JIT Mahasarakham University, Thailand http://jit.it.msu.ac.th

Journal of Applied Informatics and Technology, 2(1): 19-29

Research Article

การเพิ่มประสิทธิภาพการค้นคืนเอกสารโดยใช้วิธีการวัดความคลายคลึงระหว่างคำเชิงความหมาย

Optimizing Document Retrieval by Measurement Resemblance Between Semantic Word Methods

กมลวรรณ รัชตเวชกุล[1], อภิชัย สารทอง[2], วีระยุทธ รัชตเวชกุล[3] และยงยุทธ รัชตเวชกุล[4]
Kamonwan Ratchatawetchakul[1], Apichai Sarnthong[2],
Weerayut Ratchatawetchaku[3] and Yongyut Ratchatawetchakul[4]

[1] Department of Digital Technology, Faculty of A, Kalasin University, Thailand
[2] Department of Computer and Automation Engineering, Faculty of Engineering and Industrial Technology, Kalasin University, Thailand
[3] Department of Management, Faculty of Management Science, Loei Rajabhat University, Thailand
[4] Department of Business Computer, Mahasarakham Business School, Mahasarakham University, Thailand
kamonwan.ku@ksu.ac.th, apichai.sa@ksu.ac.th, artdy@hotmail.com, yongyut.r@mbs.msu.ac.th,

Received: 08 May 2019
Revised: 17 September 2020
Accept: 01 April 2020

Keywords:

Ontology, Keywords, Metadata, SKOS, Term Weighting

คำสำคัญ:

ออนโทโลยี, คำสำคัญ, เมตาดาตา, ระบบการจัดองค์ความรู้อย่างง่าย, การให้น้ำหนักคำ

บทคัดย่อ: บทความนี้มีวัตถุประสงค์เพื่อนำเสนอผลการทดลองและเปรียบเทียบวิธีเพิ่มประสิทธิภาพการค้นคืนเอกสารโดยใช้วิธีการวัดความคล้ายคลึงระหว่างคำเชิงความหมาย ซึ่งในทดลองครั้งนี้ผู้วิจัยได้นำข้อมูลทดสอบคือเอกสารในโดเมนคอมพิวเตอร์ จากโครงการเครือข่ายห้องสมุดในประเทศไทย (ThaiLIS: Thai Library Integration System) จำนวน 50 รายการ จากนั้นนำมาจัดทำเมทาดาทาตามมาตรฐานดับลินคอร์ให้กับเอกสารประกอบด้วย Title, Keyword/Subject, Description/Abstract และ Source การจัดทำ Ontology ใน รูปแบบภาษา Web Ontology Language (OWL) โดยใช้คำศัพท์ จาก ศัพท์สัมพันธ์ตามคลัสเตอร์ สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช) จำนวน 402 คำ ซึ่งเป็นแหล่งรวมคำศัพท์สัมพันธ์ (Taxonomy & Thesaurus) สำหรับการวิจัยครั้งนี้ได้ใช้ Dijkstra algorithm มาทำการหาระยะทางที่สั้นที่สุดจากจุดต่อ (Node) หนึ่งไปยังจุดต่อ(Node) ที่ต้องการในคลังคำ โดยกำหนดค่าความสัมพันธ์ของเส้นเชื่อม (Edge) ระหว่างจุดต่อ (Node) เป็น 5 แบบ คือ คำหลัก (skos: prefLabel) คำกว้างกว่า (skos: broader) คำแคบกว่า (skos: narrower) คำที่เกี่ยวข้อง (skos: related) และคำเหมือน (skos: altLabel) ผลการทดลองพบว่า วิธีการวัดความคล้ายคลึงเชิงความหมาย ระหว่างคำโดยใช้วิธีการหาระยะทางที่สั้นที่สุดที่ผู้วิจัยนำเสนอ สามารถเพิ่ม ประสิทธิภาพการค้นคืนเอกสารในเชิง ความหมายได้ดีกว่าการวัดความคล้ายคลึงเชิงมุม (cosine similarity) และผลลัพธ์ที่ได้จากการค้นคืนตรงกับความต้องการของผู้ใช้มากขึ้น

Abstract: The aims of article to present the method for measuring semantic similarity between words. Data test are documents in the computer domain from ThaiLIS: Thai Library Integration System 50 documents and prepare a Dublin Core metadata for documentation such as Title, Keyword/Subject, Description/Abstract and Source. Create ontology on Web Ontology Language (OWL) by the word form http://technology.in.th/thesaurus about 402 words that is thesaurus website by National Science and Technology Development Agency. This research use Dijkstra algorithm for shorted part between node to node relate by Edge and Node such as skos: prefLabel, skos: broader, skos: narrower, skos: related and skos: altLabel. The results showed that method proposed by the researcher can shows efficiency more than cosine similarity and the results of the retrieval meet users.

