์ที่กำหนดไว้ให้เก็บเอกสาร XML และสามารถแก้ไข ค้นคืน รวมทั้งค้นหาได้ Element และ Attribute ในเอกสารสามารถสร้างเป็นตารางในฐานข้อมูลได้เรียกว่าตาราง Side table ซึ่งสามารถสร้างดัชนีเพื่อให้การค้นหาทำได้รวดเร็ว ในรูปที่ ข.2 แสดงการเก็บเอกสาร XML แบบ XML Columns ร่วมกับ side table



รูปที่ ข.2 แสดงการเก็บเอกสาร XML แบบ XML Columns ร่วมกับ side table

ที่มา : [15]

รูปที่ ข.2 แสดงการเก็บเอกสาร XML แบบ XML Columns ซึ่งมีการนำ Side table มาช่วย โดยเอกสาร XML ที่จัดเก็บแบบนี้ เอกสาร XML จะถูกมองว่าเป็นข้อมูลอันหนึ่งเพื่อให้ทำให้การ

ค้นคืนเอกสารสะดวกขึ้นจึงสร้าง Side table ซึ่งเป็นตารางย่อย ๆ ภายในเอกสาร XML อีกทีหนึ่ง เพื่อสะดวกในการค้นคืน

- XML Collections : วิธีนี้อนุญาตให้ทำการเก็บโครงสร้างของเอกสาร XML ในรูปของตารางในฐานข้อมูล ซึ่งทำให้สามารถประกอบเอกสาร XML จากข้อมูลในฐานข้อมูลได้ หรือแยกเอกสาร XML เข้าไปเก็บในฐานข้อมูล

ข.2.2 Document Access Definition

XML Extender จัดเตรียมรูปแบบการทำแผนที่โครงสร้างเอกสารเรียกว่า Document Access Definition (DAD) ซึ่งเป็นไฟล์ที่ใช้สร้างแผนที่จากเอกสาร XML ไปยังข้อมูลที่เกี่ยวข้องในฐานข้อมูล DAD เป็นเอกสารที่อยู่ในรูปแบบของ XML โดยใช้งานร่วมกับโครงสร้างเอกสาร XML (DTD) กับฐานข้อมูล ในการใช้งานกับทั้ง XML Columns หรือ XML Collection

ข.2.3 การเลือกใช้วิธีจัดการกับเอกสาร XML

XML Columns นำไปใช้ในสถานการณ์ต่อไปนี้

- มีเอกสาร XML อยู่แล้วหรือได้เอกสารมาจากที่อื่น ๆ และต้องการเก็บเอกสาร XML นั้นไว้ในรูปแบบเหมือนของเดิม
- เอกสาร XML ที่เก็บไว้ใช้ในการอ่านไม่ใช่ใช้ในการปรับปรุง
- ต้องการเก็บเฉพาะชื่อไฟล์เอกสาร XML ในฐานข้อมูลแล้วเก็บไฟล์เอกสาร XML ไว้ข้างนอกฐานข้อมูลและต้องการใช้การจัดการของฐานข้อมูล
- การใช้งานเอกสาร XML เป็นการค้นหาส่วนมาก และรู้ว่า Element หรือ Attribute ใดที่ใช้ในการค้นหาบ่อย

XML Collections นำไปใช้ในสถานการณ์ต่อไปนี้

- มีข้อมูลอยู่ในตารางในฐานข้อมูลอยู่แล้ว และต้องการประกอบข้อมูลนั้นให้เป็นเอกสาร XML ที่มี DTD แน่นอน
- ต้องการสร้างมุมมองที่แตกต่างให้กับข้อมูลที่อยู่ในตารางด้วยการสร้างการประกอบเอกสารที่ต่างกัน
- มีเอกสาร XML ที่มาจากแหล่งข้อมูลที่แตกต่างกัน และสนใจเฉพาะข้อมูลไม่ได้สนในแท็กและต้องการเก็บเฉพาะข้อมูลนั้นในฐานข้อมูล
- บางส่วนในเอกสาร XML มีความจำเป็นต้องปรับปรุงแก้ไขบ่อย และการปรับปรุงแก้ไขเป็นสิ่งสำคัญ
- ต้องเก็บเอกสาร XML ทั้งหมดที่เข้ามา แต่โดยส่วนใหญ่จะต้องการค้นคืนเพียงบางส่วนในเอกสารนั้น

งานวิจัยนี้ใช้วิธีการจัดการกับเอกสาร XML ในฐานข้อมูลแบบ XML Columns ไฟล์ DAD ที่ใช้ในงานวิจัยนี้แสดงไว้ในภาคผนวก ค

ภาคผนวก ค
แสดงไฟล์ DAD ที่ใช้ในการวิจัย

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- For Database TT -->
<!-- delete  doctype before using in database -->
<!--DOCTYPE DAD SYSTEM "C:\dxx\dtd\dad.dtd" -->
<DAD>
<dtdid>D:\Study\MasterProject\XMLThesis\Paper.dtd</dtdid>
<validation>NO</validation>
<Xcolumn>
  <table name="TitleSideTable">
        <column name="Title" type="varchar(3980)" path="/paper/title"
        multi_occurrence="YES"/>
    </table>
    <table name="CreatorSideTable">
        <column name="Creator" type="varchar(3980)" path="/paper/creator"
        multi_occurrence="YES"/>
    </table>
    <table name="AddressSideTable">
        <column name="Address" type="varchar(3980)" path="/paper/address"
        multi_occurrence="YES"/>
    </table>
    <table name="AbstractSideTable">
        <column name="Abstract" type="clob(64K)" path="/paper/abstract"
        multi_occurrence="YES"/>
    </table>
    <table name="Keyword">
        <column name="Keyword" type="varchar(3980)" path="/paper/keyword"
        multi_occurrence="YES"/>
    </table>
```

```
<table name="IntroSidetable">
        <column name="Intro" type="clob(64K)" path="/paper/intro"
        multi_occurrence="YES"/>
</table>
<table name="ChapterNameSidetable">
        <column name="ChapterName" type="varchar(3000)"
        path="/paper/chapter/name" multi_occurrence="YES"/>
</table>
<table name="ChapterContentSideTable">
        <column name="ChapterContent" type="clob(64K)"
        path="/paper/chapter/content"  multi_occurrence="YES"/>
</table>
<table name="SubChapterNameSideTable">
        <column name="SubchapterName" type="varchar(3000)"
        path="/paper/chapter/subchapter/name" multi_occurrence="YES"/>
</table>
<table name="SubChapterContentSideTable">
        <column name="SubchapterContent" type="clob(64K)"
        path="/paper/chapter/subchapter/content" multi_occurrence="YES"/>
</table>
<table name="ExperimentContentSideTable">
        <column name="ExperimentContent" type="clob(64K)"
        path="/paper/experiment/content" multi_occurrence="YES"/>
</table>
<table name="SubExperimentNameSideTable">
        <column name="SubExperimentName" type="varchar(3000)"
        path="/paper/experiment/subchapter/name" multi_occurrence="YES"/>
</table>
<table name="SubExperimentContentSideTable">
        <column name="SubExperimentContent" type="clob(64K)"
        path="/paper/experiment/subchapter/content" multi_occurrence="YES"/>
</table>
```

```
<table name="ResultContentSideTable">
        <column name="ResultContent" type="clob(64K)" path="/paper/result/content"
        multi_occurrence="YES"/>
</table>
<table name="SubResultNameSideTable">
        <column name="SubResultName" type="varchar(3000)"
        path="/paper/result/subchapter/name" multi_occurrence="YES"/>
</table>
<table name="SubResultContentSideTable">
        <column name="SubResultContent" type="clob(64K)"
        path="/paper/result/subchapter/content" multi_occurrence="YES"/>
</table>
<table name="SummaryContentSideTable">
        <column name="SummaryContent" type="clob(64K)"
        path="/paper/summary/content" multi_occurrence="YES"/>
</table>
<table name="SubSummaryNameSideTable">
        <column name="SubSummaryName" type="varchar(3000)"
        path="/paper/summary/subchapter/name" multi_occurrence="YES"/>
</table>
<table name="SubSummaryContentSideTable">
        <column name="SubSummaryContent" type="clob(64K)"
        path="/paper/summary/subchapter/content" multi_occurrence="YES"/>
</table>
<table name="AcknowledgementsSideTable">
        <column name="Acknowledgements" type="clob(64K)"
        path="/paper/acknowledgements" multi_occurrence="YES"/>
</table>
<table name="BiographySideTable">
        <column name="Biography" type="clob(64K)" path="/paper/biography"
        multi_occurrence="YES"/>
</table>
```

```
        <table name="HistorycreatorSideTable">
                <column name="Historycreator" type="clob(64K)" path="/paper/historycreator"
                multi_occurrence="YES"/>
        </table>
        <table name="PaperallSideTable">
                <column name="Paperall" type="clob(256K)" path="/paper/paperall"
                multi_occurrence="YES"/>
        </table>
        <table name="Top5SideTable">
                <column name="Top5" type="varchar(3980)" path="/paper/top5"
                multi_occurrence="YES"/>
        </table>
        <table name="Top10SideTable">
                <column name="Top10" type="varchar(3980)" path="/paper/top10"
                multi_occurrence="YES"/>
        </table>
        <table name="Top20SideTable">
                <column name="Top20" type="varchar(3980)" path="/paper/top20"
                multi_occurrence="YES"/>
        </table>
</Xcolumn>
</DAD>
```

ภาคผนวก ง
ค่าเฉลี่ยจำนวนคำของเอกสารในระบบ

ค่าเฉลี่ยจำนวนคำของเอกสารในระบบที่ใช้ทดลองแสดงในตารางที่ 1

ตารางที่ ง.1 แสดงค่าฉลี่ยจำนวนคำของเอกสารในระบบ

ข้อมูลที่สนใจ	จำนวนคำเฉลี่ย
ในบทความ	1355.58
ในแต่ละแท็ก	178.23
ในแท็ก Title	14.4
ในแท็ก Abstract	135.03
ในแท็ก Keyword	5.53
ในแท็ก Introduction	196.68
ในแท็ก Chapter Name	7.96
ในแท็ก Chapter Content	388.19
ในแท็ก Sub Chapter Name	15.35
ในแท็ก Sub Chapter Content	533.77
ในแท็ก Experiment Content	158.13
ในแท็ก Sub Experiment Name	14.1
ในแท็ก Sub Experiment Content	365.23
ในแท็ก Result Content	238.21
ในแท็ก Sub Result Name	17.13
ในแท็ก Sub Result Content	404.53
ในแท็ก Summary Content	95.07
ในแท็ก Sub Summary Name	18.5
ในแท็ก Sub Summary Content	422

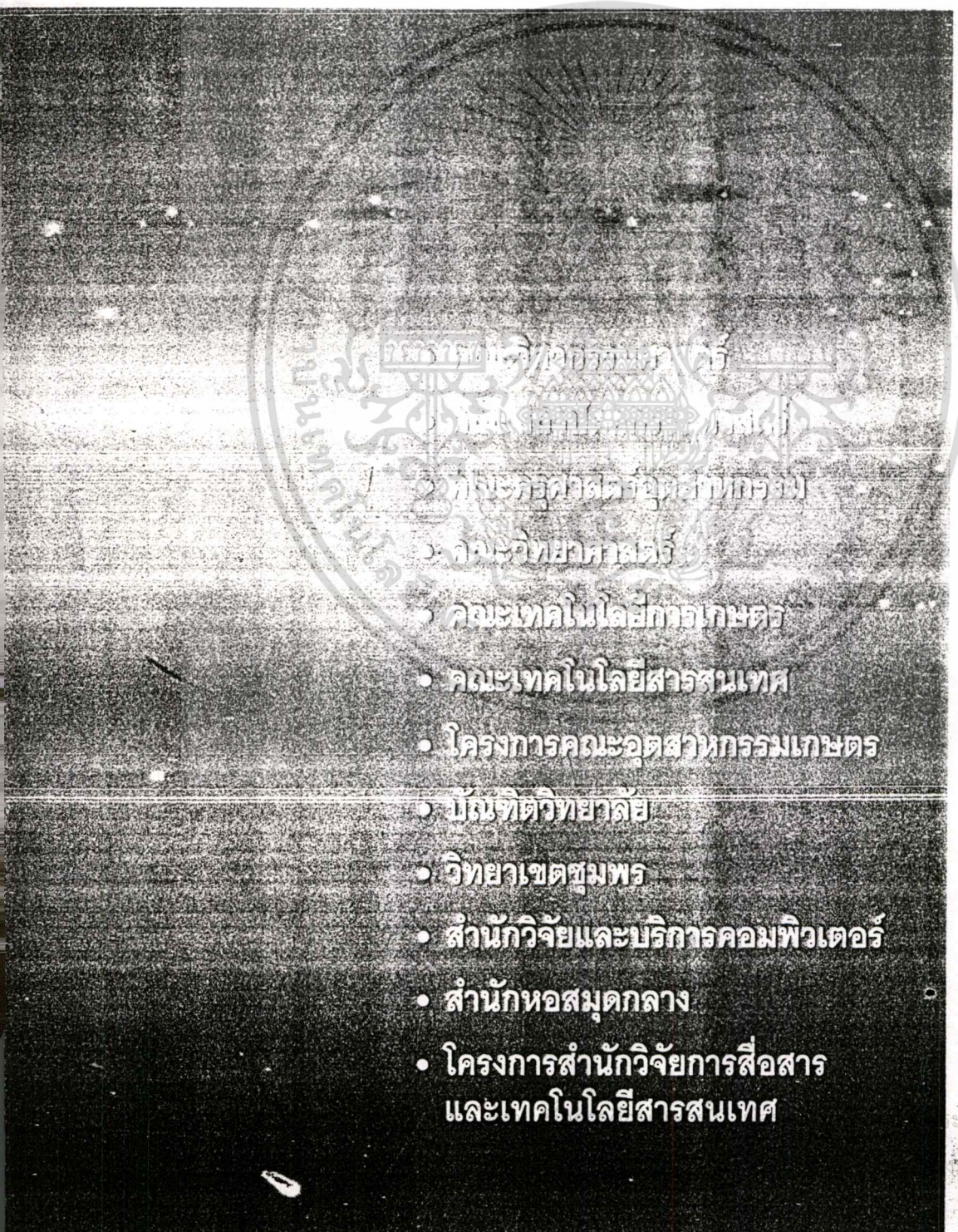
งานวิจัยที่ได้รับการตีพิมพ์

1. เกียรติณรงค์ ทองประเสริฐ และ วิศิษฏ์ หิรัญกิตติ. 2547. "การจัดเรียงลำดับคำดัชนีใน เอกสารเอ็กซ์เอ็มแอลโดยอาศัยค่าน้ำหนักของแท็ก." วารสารพระจอมเกล้า. 12(1) : 9-18.

109

ISSN

วารสารพระจอมเกล้าลาดกระบัง



การจัดเรียงลำดับคำดัชนีในเอกสารเอ็กซ์เอ็มแอลโดยอาศัยค่าน้ำหนักของแท็ก

Ranking Index Terms in XML Documents Using Tag Weights

เกียรติณรงค์ ทองประเสริฐ
นักศึกษาปริญญาโท

วิศิษฏ์ หิรัญกิตติ
อาจารย์

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทคัดย่อ

บทความเรื่องนี้เป็นการวิจัยเกี่ยวกับการค้นคืนข้อมูลข่าวสารจากเอกสาร XML ใด ๆ ด้วยวิธีเวคเตอร์โมเดล ซึ่งวิธีนี้จะจัดเรียงลำดับคำดัชนีตามความถี่ที่ปรากฏในเอกสาร ในงานวิจัยนี้เราได้ปรับปรุงวิธีการจัดเรียงลำดับคำดัชนีในเอกสาร XML แล้วนำไปทดสอบกับตัวอย่างเอกสาร XML ที่เป็นบทคัดย่อวิทยานิพนธ์และบทความวิจัย วิธีการใหม่ที่นำเสนอนี้ไม่เพียงนำเอาความถี่ของคำที่ปรากฏในเอกสารมาพิจารณาตามแบบวิธีเวคเตอร์โมเดลเท่านั้น แต่ได้เพิ่มค่าน้ำหนักที่แตกต่างกันให้กับคำที่ปรากฏในแท็ก XML ที่ต่างกัน โดยแท็กที่มีความสำคัญมากกว่าจะกำหนดให้มีค่าน้ำหนักมากกว่า ซึ่งผลจากการปรับปรุงการคำนวณค่าน้ำหนักของคำเพื่อใช้จัดลำดับโดยนำค่าน้ำหนักแท็กที่คำนั้นปรากฏมาพิจารณาร่วมด้วย ทำให้คำที่ปรากฏในแท็กที่มีความสำคัญสูงกว่ามีแนวโน้มได้ค่าน้ำหนักมากกว่าคำที่ปรากฏในแท็กที่มีความสำคัญน้อยกว่า ส่งผลให้คำดัชนีที่เลือกจากคำที่ปรากฏในแท็กที่มีความสำคัญสูงกว่า มีโอกาสถูกจัดลำดับให้สูงกว่าคำดัชนีที่เลือกจากคำที่ปรากฏในแท็กที่มีความสำคัญน้อยกว่า

คำสำคัญ : การค้นคืนข้อมูลข่าวสารจากเอกสาร XML การเรียงจัดลำดับคำดัชนี ห้องสมุดดิจิตอล

Abstract

This research is concerned with information retrieval from XML documents using the vector model. This model ranks indexes from documents according to the frequency of their appearance in the documents. In this paper we adapt this method to rank indexes in XML documents and test our approach with XML documents in the forms of thesis abstracts and the research papers. In our approach, we not only rank indexes according to the frequency of word appearance, but we also add different weights to words appearing in different tags of the documents. The more important the tag, the more weight is assigned to it. The new approach for ranking indexes, proposed in the paper, has shown that the words appearing in a more important tag are likely to be ranked as higher than the words appearing in a less important tag.

Keywords : Information retrieval from XML documents, Ranking index terms, Digital library

1. บทนำ

ปัจจุบันเอกสาร XML มีการใช้งานในด้านต่าง ๆ แพร่หลายมากขึ้น โดยเฉพาะใช้แทนเอกสารอิเล็กทรอนิกส์ เช่น e-book, e-journal เป็นต้น งานวิจัยนี้ได้นำเสนอวิธีการจัดเรียงคำดัชนีในเอกสาร XML โดยได้ทดลองกับเอกสาร XML ที่เป็นบทคัดย่อวิทยานิพนธ์ของนักศึกษาระดับบัณฑิตศึกษาของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และเอกสาร XML ที่เป็นบทความวิจัย โดยโครงสร้างของเอกสาร XML เหล่านี้ มีการกำหนดอ้างอิงตามรูปแบบ DCMI Metadata Terms [7] การค้นคืนเอกสารใช้วิธีเวคเตอร์โมเดล [6] ซึ่งจะมีการจัดลำดับความสำคัญของคำดัชนี โดยมีการคิดคำนวณค่าน้ำหนักหลาย ๆ แบบ [3, 4] งานวิจัยนี้มีการปรับปรุงการให้น้ำหนักกับคำ แตกต่างจากงานวิจัยเดิม [3, 4] นำมาซึ่งผลการจัดเรียงลำดับคำดัชนีที่ดีขึ้น

เนื้อหาในบทความนี้เริ่มจากการกล่าวถึงงานวิจัยที่เกี่ยวข้องแล้วในหัวข้อที่ 3 จะกล่าวถึงโครงสร้างของเอกสาร XML ที่นำมาใช้ทดลองกับวิธีการที่นำเสนอ จากนั้นจะกล่าวถึงการประมวลผลเอกสารดังกล่าวในหัวข้อที่ 4 ส่วนการสร้างคำดัชนีจากเอกสาร XML การวิเคราะห์วิธีการคำนวณค่าน้ำหนักแต่ละแบบ และผลการทดลองที่เป็นการเปรียบเทียบการเรียงลำดับคำดัชนี โดยวิธีที่นำเสนอและวิธีการอื่น ๆ นั้น จะกล่าวถึงในหัวข้อที่ 5 และการค้นคืนเอกสารในหัวข้อที่ 6 ตามลำดับ สุดท้ายเป็นการวิเคราะห์ผลการทดลองและสรุปผล

2. งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับการพัฒนาระบบสืบค้นเอกสารภาษาไทยได้แก่ [2] ซึ่งนำเสนอวิธีการทำดัชนีหลายระดับ ส่วนงานวิจัยการค้นคืนเอกสารภาษาไทยที่อยู่ในรูปแบบ XML นั้นได้แก่ [1] ซึ่งนำเสนอวิธีการค้นคืนข้อมูลภายในเอกสาร XML โดยผ่านภาษาค้นคืนที่พัฒนาขึ้นใหม่ให้สามารถจัดลำดับความเหมือนของเอกสารได้ ส่วนงานวิจัยที่เกี่ยวข้องกับระบบค้นคืนเอกสาร XML ที่ใช้เทคนิคการค้นคืนแบบ Information retrieval มีการนำเสนอใน [3] โดยการหาค่าน้ำหนักคำดัชนีได้มีการนำเอาค่าน้ำหนักของแท็กมาคำนวณร่วมด้วย

3. โครงสร้างเอกสาร XML ที่ใช้ในการทดลอง

ถึงแม้ว่าวิธีการจัดเรียงคำดัชนีในเอกสาร XML ที่จะนำเสนอในบทความนี้จะสามารถใช้กับเอกสาร XML ใด ๆ แต่เพื่อเป็นการแสดงให้เห็นถึงการประยุกต์ใช้งานวิธีดังกล่าว เราจึงได้เลือกบางตัวอย่างของเอกสาร XML เพื่อนำมาใช้ในการทดลอง โดยเราได้เลือกเอกสาร XML ที่เป็นบทคัดย่อวิทยานิพนธ์ และเอกสาร XML ที่เป็นบทความวิจัย ซึ่งแต่ละแบบมีโครงสร้างเอกสารอ้างอิงตามรูปแบบ DCMI Metadata Terms [7] โดยโครงสร้างเอกสารบทคัดย่อวิทยานิพนธ์ของทางสถาบันฯ ได้มีการออกแบบแท็กที่ใช้เก็บข้อมูลในส่วนต่าง ๆ ดังนี้

แท็ก abstract ใช้แสดงบทคัดย่อ subject ใช้เก็บชื่อวิทยานิพนธ์ และใช้แอตทริบิวต์ lang แสดงชนิดภาษา ซึ่งมีได้ 2 ค่า คือ en แทนภาษาอังกฤษ และ th แทนภาษาไทย

ส่วนแท็ก creator เป็นส่วนที่ใช้เก็บชื่อผู้แต่ง ประกอบด้วย initial คือคำนำหน้าชื่อ firstname คือชื่อต้น middlename คือชื่อกลาง (ถ้ามี) lastname คือนามสกุล studentid นั้นแสดงรหัสนักศึกษา degree แสดงชื่อปริญญา programme แสดงสาขาที่สำเร็จการศึกษา faculty แสดงคณะที่สำเร็จการศึกษา year แสดงปีที่สำเร็จการศึกษา advisor แสดงชื่ออาจารย์ผู้ควบคุมวิทยานิพนธ์ ประกอบด้วย initial, firstname, middlename และ lastname เช่นเดียวกัน

แท็ก abstractcontent ใช้แสดงส่วนเนื้อหาบทคัดย่อ keywords แสดงชุดของคำสำคัญประกอบด้วย keyword แต่ละตัวซึ่งใช้แสดงคำสำคัญแต่ละคำของวิทยานิพนธ์