1. บทนำ

ระบบการค้นคืนเอกสารในปัจจุบันมีการใช้คำสำคัญ (Keywords) เป็นหลักสำหรับการค้นหาโดยไม่ได้พิจารณาจากความหมายของคำ หรือความกำกวมของคำสำคัญ โดยเฉพาะคำหนึ่งคำสามารถมีหลายความหมาย (Homonym) เช่น Apple อาจจะหมายถึงผลไม้ชนิดหนึ่งหรือเครื่องคอมพิวเตอร์ยี่ห้อหนึ่ง และคำหลายคำสามารถมีความหมายเดียวกัน (Synonym) เช่น Image และ Photo ทั้งสองคำต่างก็มีความหมายว่ารูปภาพเหมือนกัน ทำให้ผลลัพธ์ที่ได้จากการค้นคืนไม่ตรงกับความต้องการของผู้ใช้ และ จำนวนเอกสารที่ค้นคืนมาได้มีปริมาณมากเกินไปจนกระทั่งไม่สามารถเข้าถึงเอกสารได้ทั้งหมด ซึ่งปัญหาที่กล่าวมาข้างต้นทำให้นักวิจัยให้ความสำคัญกับการออกแบบและพัฒนาเทคนิควิธีสำหรับการค้นคืนเอกสารให้มีประสิทธิภาพตรงกับความต้องการของผู้ใช้ โดยเฉพาะในส่วนของการวัดความคล้ายคลึง (Strasberg, Manning, Rindfleisch and Melmon, 2000) ได้เปรียบเทียบวิธีการ 7 วิธีในการวัดค่าความคล้ายคลึงระหว่างเอกสารที่เลือกมาเป็นเอกสารตัวอย่างและเอกสารอื่นทั้งที่เกี่ยวข้องและไม่เกี่ยวข้อง ซึ่งทั้ง 7 วิธีจะแตกต่างกันไปตามวิธีการเตรียมข้อมูล การคิดค่าถ่วงน้ำหนักของคำ เพื่อใช้เป็นตัวแทนเอกสารแต่ละเอกสาร ส่วนขั้นตอนการวัดความคล้ายคลึงใช้วิธีการวัดความคล้ายคลึงเชิงมุม (Cosine Similarity) ส่วนงานที่ประยุกต์ใช้การวัดความคล้ายคลึงระหว่างเอกสาร เช่น การแบ่งกลุ่มเอกสาร (Cutting, Karger, Pedersen and J.W, 1992 ; Sahami, Yusufali and Baldonado,1998) และการค้นหาเอกสารที่เกี่ยวข้องกับเอกสารที่ผู้ใช้กำหนด (Shatkay and Wibur,2000 ; Bollacker, Lawrence and Giles, 2000) และ (Plubrukarn, Waiyamai, 2001) ได้เสนอแนวทางในการปรับปรุงประสิทธิภาพการวัดความคล้ายคลึงระหว่างเอกสาร

แนวคิดของงานวิจัยนี้คือ การนำความรู้พื้นฐานเกี่ยวกับความสัมพันธ์ระหว่างคำในแง่มุมเชิงความหมายมาใช้ โดยแนวคิดคือคำแต่ละคำมีความใกล้เคียงเชิงความหมายกับคำอื่นๆ ไม่เท่ากัน เอกสารที่คล้ายคลึงกันจึงควรมีคำที่มีความหมายใกล้เคียงกันปรากฏอยู่ จากการศึกษางานวิจัยที่เกี่ยวข้องพบว่า การวัดความคล้ายคลึงที่นิยมใช้คือการวัดความคล้ายคลึงเชิงมุม (Cosine Similarity) (G. Salton, M. J. McGill, 1983) ซึ่งเป็นการแทนเอกสารด้วยระบบเวกเตอร์และวัดมุมที่กระทำต่อกันระหว่างเอกสาร เอกสารที่มีความคล้ายคลึงกันมากจะทำมุมระหว่างกันน้อย