```
<thesis>
  <abstract>
    <subject leng="th"> อิทธิพลของขนาดของแกลบต่อคุณลักษณะการเผาไหม้</subject>
    <subject leng="en">Effect of Rice Husk Sizes on Combustion Characteristics</subject>
    <creator leng="th">
      <initial>นาย</initial>
      <firstname>สมศักดิ์</firstname><middlename/>
      <lastname>โพธิ์ถวิลเกียรติ์</lastname>
    </creator>
    <creator leng="en">
      <initial>Mr.</initial>
      <firstname>Somsak</firstname><middlename/>
      <lastname>Potawinkiat</lastname>
    </creator>
    <studentid>41061039</studentid>
    <degree leng="th">วิศวกรรมศาสตรมหาบัณฑิต</degree>
    <degree leng="en">Master of Engineering</degree>
    <advisor leng="th">
      <initial>รศ.</initial>
      <firstname>พงษ์เจต</firstname><middlename/>
      <lastname>พรหมวงศ์</lastname>
    </advisor>
    <advisor leng="en">
      <initial>Assoc.Prof.</initial>
      <firstname>Pongjeat</firstname><middlename/>
      <lastname>Phromwong</lastname>
    </advisor>
    <abstractcontent leng="th">บทความนี้นำเสนอการศึกษาทดลองถึงพฤติกรรมการเผาไหม้ของขนาดเชื้อ...</abstractcontent>
    <abstractcontent leng="en">The paper presents the experimental study of rice husk...</abstractcontent>
    <keywords>
      <keyword>เตาอร์เทค</keyword>
      <keyword>rice husk fuel sizes</keyword>
    </keywords>
  </abstract>
</thesis>
```

รูปที่ 1 แสดงตัวอย่างเอกสาร XML บทคัดย่อวิทยานิพนธ์

สำหรับโครงสร้างเอกสาร XML ที่เป็นบทความวิจัยได้รวมเอาส่วนของบทคัดย่อไว้ด้วยและเพิ่มเติมแท็กต่าง ๆ ดังต่อไปนี้

แท็ก intro ใช้แสดงส่วนบทนำของบทความ ส่วนหัวข้ออื่น ๆ แสดงในส่วนของ section โดยมีแอตทริบิวต์ number ใช้แสดงเลขหัวข้อ แท็ก name แสดงชื่อหัวข้อ และ content แสดงส่วนของเนื้อหา ส่วนเนื้อหาย่อยภายใน section แสดงด้วยแท็ก subsection ซึ่งมีแท็กอื่น ๆ เหมือนของแท็ก section สำหรับ แท็ก conclusion แสดงส่วนที่เป็นบทสรุป และในแท็ก references แสดงส่วนเอกสารอ้างอิง