แบบจำลองเวกเตอร์ (Vector Space Model) มีหลักการทำงานโดยจะแทนเอกสาร (Document) และ คำ (Term) ที่ผู้ใช้สอบถาม (User Query) เข้ามาด้วยเวกเตอร์ จากนั้นทำการเปรียบเทียบหาเอกสารที่มีเวกเตอร์คล้ายคลึงกับคำค้นของผู้ใช้ให้มากที่สุด ซึ่งเป็นผลลัพธ์โดยใน

การแทนเป็นเวกเตอร์สามารถทำได้ โดยการแทนขนาดแต่ละ Dimension ของเวกเตอร์ ด้วยการให้น้ำหนักคำในเอกสาร โดยใช้วิธี tf-idf weighting และทำการเปรียบเทียบความคล้ายกันของเวกเตอร์ (Similarity Measurement) ด้วยการวัดความคล้ายคลึงด้วยโคซายน์ (Cosine) ซึ่ง Salton (1989) และ Lee (1995) ได้อธิบายว่าการแทนคำที่ปรากฏในเอกสารจะแทนคำให้อยู่ในรูปแบบของเวกเตอร์หลายมิติ (t-Dimensional Space Vector) ซึ่งมีการให้ค่าน้ำหนักของคำ (Term) โดยใช้จำนวนครั้งของคำที่สนใจที่ปรากฏในเอกสาร (Term Frequency-TF) และค่าสัดส่วนของจำนวนเอกสารทั้งหมดกับจำนวนของเอกสารที่ปรากฏคำที่สนใจ (Inverse Document Frequency-IDF) หรือเรียกว่าค่า TF-IDF แล้วเปรียบเทียบค่าความคล้ายคลึงด้วยการวัดมุมของเวกเตอร์ ซึ่งข้อดีของการวัดความคล้ายคลึงวิธีนี้คือ เป็นการจับคู่แบบวัดความคล้ายคลึงทำให้สามารถค้นคืนเอกสารได้ถึงแม้ว่าเอกสารนั้นจะไม่มีคำปรากฏอยู่ครบทุกคำ แต่มีข้อเสียคือ จะไม่มีการคำนึงถึงความหมายของคำ โดยจะถือว่าแต่ละคำเป็นอิสระต่อกัน ซึ่งความเป็นจริงแล้วคำแต่ละคำอาจมีความหมายเดียวกันหรือใกล้เคียงกัน

ดังนั้น งานวิจัยนี้จึงมีแนวคิดที่จะนำเสนอแนวทาง**การเพิ่มประสิทธิภาพการค้นคืนเอกสารโดยใช้วิธีการวัดความคล้ายคลึงระหว่างคำเชิงความหมาย**โดยใช้วิธีการหาระยะทางที่สั้นที่สุดด้วย Dijkstra Algorithm ซึ่งวิธีการนี้จะทำการค้นหาเส้นทางจากคำหนึ่งไปยังคำที่ต้องการที่มีความเกี่ยวข้องกันเชิงความหมายในคลังคำ ซึ่งมีการคำนวณจากเส้นเชื่อม (Edge) ระหว่างจุดต่อ (Node) ที่มีผลรวมน้ำหนักน้อยที่สุด ซึ่งถ้าหากคำสองคำมีค่าระยะทางห่างกันน้อยหรือสั้นที่สุดจะถือว่าคำสองคำนั้นมีความเกี่ยวพันกันเชิงความหมายค่อนข้างมาก แต่ถ้าหากคำสองคำมีค่าระยะทางห่างกันมากจะถือว่าคำทั้งสองมีความเกี่ยวพันกันเชิงความหมายน้อย ดังนั้น เอกสารที่คล้ายคลึงกันจึงไม่จำกัดอยู่เพียงแค่คำเดียวกันต้องปรากฏในแต่ละเอกสารเหมือนกัน ซึ่งคำแต่ละคำอาจมีความหมายเหมือนกัน หรือใกล้เคียงกันในเชิงความหมายกับคำอื่น ๆ ไม่เท่ากัน ดังนั้น เอกสารที่คล้ายคลึงกันจึงควรมีคำที่มีความหมายใกล้เคียงกันปรากฏอยู่ด้วย วิธีการที่นำเสนอในบทความนี้จะสามารถแก้ไขปัญหาความหมายของคำ หรือ ความกำกวมของคำค้น และได้ผลลัพธ์ที่ตรงกับความต้องการของผู้ใช้มากที่สุด