```
<paper>
  <abstract>
    <subject leng="th">อิทธิพลของขนาดของแกลบต่อคุณลักษณะ
การเผาไหม้</subject>
    <subject leng="en">Effect of Rice Husk Sizes on
Combustion Characteristics</subject>
    <creator leng="th">
      <initial>นาย</initial>
      <firstname>สมศักดิ์</firstname><middlename/>
      <lastname>โพธิ์ถวิลเกียรติ์</lastname>
    </creator>
    <creator leng="th">
      <initial>รศ.</initial>
      <firstname>พงษ์เจต</firstname><middlename/>
      <lastname>พรหมวงศ์</lastname>
    </creator>
    <abstractcontent leng="th">บทความนี้นำเสนอการศึกษา
ทดลองถึงพฤติกรรมการเผาไหม้ของขนาดเชื้อ...</abstractcontent>
    <abstractcontent leng="en">The paper presents the
experimental study of rice husk...</abstractcontent>
    <keywords>
      <keyword>เตาอร์เทค</keyword>
      <keyword>rice husk fuel sizes </keyword>
    <keywords>
  </abstract>
  <intro>ปัจจุบันความต้องการใช้พลังงานมีอัตราส่วนที่เพิ่มขึ้น
โดยเฉพาะในภาคโรงงานอุตสาหกรรม พลังงานส่วนใหญ่นั้น...</intro>
  <section number="2">
    <name>เครื่องมือและอุปกรณ์การทดลอง</name>
    <content>การทดลองการเผาไหม้ในเตาเผาแบบวอร์เทค
ใช้เชื้อเพลิงแกลบในการทดสอบอยู่สองขนาดคือ...</content>
  </section>
                    ...
  <section number="4">
    <name>ผลการทดลอง</name>
    <content>จากการทดลองการวัดการกระจายอุณหภูมิ
ภายในเตาทั้งสิ้น...</content>
    <subsection number "4.1">
      <name>อิทธิพลของค่า Equivalence
Ratios...</name>
      <content>ลักษณะการกระจายอุณหภูมิภาย
...</content>
    </subsection>
  </section>
                    ...
  <conclusion>
    <content>จากการทดลองพบว่าเมื่อปริมาณอากาศที่ใช้
เท่ากับปริมาณอากาศทางทฤษฎี...</content>
  </conclusion>
  <references>
    <reference>[1] สุพจน์ น้ำน่าโชค "การเผาไหม้ชีเลื่อยใน
ห้องเผาไหม้แบบ</reference>
                    ...
  </references>
  <acknowledgements>บทความนี้สำเร็จไปได้ด้วยดีต้อง
```

รูปที่ 2 แสดงตัวอย่างเอกสาร XML บทความวิจัย

แท็ก acknowledgements แสดงส่วนกิตติกรรมประกาศ และส่วนประวัติผู้เขียนแสดงในแท็ก creatorprofile ซึ่งข้อมูลของทั้ง 2 แท็กจะมีหรือไม่ก็ได้ ตัวอย่างเอกสาร XML ที่ใช้แทนบทความวิจัยแสดงได้ดังรูปที่ 2

4. การประมวลผลเอกสาร

เมื่อได้เอกสาร XML แล้วจะนำมาเข้ากระบวนการประมวลผลเอกสาร โดยเอกสาร XML ที่นำมาใช้ประกอบด้วยภาษาไทยและภาษาอังกฤษ ซึ่งขั้นตอนนี้ประกอบด้วย

4.1 การคัดเลือกแท็ก ทำการคัดเลือกแท็กในเอกสารที่จะนำข้อมูลภายในแท็กนั้นมาคิดหาคำดัชนี โดยคัดเลือกแท็กที่มีความสำคัญสูงของเอกสาร เช่น ชื่อวิทยานิพนธ์และคำสำคัญ ส่วนแท็กที่ไม่เกี่ยวข้องจะทำการตัดทิ้งเนื่องจากเราไม่ให้ความสำคัญและเพื่อลดจำนวนข้อมูลที่จะนำมาประมวลผล (ในทางทฤษฎีอาจถือว่าข้อมูลในแท็กเหล่านี้สามารถตัดทิ้งไปได้เนื่องจากค่าน้ำหนักของแท็กดังกล่าวมีค่าเป็น 0 จึงทำให้ไม่ต้องพิจารณาว่าข้อมูลเหล่านี้เป็นคำดัชนี)

4.2 การตัดคำในแต่ละแท็ก ขั้นตอนนี้จะตัดคำภาษาไทยและภาษาอังกฤษออกเป็นคำหรือกลุ่มคำ ในส่วนของภาษาไทยจะตัดประโยคที่ยาวให้เป็นคำหรือกลุ่มคำที่มีความหมาย หลังจากเสร็จสิ้นกระบวนการนี้จะได้คำที่แยกออกจากกัน

4.3 ตัดคำที่ไม่สำคัญอื่น ๆ คำที่ไม่สำคัญในที่นี้คือคำที่พบบ่อยในเอกสาร ซึ่งคำเหล่านี้ควรตัดทิ้ง เนื่องจากไม่มีประโยชน์ในการใช้จำแนกเอกสาร ตัวอย่างเช่น คำนำหน้าชื่อ คำบุพบท คำสันธาน ในการตัดคำเหล่านี้ออกทำให้โครงสร้างของคำดัชนีในระบบมีขนาดเล็กลง

5. การคำนวณค่าน้ำหนัก

หลังจากการประมวลผลเอกสารแล้ว จะได้คำหรือกลุ่มคำที่แยกกัน จากนั้นจะนำคำหรือกลุ่มคำเหล่านี้มานับความถี่ในแต่ละเอกสารเพื่อนำไปคำนวณค่าน้ำหนักต่อไป

5.1 การคำนวณหาค่าน้ำหนัก

การคำนวณหาค่าน้ำหนัก เราใช้วิธีเวกเตอร์โมเดล [6] ในการหาค่าน้ำหนักของคำดัชนีเพื่อนำไปคำนวณหาค่าความเหมือน ในส่วนการค้นคืนเอกสารนั้นเราจะกล่าวถึงภายหลังในหัวข้อที่ 6 การคำนวณหาค่าน้ำหนักของคำดัชนีนั้นหาได้จากสมการที่จะกล่าวถึงต่อไปนี้

วิธีที่จะกล่าวถึง **วิธีที่ 1** เป็นการคำนวณตามแบบเวกเตอร์โมเดล

การหาค่าความถี่มาตรฐาน $f_{i,j}$ ของเทอม k_i ในเอกสาร d_j หาได้จากสมการ

$$f_{i,j} = \frac{freq_{i,j}}{\max_l freq_{i,j}} \quad (1)$$

โดยที่ $freq_{i,j}$ คือ ค่าความถี่ของแต่ละเทอม k_i ที่พบในเอกสาร d_j, $max_l\ freq_{i,j}$ คือ ค่าความถี่ของเทอม l ที่มีค่ามากที่สุดในแต่ละเอกสาร

ในการหาค่าส่วนกลับความถี่เอกสารของเทอม k_i หรือค่า idf_i หาได้จากสมการ

$$idf_i = \log \frac{N}{n_i} \quad (2)$$

โดยที่ N คือ จำนวนเอกสารทั้งหมดในระบบ, n_i คือ จำนวนเอกสารที่มีเทอม k_i ปรากฏอยู่

ส่วนค่าน้ำหนักคำดัชนีตามวิธีเวกเตอร์โมเดลหาได้จาก

$$w_{i,j} = f_{i,j} \times idf_i \quad (3)$$

5.2 การปรับปรุงการคำนวณค่าน้ำหนักของวิธีเวกเตอร์โมเดล

การคำนวณค่าน้ำหนักคำด้วยวิธีที่ 1 ตามแบบเวกเตอร์โมเดลกับเอกสาร XML เราพบว่านอกจากความถี่ของคำแล้ว การปรากฏของคำในแต่ละแท็กของเอกสารนั้นควรจะมีความสำคัญแตกต่างกันด้วย เช่น คำที่อยู่ในแท็กคำสำคัญ ควรมีความสำคัญสูงสุด รองลงมาควรจะเป็นแท็กชื่อวิทยานิพนธ์ ดังนั้นเราจึงควรให้ความสำคัญกับแท็กที่มีคำในเอกสารปรากฏอยู่ด้วยการกำหนดค่าน้ำหนักแท็ก w_k ให้กับแต่ละแท็กแล้วนำมาคำนวณร่วมกับการคำนวณค่าน้ำหนักโดยวิธีเวกเตอร์โมเดล ซึ่งการคำนวณวิธีนี้ได้นำเสนอใน [3]

วิธีที่ 2 ซึ่งนำมาจาก [3] มีแนวคิดในการคำนวณ คือ ถ้าพบคำที่แท็กใดให้นำค่าน้ำหนักของแท็กมาคิดเพิ่มให้กับค่า $w_{i,j}$ ที่ได้จากวิธีที่ 1 ดังสมการ

$$w'_{i,j} = \sum w_k \times f_{i,j} \times idf_i \quad (4)$$

จากการทดลองเราพบว่าในกรณีที่คำดัชนีปรากฏหลายครั้งในแท็กหนึ่ง ๆ สมการนี้อาจให้ผลการคำนวณค่าน้ำหนักของคำที่ไม่สอดคล้องกับสิ่งที่ควรจะเป็น นั่นคือ คำที่ปรากฏบ่อยครั้งในแท็กเดียวกันควรจะมีค่าน้ำหนักมากกว่าคำที่ปรากฏน้อยครั้งกว่า แต่จากวิธีที่ 2 ค่าน้ำหนักของคำทั้ง 2 จะไม่แตกต่างกันเลย ดังนั้นจากปัญหาดังกล่าวเราจึงปรับปรุงสมการที่ใช้ในการคำนวณค่าน้ำหนักคำใหม่เป็นวิธีที่ 3

วิธีที่ 3 ซึ่งเป็นวิธีที่เรานำเสนอ มีการหาค่าความถี่มาตรฐาน $f_{i,j,k}$ ของเทอม k_i ในเอกสาร d_j ในแท็ก t_k จากสมการที่ (5) ซึ่งเป็นวิธีการคำนวณแบบใหม่ดังนี้

$$f_{i,j,k} = \frac{\sum_{k=1}^{t} (freq_{i,j} \times w_k)}{\max_l freq_{l,j}} \quad (5)$$

โดยที่ $freq_{i,j}$ คือ ค่าความถี่ของแต่ละเทอม k_i ที่พบในเอกสาร d_j, w_k คือ ค่าน้ำหนักแต่ละแท็ก t_k ในเอกสาร, $max_l\ freq_{i,j}$ คือ ค่าความถี่ของเทอม l ที่มีค่ามากที่สุดในแต่ละเอกสาร ส่วนการหาค่า idf_i ยังคงใช้สมการ (2)

ดังนั้นค่าน้ำหนักคำดัชนีตามวิธีที่ปรับปรุงใหม่คำนวณได้ตามสมการ

$$w''_{i,j} = f_{i,j,k} \times idf_i \quad (6)$$

```
<d1>
<tag1>. word...word...word...</tag1>
<tag2> .word...word...word...</tag2>
<tag3>. word...word...word...</tag3>
</d1>

<d2>
<tag1>  word...</tag1>
<tag2>  word .word...word...word...</tag2>
<tag3>...word...word...word...word...</tag3>
</d2>