หัวข้อที่จะกล่าวถึงต่อไปในบทความวิจัยนี้จะอธิบายถึงส่วนที่ 2 คือ **แนวคิดและทฤษฎีที่เกี่ยวข้อง** ส่วนที่ 3 อธิบายถึง Methodology ซึ่งเป็นรายละเอียดของขั้นตอนการวิจัย ตั้งแต่กระบวนการเตรียมข้อมูล การวัดความคล้ายคลึงระหว่างเอกสาร และการประเมินประสิทธิภาพ ส่วนที่ 4 อธิบายถึง Experiments และส่วนสุดท้ายจะอธิบายถึง Conclusions and Further Studies

2. แนวคิดและทฤษฎีที่เกี่ยวข้อง

2.1. การวัดความคล้ายคลึง (Similarity Measure)

ระบบค้นคืนสารสนเทศขึ้นอยู่กับแนวคิดที่ว่า คำหรือเทอมที่คล้ายคลึงกันจะมีความเกี่ยวพันธ์ในการค้นหาได้จากการสอบถามเดียวกัน (Query) โดยปกติแล้วการค้นหาเอกสารที่ต้องการนั้นจะมีการจับคู่กับคำของข้อสอบถาม (Query) ในระบบการค้นคืนสารสนเทศ การวัดความคล้ายคลึงกันระหว่างคำนั้น จะอยู่บนแนวคิดที่ว่า คำที่คล้ายคลึงกัน หรือมีความหมายเกี่ยวข้องสัมพันธ์กันจะต้องถูกค้นออกมาได้จากข้อสอบถาม (Query) เดียวกัน ซึ่งโดยทั่วไปการค้นหาเอกสารจะทำโดยการจับคู่คำที่อยู่ในข้อคำถามกับคำที่ปรากฏอยู่ในเอกสาร

การวัดความคล้ายคลึงนั้น สามารถทำได้หลายวิธี เช่น String Matching/Comparison, Same Vocabulary Used, Probability that Document Arise from Same Model, Same Meaning of Text เป็นต้น

สำหรับเทคนิควิธีการวัดความคล้ายคลึง หรือความสัมพันธ์ของเอกสารในปัจจุบันมีอยู่หลายวิธี เช่น Jaccard, Cosine, Dice, และ Overlap Similarity Jaswinder, Parvinder, Yogesh, (2014) Plubrukarn.P, Waiyamai.K (2001) เคยได้นำเสนอการนำค่าความคล้ายคลึงระหว่างคำมาใช้คำนวณค่าความคล้ายคลึงระหว่างเอกสาร ซึ่งการหาค่าความเกี่ยวข้องระหว่างคำสองคำจากระยะห่างน้อยที่สุด คือการเปลี่ยนค่าระยะห่างที่น้อยที่สุดระหว่างคำเป็นค่าความเกี่ยวข้องให้อยู่ในรูปของสมการคณิตศาสตร์ที่แปรผกผันกัน ซึ่งการวัดความคล้ายคลึงทางความหมายระหว่างคำในโครงข่ายโดยพิจารณาระยะทาง (Edge Based) น้อยที่สุดระหว่างคำในเอกสารเปรียบเทียบกับทุกคำในเอกสาร และทำการจัดกลุ่มโดยเลือกกำหนดกลุ่มให้กับตัวที่มีความคล้ายคลึงมากที่สุด และหาค่าความคล้ายคลึงเฉลี่ยภายในกลุ่ม

2.2. Shortest Path and Dijkstra Algorithm

McConnell (2001) ได้อธิบายเกี่ยวกับไว้ว่าเป็นการหาเส้นทางสั้นที่สุดระหว่างจุดต่อ (Node) จากเส้นเชื่อม (Edge) โดยจะค้นหาจากทุกเส้นทางที่เป็นไปได้ กราฟที่ได้จะมีค่าน้ำหนักทั้งหมดเป็นค่าบวก มีข้อดีคือ ง่ายต่อการปรับค่าระยะทางที่สั้นที่สุด วิธีการหาเส้นทางที่สั้นที่สุดระหว่างจุดต่อ (Node) ที่นิยมใช้กันคือ Dijkstra Algorithm (Dijkstra, E., 1959) เป็นการหาระยะทางที่สั้นที่สุดจาก Vertex จุดเริ่มต้นไปยังจุดที่สองและหาไปเรื่อย ๆ จนกว่าจะเจอเส้นทางที่สั้นที่สุด กล่าวคือจะค้นหาระยะทางจากเส้นเชื่อม (Edge) ระหว่างสองจุดต่อ (Node) ที่มีผลรวมน้ำหนักน้อยที่สุด ซึ่ง Algorithm นี้จะอาศัยชุดของการวนซ้ำ โดยชุดของจุดที่แตกต่างกันจะถูกสร้างโดยการเพิ่มที่ละจุดเข้าไปในการวนซ้ำแต่ละรอบ แสดงดังภาพที่ 1