<d3>
<tag1> .word...</tag1>
<tag2>...word...word...word...</tag2>
<tag3>...word...word...word...word...word...</tag3>
</d3>
```

รูปที่ 3 แสดงตัวอย่างเอกสาร XML 3 เอกสารที่ทำให้ค่าน้ำหนักคำ "word" ที่ได้จากการคำนวณวิธีที่ 2 มีค่าเท่ากัน

จากตัวอย่างเอกสาร XML ทั้ง 3 ในรูปที่ 3 เราสมมุติให้มีคำว่า "word" ปรากฏซ้ำ ๆ ในเอกสาร จากนั้นได้แสดงให้เห็นถึงความแตกต่างในการคำนวณค่าน้ำหนักคำดัชนีจากทั้ง 3 วิธีดังตารางต่อไปนี้ โดยวิธีแรกให้ $w_{i,j}$ คำนวณจากสมการที่ (3) วิธีที่ 2 ให้ $w'_{i,j}$ คำนวณจากสมการที่ (4) และวิธีที่ 3 ให้ $w''_{i,j}$ คำนวณจากสมการที่ (6)

ตารางที่ 1 แสดงผลการคำนวณค่าน้ำหนักด้วยวิธีที่ต่างกัน

$f_{1,j}$	w_1	$f_{2,j}$	w_2	$f_{3,j}$	w_3	$w_{i,j}$	$w'_{i,j}$	$w''_{i,j}$
1	0.9		0.8	8	0.4	1.61	0.58	0.73
8	0.9		0.8	1	0.4	1.61	0.58	1.36
3	0.9	3	0.8	3	0.4	1.61	0.46	1.13
1	0.9	4	0.8	4	0.4	1.61	0.46	1.02
1	0.9	3	0.8	5	0.4	1.61	0.46	0.95

จากตารางที่ 1 ในแถวที่ 1 และ 2 แสดงให้เห็นว่าคำที่ปรากฏในแท็กเดียวกันแต่มีความถี่ต่างกัน ค่าน้ำหนักที่คำนวณด้วยวิธีที่ 2 ($w'_{i,j}$) จะให้ค่าไม่ต่างกัน แต่ถ้าคำนวณด้วยวิธีที่ 3 จะทำให้ได้ค่าน้ำหนัก $w''_{i,j}$ ที่แตกต่างกัน ซึ่งผลที่ได้ก็จะเหมือนกันกับกรณีแถวที่ 3, 4 และ 5 ส่วนวิธีการคำนวณที่ 1 จะให้ค่าไม่แตกต่างกันเลยในทุกแถว เนื่องจากคำนวณจากเพียงความถี่ของคำที่พบและค่าส่วนกลับความถี่เอกสารเพียงอย่างเดียว (สังเกตว่าทั้ง 3 เอกสารมีคำว่า "word" ปรากฏ 9 ครั้งเช่นเดียวกัน) ดังนั้นการคำนวณค่าน้ำหนักคำดัชนีด้วยวิธีที่ 3 จะให้ผลการคำนวณเพื่อเรียงลำดับความสำคัญของคำดัชนีดีกว่าเมื่อเทียบกับวิธีอื่น ๆ ในกรณีที่เราต้องการให้ความสำคัญกับคำและความถี่ของคำที่ปรากฏในแท็กต่าง ๆ ไม่เท่ากัน

หลังจากคำนวณค่าน้ำหนักของเทอมแล้ว จะนำคำดัชนีและค่าน้ำหนักของคำที่ได้ไปจัดเก็บในฐานข้อมูลเพื่อนำไปใช้ในการค้นคืนเอกสาร ซึ่งจะกล่าวถึงในหัวข้อที่ 6 ต่อไป

5.3 การเปรียบเทียบผลของการคำนวณค่าน้ำหนักคำทั้ง 3 วิธี

ตัวอย่างต่อไปนี้แสดงการเปรียบเทียบวิธีการคำนวณค่าน้ำหนักคำในแต่ละแบบ จากตัวอย่าง 3 เอกสาร โดยกำหนดให้น้ำหนักของแท็ก subject มีค่า 0.8 แท็ก studentname มีค่า 0.6 และ แท็ก abstractcontent มีค่า 0.4 การกำหนดค่าน้ำหนักแท็กนี้เป็นไปตามลำดับความสำคัญของแท็กในเอกสาร ซึ่งแท็ก subject มีความสำคัญมากกว่าแท็ก studentname และแท็ก studentname มีความสำคัญมากกว่าแท็ก abstractcontent ตามลำดับ

ซึ่งเอกสารทั้ง 3 มีรายละเอียดดังนี้

d_1=<subject>การศึกษาเลือกสร้างฟิล์มเพชรเฉพาะพื้นที่ด้วยวิธี CVD แบบความร้อน</subject>
<studentname>นาย กอบศักดิ์ ศรีประภา</studentname>
<abstractcontent>ปัจจุบันมีการนำสารกึ่งตัวนำชนิดใหม่เพื่อใช้งานแทนสารกึ่งตัวนำซิลิกอนและแกลเลียมอาร์เซไนด์ เพชรซึ่งอยู่ในรูปหนึ่งของคาร์บอนได้รับการสนใจอย่างมาก</abstractcontent>

d_2= <subject> การวิเคราะห์ปรากฏการณ์ของความดันและอุณหภูมิของหม้อแปลงจำหน่ายแบบแช่น้ำมันเมื่อเกิดฟอลท์</subject>
<studentname>นาย เจนศักดิ์ เอกบูรณะวัฒน์</studentname>
<abstractcontent>วิทยานิพนธ์นี้เป็นการเสนอวิธีการวิเคราะห์ปรากฏการณ์ของความดันและอุณหภูมิที่เปลี่ยนแปลงไปของหม้อแปลงจำหน่ายแบบแช่น้ำมัน</abstractcontent>

d_3 = <subject>การชดเชยผลของอุณหภูมิในวงจรสายพานกระแสแบบทรานส์ลิเนียร์และวงจรขยายโอทีเอ</subject>
<studentname>นาย เฉลิมภัณฑ์ ฟองสมุทร</studentname>
<abstractcontent>ถ้าเกิดการลัดวงจรที่รุนแรงอาจเกิดการระเบิดได้และฉนวนของขดลวดอาจเกิดการชำรุดเสียหายได้</abstractcontent>

เมื่อนำเอกสารทั้ง 3 มาตัดคำแล้วได้ผลดังนี้

d_1 = <subject>การศึกษาเลือกสร้างฟิล์มเพชรเฉพาะพื้นที่ด้วยวิธีCVDแบบความร้อน</subject>
<studentname>นายกอบศักดิ์ศรีประภา</studentname>
<abstractcontent>ปัจจุบันมีการนำสารกึ่งตัวนำชนิดใหม่เพื่อใช้งานแทนสารกึ่งตัวนำซิลิกอนและแกลเลียมอาร์เซไนด์เพชรซึ่งอยู่ในรูปหนึ่งของคาร์บอนได้รับการสนใจอย่างมาก</abstractcontent>
d_2= <subject>การวิเคราะห์ปรากฏการณ์ของความดันและอุณหภูมิของหม้อแปลงจำหน่ายแบบแช่น้ำมันเมื่อเกิดฟอลท์</subject>
<studentname>นายเจนศักดิ์เอกบูรณะวัฒน์</studentname>
<abstractcontent>วิทยานิพนธ์นี้เป็นการเสนอวิธีการวิเคราะห์ปรากฏการณ์ของความดันและอุณหภูมิที่เปลี่ยนแปลงไปของหม้อแปลงจำหน่ายแบบแช่น้ำมัน</abstractcontent>
d_3 = <subject>การชดเชยผลของอุณหภูมิในวงจรสายพานกระแสแบบทรานส์ลิเนียร์และวงจรขยายโอทีเอ</subject>
<studentname>นาย เฉลิมภัณฑ์ ฟองสมุทร</studentname>
<abstractcontent>ถ้าเกิดการลัดวงจรที่รุนแรงอาจเกิดการระเบิดได้และฉนวนของขดลวดอาจเกิดการชำรุดเสียหายได้</abstractcontent>

หลังจากนั้นนำเอกสารทั้ง 3 มาตัดคำที่พบบ่อยทิ้ง นับความถี่ของคำในแต่ละแท็กของแต่ละเอกสารเพื่อนำไปคำนวณค่าน้ำหนักของคำดัชนี ผลที่ได้จะขอแสดงเฉพาะของเอกสาร d_1 เท่านั้น ตารางที่ 2 แสดงการคำนวณความถี่ของคำดัชนีและค่าน้ำหนักคำดัชนีของเฉพาะเอกสาร d_1

ส่วนตารางที่ 3 แสดงการหาค่า idf_i ของคำในระบบตัวอย่างเช่น คำว่า "เพชร" จะมีค่า $w_{i,j}$ = 2/2*0.477 = 0.477

ค่า $w'_{i,j}$ คำนวณจากการคิดค่าน้ำหนักของแท็กร่วมด้วย แต่คิดค่าน้ำหนักของแท็กโดยไม่ได้สนใจกับความถี่ของคำที่พบในแท็กนั้น ดังสมการที่ (4) จากการคำนวณ คำว่า "เพชร" จะมีค่า $w'_{i,j}$ = (0.8*0.4)*(2/2)*0.477 = 0.153

ค่า $w''_{i,j}$ คำนวณจากสมการที่ (6) คำว่า "เพชร" จะมีค่า $w''_{i,j}$ = (((1*0.8)+(1*0.4))/2)*0.477 = 0.286

จะพบว่า โดยวิธีที่ 1 คำว่า "เพชร" จะถูกเรียงอยู่ในลำดับที่ 1 โดยวิธีที่ 2 จะอยู่ในลำดับที่ 3 โดยวิธีที่ 3 จะอยู่ในลำดับที่ 1 ให้สังเกตว่าค่าน้ำหนักคำยิ่งสูง ลำดับของคำนั้นก็ยิ่งดี ถึงแม้ว่าการจัดลำดับด้วยวิธีที่ 3 และวิธีที่ 1 จะให้ผลเท่ากัน แต่วิธีที่ 1 จะมีคำที่อยู่ในลำดับที่ 1 หลายคำ คือ คำว่า "ศึกษา" และ "สารกึ่งตัวนำ" แต่วิธีที่ 3 จะมีอยู่เพียงคำเดียว ประกอบกับตำแหน่งของคำที่มีอยู่ คำว่า "เพชร" ควรจะมีความสำคัญที่สุด เนื่องจากปรากฏในแท็กที่สำคัญคือ <subject> และแท็ก <abstractcontent> ดังนั้นวิธีที่ 3 จึงเป็นวิธีการคำนวณที่ดีกว่าวิธีอื่นเนื่องจากสามารถทำให้เกิดความแตกต่างของค่าน้ำหนักคำที่ขึ้นกับแท็กที่คำนั้นปรากฏอยู่ได้ดี

6. การค้นคืนเอกสาร

สำหรับการคำนวณการค้นคืนเอกสาร เราใช้โมเดลการคำนวณเหมือนกับการคำนวณของเวกเตอร์โมเดลซึ่งมีการคำนวณดังต่อไปนี้

เมื่อได้คำค้น q จะทำการเปลี่ยนคำค้นให้อยู่ในรูปเวกเตอร์ $w_{i,q}$ ของคำดัชนี k_i ในระบบโดยใช้สมการ

$$w_{i,q} = \left(\frac{freq_{i,q}}{\max_l freq_{l,q}}\right) \times \log\frac{N}{n_i} \qquad (7)$$