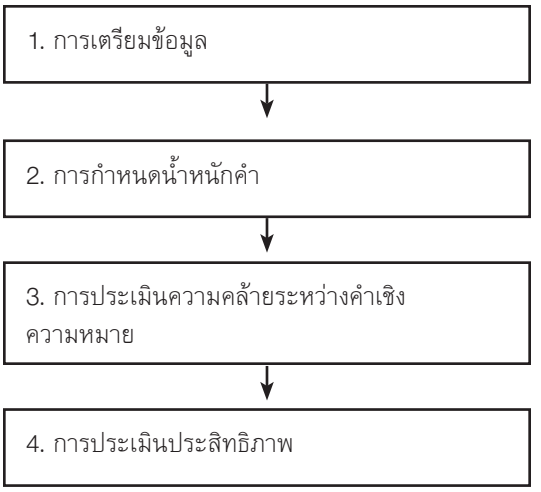
1. PriorityQueue={startVertex}
2. Until PriorityQueue ว่าง Do
    1.1 ดึงสมาชิกตัวแรกใน Queue ออกมา (ให้ชื่อว่า X)
    1.2 หาก X เคยถูกเลือกเป็นเส้นทางแล้ว กลับไปข้อ 1.1
    1.3 เลือกจุดต่อ (Node) X ให้เป็นเส้นทางจริง
    1.4 สำหรับจุดต่อ (Node) ใดๆ ที่เชื่อมต่อกับ X ให้ทำดังนี้
        1.4.1 คำนวณระยะทางรวมของจุดต่อ (Node) X มายังจุดต่อ (Node) นั้นๆ
        1.4.2 นำทุกเส้นทางไปไว้ใน PriorityQueue

ภาพประกอบที่ 1 Dijkstra Algorithm

จากภาพประกอบที่ 1 Algorithm นี้จะใช้ queue ชนิด priority queue และใช้เส้นทางรวมจากจุดเริ่มต้นมายังจุดต่อนั้นๆ เพื่อระบุ priority ของสมาชิกที่อยู่ใน queue

3. วิธีวิจัย

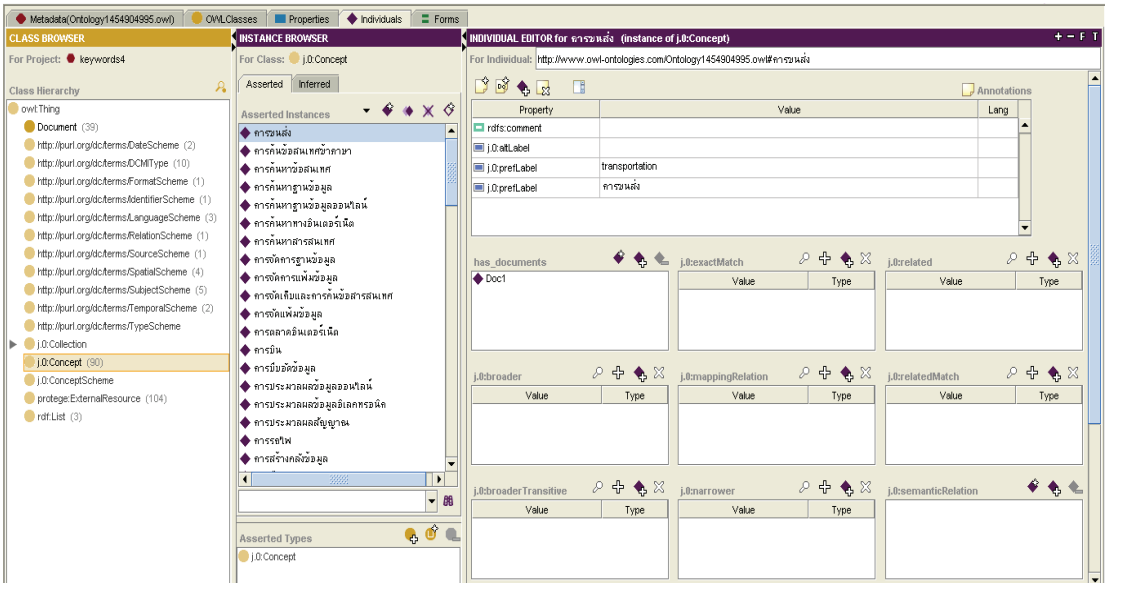
วิธีดำเนินการวิจัยในครั้งนี้ประกอบด้วย 4 ขั้นตอนหลัก ได้แก่ (1) การเตรียมข้อมูล (Data Preparation) คือ การเก็บรวบรวมข้อมูล การเตรียมข้อมูลพื้นฐานของคำและความสัมพันธ์ระหว่างคำ (2) การกำหนดน้ำหนักคำ (3) การวัดความคล้ายคลึงระหว่างคำเชิงความหมาย และ (4) การวัดประสิทธิภาพ แสดงดังภาพประกอบที่ 2