หลังจากนั้นนำเวกเตอร์คำค้น $w_{i,q}$ มาคำนวณหาค่าความเหมือน $sim(d_j, q)$ ร่วมกับเวกเตอร์เอกสาร $w_{i,j}$ ด้วยสมการ

$$sim(d_j,q) = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}} \qquad (8)$$

ตารางที่ 2 แสดงตัวอย่างการคำนวณค่าน้ำหนักคำของเอกสาร d_1

คำดัชนี	$f_{1,j}$	w_1	$f_{2,j}$	w_2	$f_{3,j}$	w_3	Sum freq	$w_{i,j}$	$w'_{i,j}$	$w''_{i,j}$
ศึกษา	1	0.8					1	0.477	0.286	0.153
เลือก	1	0.8					1	0.239	0.191	0.191
สร้าง	1	0.8					1	0.239	0.191	0.191
ฟิล์ม	1	0.8					1	0.239	0.191	0.191
เพชร	**1**	**0.8**			1	0.4	**2**	**0.477**	**0.153**	**0.286**
พื้นที่	1	0.8					1	0.239	0.191	0.191
วิธี	1	0.8					1	0.088	0.07	0.07
cvd	1	0.8					1	0.239	0.191	0.191
ความ	1	0.8					1	0.088	0.07	0.07
ร้อน	1	0.8					1	0.239	0.191	0.191
กอบศักดิ์			1	0.6			1	0.239	0.143	0.143
ศรีประภา			1	0.6			1	0.239	0.143	0.143
สารกึ่งตัวนำ					2	0.4	2	0.477	0.191	0.191
ชนิด					1	0.4	1	0.239	0.095	0.095
ใหม่					1	0.4	1	0.239	0.095	0.095
งาน					1	0.4	1	0.239	0.095	0.095
ซิลิกอน					1	0.4	1	0.239	0.095	0.095
แกลเลียม					1	0.4	1	0.239	0.095	0.095
อาร์เซไนด์					1	0.4	1	0.239	0.095	0.095
รูป					1	0.4	1	0.239	0.095	0.095

หลังจากนั้นจะนำค่าความเหมือน $sim\ (d_{j,\ q})$ มาเรียงลำดับ โดยที่ถ้าค่าความเหมือนของคำค้นกับเอกสารหมายเลขใดมีค่าสูงสุด แสดงว่าเอกสารนั้นมีความเหมือนกับคำค้นมากที่สุด

ตารางที่ 3 แสดงค่า idf_i ของคำในระบบ

คำดัชนี	จำนวนเอกสารที่พบ	ค่า log (N/n_i)
ศึกษา	1	0.477
เลือก	1	0.477
สร้าง	1	0.477
ฟิล์ม	1	0.477
เพชร	1	0.477
พื้นที่	1	0.477
วิธี	2	0.176
cvd	1	0.477
ความ	2	0.176
ร้อน	1	0.477
กอบศักดิ์	1	0.477
ศรีประภา	1	0.477
สารกึ่งตัวนำ	1	0.477
ชนิด	1	0.477
ใหม่	1	0.477
งาน	1	0.477
ซิลิกอน	1	0.477
แกลเลียม	1	0.477
อาร์เซไนด์	1	0.477
รูป	1	0.477
หนึ่ง	1	0.477

7. ผลการทดลอง

บทคัดย่อที่ใช้ในการทดลองมีทั้งหมด 259 ฉบับ ได้นำมาจากการเก็บรวบรวมของบัณฑิตวิทยาลัย สจล. ซึ่งจัดเก็บบทคัดย่อในรูปแบบไฟล์ Microsoft Words ในการทดลองได้ทำการแปลงเอกสารให้อยู่ในรูปเอกสาร XML บทคัดย่อก่อน (ดังที่ได้เสนอในหัวข้อที่ 3) เมื่อใช้วิธีการคำนวณหาค่าน้ำหนักทั้ง 3 วิธี กับบทคัดย่อทั้งหมดปรากฏว่า กลุ่มคำที่มีค่าน้ำหนักสูงสุดในแต่ละวิธีมีความใกล้เคียงกัน มีความแตกต่างกันไม่มาก ดังแสดงตัวอย่างผลการจัดเรียงลำดับค่าน้ำหนักของคำในเอกสารบทคัดย่อวิทยานิพนธ์เรื่อง "การศึกษาการเลือกสร้างฟิล์มเพชรเฉพาะพื้นที่ด้วยวิธี CVD แบบความร้อน" ในตารางที่ 4

เพื่อให้เห็นข้อแตกต่างของแต่ละวิธี จึงได้ทำการทดลองกับเอกสารชุดใหม่ที่เป็นบทความวิจัยที่ได้รับการตีพิมพ์ในวารสารพระจอมเกล้าลาดกระบัง จำนวน 50 ฉบับ ซึ่งมีโครงสร้างเอกสารดังแสดงไว้ในหัวข้อที่ 3

ตารางที่ 4 แสดงผลการคำนวณค่าน้ำหนักคำในบทคัดย่อด้วยค่า $w_{i,j}$, $w'_{i,j}$ และ $w''_{i,j}$ ตามลำดับ

วิธีที่ 1		วิธีที่ 2		วิธีที่ 3	
คำดัชนี	$w_{i,j}$	คำดัชนี	$w'_{i,j}$	คำดัชนี	$w''_{i,j}$
cvd	3.34	cvd	3.73	cvd	4.39
diamond	2.22	diamond	2.56	ตัวนำ	2.92
films	2.22	films	2.56	diamond	2.92
ฟิล์มเพชร	1.61	ฟิล์มเพชร	1.61	films	2.92
selective-area	1.36	selective-area	1.35	กึ่ง	2.00
ตัวนำ	1.17	ตัวนำ	1.17	สาร	1.85
deposition	1.17	deposition	1.17	แกลเลียม	1.69
thermal	1.17	thermal	1.17	gaas	1.69
เพชร	1.06	เพชร	1.06	si	1.69
ฟิล์ม	1.06	ฟิล์ม	1.06	กอบศักดิ์	1.69

ตารางที่ 5 แสดงผลการคำนวณค่าน้ำหนักคำในบทความวิชาการด้วยค่า $w_{i,j}$, $w'_{i,j}$ และ $w''_{i,j}$ ตามลำดับ

วิธีที่ 1		วิธีที่ 2		วิธีที่ 3	
คำดัชนี	$w_{i,j}$	คำดัชนี	$w'_{i,j}$	คำดัชนี	$w''_{i,j}$
แกลบ	2.36	แกลบ	302.575	เตา	102.5
เตา	2.31	เผา	231.34	แกลบ	94.71
เผา	1.84	combustion	221.17	เผา	68.53
เชื้อเพลิง	1.65	ไหม้	160.52	เชื้อเพลิง	67.61
ไหม้	1.24	เตา	95.98	vortex combustor	51.03
มม	0.69	เชื้อเพลิง	58.39	rice husk fuel sizes	51.03
ไหล	0.5	vortex combustor	51.03	การเผาไหม้	51.03
อากาศ	0.5	rice husk fuel sizes	51.03	ขนาดเชื้อเพลิงแกลบ	51.03
กระจาย	0.45	การเผาไหม้	51.03	ไหม้	48.73
อุณหภูมิ	0.4	ขนาดเชื้อเพลิงแกลบ	51.03	combustion	46.86

จากการคำนวณหาค่าน้ำหนักคำทั้ง 3 วิธีในตารางที่ 5 ปรากฏว่า การคำนวณวิธีที่ 3 ทำให้ได้คำหรือกลุ่มคำที่มีความสำคัญในบทความนั้นอยู่ลำดับต้น ๆ ซึ่งเราเห็นว่าคำที่ควรอยู่ในลำดับต้น ๆ ของตารางนั้นควรปรากฏในแท็กคำสำคัญๆ ได้แก่ แท็กคำสำคัญ แท็กชื่อเรื่อง แท็กเนื้อหาบทคัดย่อ เป็นต้น เนื่องจากผลการทดลองนี้ เป็นการเรียงลำดับคำดัชนีจากบทความเรื่อง "อิทธิพลของขนาดของแกลบต่อคุณลักษณะการเผาไหม้" ซึ่งมีคำสำคัญ "เตาวอร์เทค, ขนาดเชื้อเพลิงแกลบ, การเผาไหม้,Vortex combustor, rice husk fuel sizes, combustion" จะพบว่าคำดัชนีที่จัดเรียงด้วยวิธีที่ 3 จะมีความสอดคล้องกับคำที่สำคัญมากกว่าของวิธีอื่น ๆ

ผลการทดลองกับทั้ง 50 ฉบับ แต่นำมาแสดงเพียง 25 ฉบับ ได้ผลดังตารางที่ 6 และ 7 (แสดงไว้ต่อท้ายบทความ) ซึ่งผลที่ได้สอดคล้องกับการทดลองก่อนหน้านี้

8. สรุป

เราได้นำเสนอวิธีการเรียงลำดับคำดัชนีในเอกสาร XML ซึ่งนอกจากจะใช้ความถี่ของคำที่ปรากฏในเอกสารเพื่อการจัดเรียงคำดัชนีตามวิธีเวคเตอร์โมเดลมาตรฐานแล้ว ยังนำเอาค่าน้ำหนักของแท็ก XML ที่คำดัชนีนั้น ๆ ปรากฏอยู่มาพิจารณาร่วมด้วย ผลที่ได้ทำให้คำดัชนีที่ปรากฏในแท็กที่มีความสำคัญสูงกว่ามีแนวโน้มที่จะมีอันดับในการจัดเรียงสูงกว่าเมื่อเทียบกับคำดัชนีที่ปรากฏในแท็กที่มีความสำคัญน้อยกว่า วิธีการนี้ได้นำไปทดสอบจัดเรียงลำดับคำดัชนีกับตัวอย่างเอกสาร XML ที่เป็นบทคัดย่อวิทยานิพนธ์และบทความวิจัยจำนวนมาก ซึ่งให้ผลยืนยันหลักการดังกล่าว

กิตติกรรมประกาศ

คณะผู้วิจัยขอขอบคุณบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้ความอนุเคราะห์ด้านข้อมูลบทคัดย่อวิทยานิพนธ์ของนักศึกษา อุปกรณ์เครื่องคอมพิวเตอร์ และทุนสนับสนุนการวิจัย ตลอดจนขอขอบคุณบุคลากรของบัณฑิตวิทยาลัยทุกท่านที่ให้ความช่วยเหลือด้วยดีมาโดยตลอด รวมทั้งขอขอบคุณงานส่งเสริมการวิจัยของทางวารสารพระจอมเกล้าลาดกระบัง ที่เอื้อเฟื้อข้อมูลเอกสารบทความวิจัยที่ได้ลงตีพิมพ์ในวารสารเพื่อใช้ในการทดลองในงานวิจัยชิ้นนี้

เอกสารอ้างอิง

[1] วุฒิชัย ปิยะพันธวงศ์. **"การสืบค้นข้อมูลอิงแบบจำลองออปเจคสำหรับการทำดัชนีเอกสาร XML."** วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย, มหาวิทยาลัยเกษตรศาสตร์, 2545.

[2] ธนุสศักดิ์ ธัญญสิริ. **"การออกแบบและพัฒนาระบบต้นแบบสำหรับการสืบค้นข้อมูลสารสนเทศภาษาไทยด้วยดัชนีหลายระดับ."** วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย, มหาวิทยาลัยเกษตรศาสตร์, 2543.