ภาพประกอบที่ 2 กรอบแนวคิดการวิจัย

จากภาพประกอบที่ 2 กรอบแนวคิดการวิจัย แสดงถึงขั้นตอนในการทำวิจัยตามขั้นตอนดังนี้

การเตรียมข้อมูล ข้อมูลทดสอบในครั้งนี้ ได้แก่ เอกสารในโดเมนคอมพิวเตอร์ ที่มีคำอธิบายครบถ้วน และเผยแพร่บนเครือข่ายทางอินเทอร์เน็ต จำนวน 50 รายการ จากนั้น ทำเมทาดาทาตามมาตรฐานดับลินคอร์ให้กับเอกสาร ประกอบด้วย Title, Keyword/Subject, Description/Abstract และ Source ลำดับต่อมาคือการจัดทำ Ontology ในรูปแบบภาษา Web Ontology Language (OWL) โดยอาศัยคำศัพท์จากโครงการเครือข่ายห้องสมุดในประเทศไทย (ThaiLIS: Thai Library Integration System) หรือ โครงการเครือข่ายห้องสมุด ในประเทศไทย สำนักงานคณะกรรมการอุดมศึกษา จำนวน 50 รายการ จากนั้น ทำเมทาดาทาตามมาตรฐานดับลินคอร์ให้กับ เอกสารประกอบ ด้วย Title, Keyword/Subject, Description/Abstract และ Source ลำดับต่อมาคือการจัดทำ Ontology ในรูปแบบภาษา Web Ontology Language (OWL) โดยอาศัยคำศัพท์จากเว็บไซตรวมคำศัพท์สัมพันธ์ (Taxonomy & Thesaurus) จำนวน 402 คำ ที่นำมาจัดทำเป็นคลังคำตามแบบแผนโครงสร้างความสัมพันธ์ของ SKOS (Simple Knowledge Organization System) ประกอบด้วย คำหลัก (skos: preLabel) คำกว้างกว่า (skos: broader) คำแคบกว่า (skos: narrower) คำเกี่ยวข้อง (Related Term-RT) และคำเหมือน (skos: altLabel) แสดงดังภาพประกอบที่ 3

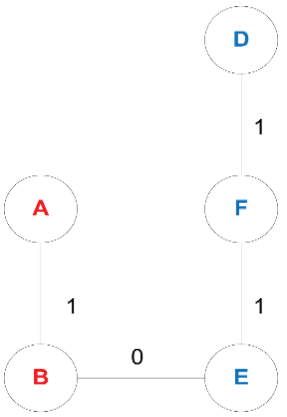


ภาพประกอบที่ 3 การสร้างคลังคำด้วย SKOS

2. การกำหนดค่าน้ำหนักคำ มีกระบวนการดังนี้ เมื่อผู้ใช้ระบุคำค้นเข้ามาเพื่อค้นคืนเอกสารที่ต้องการ กระบวนการค้นคืนจะนำคำค้นจากผู้ใช้มาหารายการคำค้นที่เกี่ยวข้องสัมพันธ์กัน (list of related keywords) ตามโครงสร้างความสัมพันธ์ในคลังคำ จากนั้น จะคำนวณค่าน้ำหนักคำดังนี้