[3] Evangelos Kotsakis. "Structured Information Retrieval in XML documents", **SAC 2002: 9th Annual Workshop on Selected Areas in Cryptography**, Madrid, Spain, 2002. pp 663-667.

[4] Gerad Salton., Christopher Buckley. "Term-weighting Approaches in Automatic Text Retrieval." **Information Processing & Management**, Vol. 24, No. 5, 1988. pp. 513-523.

[5] Tim Bray, Jean Paoli and C. M. Sperberg-McQueen. Extensible Markup Language (XML) 1.0, W3C Recommendation, available at **http://www.w3.org**/TR/1998/ REC-xml-19980210.

[6] Ricardo Baeza-Yates, Berthier Ribeiro-Neto. **Modern Information Retrieval.** Addison Wesley Longman Limited, 1999.

[7] DCMI Usage Board, DCMI Metadata Terms, available at **http://dublincore.org/documents/2003/03/04/dcmi-terms**

ตารางที่ 6 แสดงรายชื่อบทความ 25 บทความ และคำสำคัญของบทความนั้น ๆ

บทความ	ชื่อบทความ	ความสำคัญ
1	อิทธิพลของขนาดของแกลบต่อคุณลักษณะการเผาไหม้	เตาอร์เทค, ขนาดเชื้อเพลงแกลบ, การเผาไหม้, vortex combustor, rice husk fuel sizes, combustion
2	การทำนายเชิงตัวเลขของการไหลปั่นป่วนหมุนวนแบบรุนแรงในเตาเผา	แบบจำลองความปั่นป่วน, วิธีปริมาตรลีบเนื่อง, การไหลปั่นป่วนหมุนวนแบบรุนแรง, เตาเผา, turbulent model
3	การจำลองคุณลักษณะการสเปรย์ของเครื่องยนต์เชื้อเพลิงก๊าซธรรมชาติแบบฉีดตรงเข้าห้องเผาไหม้	ก๊าซธรรมชาติ, สเปรย์, แบบจำลอง, การทะลุทะลวง, natural gas, spray, model
4	การสร้างภาพตัดขวางโดยใช้คลื่นอัลตร้าโซนิค	ฟิลเตอร์แบ็ค โปรเจคชัน, สตาร์อาร์ทิแฟค, filtered backprojection, star artifact
5	การออกแบบการลดทอนในช่วง หยุดความถี่ผ่านอย่างสูงของวงจรกรองความถี่ iir ที่มีเฟสเป็นเชิงเส้น	องค์ประกอบวงจรกรองความถี่ผ่านตลอด, ความถี่ iir ที่มีเฟสเป็นเชิงเส้น, การลดทอนในช่วงหยุดความถี่ผ่าน
6	การวิเคราะห์สมรรถนะการสื่อสารผ่านดาวเทียมเอทีเอ็มที่มีทราฟฟิกแบบคล้ายคลึงตัวเอง โดยการจำลองแบบและมีการส่งซ้ำแบบเอกซ์โพเนนเชียล	ทราฟฟิกแบบคล้ายคลึงตัวเอง, สล๊อต-อะโลฮา, เอกซ์โพเนนเชียล แบ็คออฟ, การกระจายแบบพาเรโต้, self-similar traffic, slotted-aloha, exponential backoff, pareto distribution
7	อิทธิพลของความชื้นของแกลบต่อคุณลักษณะการเผาไหม้	เตาอร์เทค, ปริมาณความชื้น, แกลบ, การเผาไหม้, vortex combustor, moisture content, rice husk, combustion
8	ปัจจัยที่มีผลต่อคุณภาพของเจลเต้าหู้หลอด ตอนที่ ii อัตราส่วนน้ำต่อถั่วและชนิดของสารตกตะกอน	เต้าหู้หลอด, อัตราส่วนน้ำต่อถั่ว, สารตกตะกอน, ปริมาณโปรตีน, packaged tofu, water, bean ratio, coagulant, protein yield
9	การจำลองคุณลักษณะการสเปรย์ของเครื่องยนต์เชื้อเพลิงก๊าซธรรมชาติแบบฉีดตรงเข้าห้องเผาไหม้	แบบจำลองความปั่นป่วน, วิธีปริมาตรลีบเนื่อง, การไหลปั่นป่วนหมุนวนแบบรุนแรง, เตาเผา, turbulent model
10	การออกแบบวงจรลูปคอสตาสที่ใช้เกตเอกซ์คลูซีฟออร์เป็นมิกเซอร์ด้านอินพุต	ก๊าซธรรมชาติ, สเปรย์, แบบจำลอง, การทะลุทะลวง, natura gas, spray, model
11	การลดปริมาณผงซักฟอกที่ไม่ได้มาตรฐานในกระบวนการผลิตผงซักฟอกชนิดธรรมดา	ลูปคอสตาส, ลูปกู้คลื่นพาห์, บีพีเอสเค, เฟสล็อกลูป (พีแอลแอล), costas loop, carrier recovery loop, bpsk, phase-locked loop (pll).
12	ผลของขนาดชิ้นสลัดผักต่ออัตราการหายใจการผลิตเอทิลีนและอายุการเก็บรักษาที่ 12 ซ	ผงซักฟอก, ผงซักฟอกที่ไม่ได้มาตรฐาน, การออกแบบทดลอง, การควบคุมกระบวนการเชิงสถิติ, สารทำความสะอาด detergent, rework, design of experiment (doe), statistical process control (spc), active detergent (ad)
13	การใช้เครื่องหมายพันธุกรรมและดิฟเฟอเร็นเชียล เอฟโวลูชัน เพื่อการตัดสินใจอนุรักษ์สัตว์พันธุ์แท้	สลัดผัก, เอทิลีน, พอลิเอทิลีน, คาร์บอนไดออกไซด์, vegetables salad, ethylene, polyethylene, carbondioxide
14	การศึกษาความคิดเห็นของมหาบัณฑิตที่มีต่อระบบการจัดการศึกษาระดับบัณฑิตศึกษา ของคณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง	เครื่องหมายทางพันธุกรรม, การอนุรักษ์พันธุกรรม, พันธุ์ที่ใกล้สูญพันธุ์, ดิฟเฟอเร็นเชียล เอฟโวลูชัน, ไมโครแซทเทลไลท์, genetic markers, genetic conservation, endangered breed, differential evolution, microsatellite
15	การศึกษาระดับความคิดสร้างสรรค์ของนักศึกษาระดับปริญญาตรี ภาควิชาครุศาสตร์สถาปัตยกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง	ระบบการจัดการศึกษา, ปัจจัย, กระบวนการ, ผลผลิต, ระดับบัณฑิตศึกษา, มหาบัณฑิต, educational management system, input, process, output, graduate study level, master degree graduates
16	การวิเคราะห์ตัวแปรที่มีผลกระทบต่อการพิมพ์ระบบเฟล็กโซกราฟฟีแบบพิมพ์พื้นผิวบนเทปกาวพอลิโพรพิลีน ด้วยวิธีการออกแบบการทดลอง	ความคิดสร้างสรรค์ เพศ ระดับผลสัมฤทธิ์ทางการเรียน, creative thinking, gender, learning achievement
17	การสังเคราะห์คาร์บอนกัมมันต์จากกะลามะพร้าว โดยวิธีการกระตุ้นทางเคมีสำหรับการดูดซับฟีนอล	คาร์บอนกัมมันต์, กะลามะพร้าว, การดูดซับ, ฟีนอล, activated carbon, coconut shell, adsorption, phenol
18	synthesis of activated carbon from coconut shell by chemical activation for the adsorption	เต้าหู้หลอด, การแช่ถั่ว, พันธุ์ถั่วเหลือง, อุณหภูมิการเกิดเจล, packaged tofu, soaking method, bean varieties, gelling temperature
19	ปัจจัยที่มีผลต่อคุณภาพของเจลเต้าหู้หลอด ตอนที่ i พันธุ์ถั่ว การแช่ถั่ว และอุณหภูมิการเกิดเจล	ผักกวางตุ้ง, ไนเตรต, ปุ๋ย, chinese leafy cabbage, nitrate, nitrite, fertilizer
20	การปลูกผักกวางตุ้งให้ได้ผลผลิตสูงและลดปริมาณไนเตรตและไนไตรต์	แพลงก์ตอนพืช, ออสซิลลาทอเรีย, ไฮโดรเจนเปอร์ออกไซด์, phytoplankton, oscillatoria, hydrogen peroxide
21	การใช้ไฮโดรเจนเปอร์ออกไซด์ควบคุมปริมาณแพลงก์ตอนพืช oscillatoria	เอนไซม์, แอลฟา-แอมิเลส, รำข้าว, น้ำตาล, enzyme, amylase, rice bran, sugar
22	การศึกษาสารละลายน้ำตาลที่ได้จากกระบวนการย่อยรำข้าวสกัดไขมันด้วยเอนไซม์แอลฟา แอมิเลส	พฤติกรรมการบริโภคอาหาร, วัยรุ่น, food consumption behavior, teenagers
23	ปัจจัยที่มีอิทธิพลต่อพฤติกรรมการบริโภคอาหารของกลุ่มวัยรุ่นในเขตกรุงเทพมหานคร	ส่วนประกอบทางเคมี, ของเสียจากโรงฟักไข่, chemical composition, hatchery wastes
24	ชนิดและวิธีการแปรรูปต่อส่วนประกอบทางเคมีในของเสียจากตู้เกิดลูกไก่	คุณภาพน้ำเชื้อ, จำนวนอสุจิต่อไข่, อัตราการปฏิสนธิ, semen qualities, sperm number per egg, fertilization rate
25	type and processing method on the chemical composition of hatcher wastes	ก๊าซธรรมชาติ, สเปรย์, แบบจำลอง, การทะลุทะลวง, nutura, gas, spray, madel

ตารางที่ 7 แสดงผลการคำนวณค่าน้ำหนักคำดัชนีของ 10 บทความจากบทความที่เลือกมาแสดง

บทความ	วิธี	คำดัชนี 1, น้ำหนัก	คำดัชนี 2, น้ำหนัก	คำดัชนี 3, น้ำหนัก	คำดัชนี 4, น้ำหนัก	คำดัชนี 5, น้ำหนัก	คำดัชนี 6, น้ำหนัก	คำดัชนี 7, น้ำหนัก	คำดัชนี 8, น้ำหนัก	คำดัชนี 9, น้ำหนัก	คำดัชนี 10, น้ำหนัก
1	1	แกลบ, 2.36	เตา, 2.3	เผา, 1.83	เชื้อเพลิง, 1.64	ไหม้, 1.23	ผง, 0.68	ไซโล, 0.5	อากาศ, 0.49	กระจาย, 0.44	อุณหภูมิ, 0.39
	2	แกลบ, 302.57	เผา, 231.33	combustion, 221.17	ไหม้, 160.51	เตา, 95.96	เชื้อเพลิง, 58.38	vortex combustor, 51.03	rice husk fuel sizes, 51.03	การเผาไหม้, 51.03	ขนาดเชื้อเพลิงแกลบ, 51.03
	3	เตา, 102.49	แกลบ, 94.7	เผา, 69.52	เชื้อเพลิง, 67.6	vortex combustor, 51.03	rice husk fuel sizes, 51.03	การเผาไหม้, 51.03	ขนาดเชื้อเพลิงแกลบ, 51.03	ไหม้, 48.72	combustion, 46.85
2	1	model, 1.97	แรง, 1.54	stress, 1.4	ท้าทาย, 1.12	reynolds, 1.02	turbulence, 0.73	สมการ, 0.7	turbulent, 0.68	ไซโล, 0.66	swirling, 0.6
	2	combustor, 193.17	ท้าทาย, 96.32	ไซโล, 86.25	strongly swirling flow, 80.02	วิธีปริมาตรสิ้นเนื่อง, 80.02	เคนเค, 80.02	turbulent model, 80.02	แบบจำลองความปั่นป่วน, 80.02	finite volume method, 80.02	การไหลปั่นป่วนหนุนวนแบบรุนแรง, 80.02
	3	strongly swirling flow, 80.02	วิธีปริมาตรสิ้นเนื่อง, 80.02	เคนเค, 80.02	turbulent model, 80.02	แบบจำลองความปั่นป่วน, 80.02	finite volume method, 80.02	การไหลปั่นป่วนหนุนวนแบบรุนแรง, 80.02	combustor, 68.42	model, 62.11	asm, 55.6
3	1	ฉีด, 1.97	ทะลวง, 0.87	ทะลุ, 0.87	เชื้อเพลิง, 0.86	ไหม้, 0.84	กัฟ, 0.77	จำลอง, 0.76	ของสเปรย์, 0.68	ท้า, 0.67	เผา, 0.67
	2	spray, 337.65	ฉีด, 253.63	model, 185.76	สเปรย์, 121.78	ไหม้, 118.49	เชื้อเพลิง, 113.96	จำลอง, 96.02	เผา, 94.3	กัฟ, 89.26	การทะลุทะลวง, 73.35
	3	ฉีด, 121.65	spray, 31.14	การทะลุทะลวง, 73.35	natural gas, 73.35	สเปรย์, 69.72	กังหันธรรมชาติ, 60.35	ทะลวง, 55.66	ทะลุ, 55.66	แบบจำลอง, 52.75	จำลอง, 51.98
4	1	ชวาง, 2.43	คลื่นอัลตร้า, 1.27	ลัด, 1.23	พิรคณิต, 0.99	ภาพ, 0.92	โซ, 0.85	filtered, 0.74	ลา, 0.74	backprojection, 0.67	นิต, 0.64
	2	ชวาง, 349.57	ลัด, 177.32	คลื่นอัลตร้า, 152.45	ภาพ, 138.36	โซ, 108.32	นิต, 73.53	สร้าง, 66.74	พิรคณิต, 37.62	สตาร์อาร์ทิฟฟ, 36.68	ฟิลเตอร์แบ็คโปรเจคชั่น, 36.68
	3	ชวาง, 136.27	ลัด, 69.13	พิรคณิต, 66.39	ภาพ, 53.92	คลื่นอัลตร้า, 52.99	filtered, 38.71	backprojection, 37.49	สตาร์อาร์ทิฟฟ, 36.68	ฟิลเตอร์แบ็คโปรเจคชั่น, 36.68	filtered backprojection, 36.68
5	1	วงจร, 2.28	ความถี่, 1.71	ฟิลเตอร์, 1.09	กรอง, 1	โอน, 0.99	ม, 0.99	เรา, 0.37	ทอน, 0.36	สแอน, 0.36	allpass, 0.35
	2	วงจร, 267.47	ความถี่, 207.49	กรอง, 123.34	ความถี่ ม ริมิฟสะเป็นเชิงเส้น, 50.3	การลดทอนในช่วงหยุดความถี่ผ่าน, 50.3	องค์ประกอบวงจรกรองความถี่ผ่านตลอด, 50.3	allpass complementary filter, 50.3	linear phase iir filters, 50.3	stopband attenuation, 50.3	ทอน, 50.08
	3	วงจร, 102.78	ความถี่, 83.78	ฟิลเตอร์, 51.42	ความถี่ ม ริมิฟสะเป็นเชิงเส้น, 50.3	การลดทอนในช่วงหยุดความถี่ผ่าน, 50.3	องค์ประกอบวงจรกรองความถี่ผ่านตลอด, 50.3	allpass complementary filter, 50.3	linear phase iir filters, 50.3	stopband attenuation, 50.3	กรอง, 48.52
6	1	เซลล์, 1.51	พาเก็ต, 0.96	สถานี, 0.63	ทอง, 0.63	traffic, 0.63	ซัน, 0.6	คล้ายคลึง, 0.59	สล็อตอะโลฮา, 0.56	lp, 0.56	วิสัย, 0.56
	2	คล้ายคลึง, 68.19	traffic, 65.53	ที่, 65.49	ล็อสาร, 64.14	self similar traffic, 62.97	การกระจายแบบพาเรโต้, 62.87	ทราฟฟิกแบบคล้ายคลึงตัวเอง, 62.87	เอกซ์โพเนนเชียล แบ็คออฟ, 62.87	exponential backoff, 62.87	slotted-aloha, 62.87
	3	self-similar traffic, 62.87	การกระจายแบบพาเรโต้, 62.87	ทราฟฟิกแบบคล้ายคลึงตัวเอง, 62.87	เอกซ์โพเนนเชียล แบ็คออฟ, 62.87	exponenual backoff, 62.87	slotted-aloha, 62.87	pareto distribution, 62.87	สล็อต-อะโลฮา, 62.87	เซลล์, 60.93	พาเรโต้ 28.64
7	1	เผา, 1.83	แกลบ, 1.78	เตา, 1.78	เชื้อเพลิง, 1.2	ไหม้, 1.14	equivalence, 0.89	ความชื้น, 0.77	ทุ, 0.51	อากาศ, 0.5	ratio, 0.5
	2	แกลบ, 1892.93	เผา, 230.91	ไหม้, 131.42	ความชื้น, 97.25	combustion, 61.6	เตา, 51.25	การเผาไหม้ vortex combustor, 42.94	moisture content, 42.94	ปริมาณความชื้น, 42.94	rice husk, 42.94
	3	แกลบ, 100.1	เตา, 60.68	เผา, 52.97	การเผาไหม้ vortex combustor, 42.94	moisture content, 42.94	ปริมาณความชื้น, 42.94	rice husk, 42.94	เชื้อเพลิง, 39.3	equivalence, 37.68	เตาอาร์ทฟ, 35.33
8	1	เจลเต้าหู้, 1.88	ตกตะกอน, 1.77	ถั่ว, 1.72	gdl, 1.14	caso, 1.11	หลอด, 1.11	aggregates, 0.9	อัตราส่วน, 0.75	bc, 0.7	สัมผัส, 0.6
	2	coagulant, 277.37	เจลเต้าหู้, 248.63	ถั่ว, 241.57	ตกตะกอน, 228.3	หลอด, 142.03	อัตราส่วน, 96.17	อัตราส่วนน้ำต่อถั่ว, 76.54	water	bean ratio, 76.54	ปริมาณโปรตีน packaged tofu, 76.54
	3	อัตราส่วนน้ำต่อถั่ว, 76.54	water	bean ratio, 76.54	ปริมาณโปรตีน packaged tofu, 76.54	สารตกตะกอน, 76.54	protein yield, 76.54	coagulant, 72.17	เจลเต้าหู้, 62.98	เต้าหู้หลอด, 62.98	ถั่ว, 61.07
9	1	ฉีด, 1.97	ทะลวง, 0.87	ทะลุ, 0.87	เชื้อเพลิง, 0.86	ไหม้, 0.84	กัฟ, 0.77	จำลอง, 0.76	ของสเปรย์, 0.68	ท้า, 0.67	เผา, 0.57
	2	spray, 311.67	ฉีด, 253.63	สเปรย์, 112.41	เชื้อเพลิง, 110.46	ไหม้, 109.37	จำลอง, 96.94	กัฟ, 89.88	เผา, 87.05	model, 67.71	การทะลุทะลวง natural gas, 67.71
	3	ฉีด, 119.86	spray, 74.9	model, 67.71	การทะลุทะลวง natural gas, 67.71	สเปรย์, 60.66	ทะลวง, 56.33	ทะลุ, 56.33	กังหันธรรมชาติ, 55.71	จำลอง, 50.92	แบบจำลอง, 48.69
10	1	วงจร, 2.63	เลค, 2.17	มอ, 2.14	อินพุต, 1.87	vco, 1.79	การ์ด, 1.62	xor, 1.62	สัญญาณ, 1.61	ความถี่, 1.49	ถี่, 1.48
	2	bpsk, 1200.99	อินพุต, 231.42	มัดชร, 136.75	อาร์ต้น, 105.19	วงจร, 89.28	phase-locked loop (pll), 78.24	ลูปคอสตาส, 78.24	carrier recovery loop, 78.24	เฟสล็อกลูป (พีแอลแอล) costas loop, 78.24	บีพีเอสเค, 78.24
	3	bpsk, 146.05	เลค, 135.19	มอ, 126.29	วงจร, 110.04	การ์ด, 96.5	สัญญาณ, 89.55	phase-locked loop (pll), 78.24	ลูปคอสตาส, 78.24	carrier recovery loop, 78.24	เฟสล็อกลูป (พีแอลแอล) costas loop, 78.24

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายเกียรติณรงค์ ทองประเสริฐ
วัน เดือน ปีเกิด	28 พฤศจิกายน 2520
ที่อยู่	130/1 ม.3 ต.บางน้ำเชี่ยว อ.พรหมบุรี จ.สิงห์บุรี 16120
ประวัติการศึกษา	สำเร็จการศึกษาวิศวกรรมศาสตรบัณฑิต สาขาวิชา วิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปี 2542