ขั้นตอนที่ 1 กำหนดให้คำทุกคำในคลังคำมีค่าเริ่มต้นเท่ากับ 1

ขั้นตอนที่ 2 กำหนดค่าความสัมพันธ์ให้คำกว้างกว่า (Broader Term-BT) คำแคบกว่า (Narrower Term-NT) มีค่าเท่ากับ 1 และกำหนดคำเหมือน และคำเกี่ยวข้อง (Related Term-RT) มีค่าเท่ากับ 0 ตัวอย่างดังภาพประกอบที่ 4



ภาพประกอบที่ 4 การกำหนดค่าความสัมพันธ์ให้กับคำ

จากภาพประกอบที่ 4 แสดงให้เห็นว่า A (โปรแกรมประยุกต์) เป็นคำกว้างกว่า (Broader Term-BT) ของคำ B (โปรแกรมประยุกต์ทางด้านการเงิน) โดยมีคำ E (โปรแกรมประยุกต์ทางด้านบัญชี) เป็นคำเกี่ยวข้อง)Related Term-RT) และ E (โปรแกรมประยุกต์ทางด้านบัญชี) มีคำ F (โปรแกรมประยุกต์ทางธุรกิจ) F เป็นคำกว้างกว่า (Broader Term-BT) และ F เป็นคำแคบกว่า)Narrower Term-NT) ของคำ D (โปรแกรม)

ขั้นตอนที่ 3 ให้น้ำหนักคำ โดยคำนวณจากสมการที่ 1

$$T = \frac{\sum SP}{N} \qquad (1)$$

เมื่อ

SP คือ ค่าระยะทางระหว่างจุดต่อ

N คือ จำนวนจุดต่อที่มีความสัมพันธ์ระหว่างจุดต่อ(Node)

ดังนั้น ตัวอย่างการให้น้ำหนักคำ F (โปรแกรมประยุกต์ทางธุรกิจ) มีค่าเท่ากับ ((1+1)/2)=1) (B โปรแกรมระบบ))1+0)/2)=0.5 และ A (โปรแกรมประยุกต์) (1/2)=0.5 เป็นต้น และจากภาพประกอบที่ 4 น้ำหนักคำ คอมพิวเตอร์ มีค่าเท่ากับ (0+0+1+1+1)/5=0.6

ขั้นตอนที่ 4 ให้ระบุคำค้นจากผู้ใช้กับคำในคลังคำเป็นการกำหนด กลุ่มของคำค้นที่เกี่ยวข้องโดยใช้แนวทางเชิงความหมาย ซึ่งจะได้มาซึ่งกลุ่มคำค้นและรายการเอกสารที่เกี่ยวข้อง

ตัวอย่างที่ 1 ถ้าเมื่อผู้ใช้ระบุคำค้นเข้ามาเพื่อค้นคืนเอกสารใน คือ A สามารถคำนวณหากลุ่มของคำค้นได้ดังนี้

คำค้น A ∈ A ชุดคำค้นที่เกี่ยวข้อง ของ A คือ {A, B, E} ฉะนั้นน้ำหนัก Normalize คำค้นจาก กลุ่มของคำค้นในตัวอย่างที่ 1 แสดงดังตารางที่ 1

ตารางที่ 1 แสดงตัวอย่างผลการ normalize น้ำหนัก

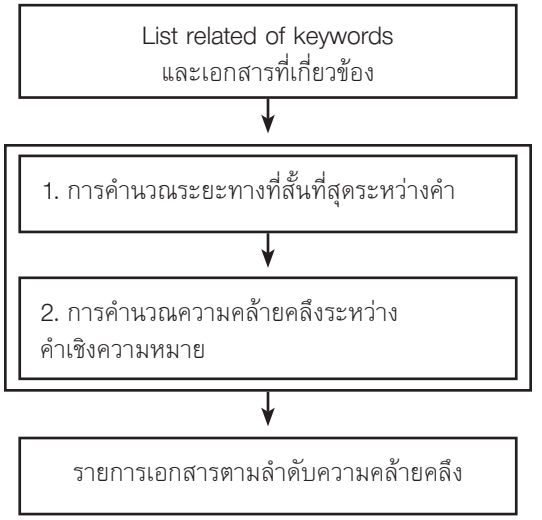
Term Weighting	คำค้น	คำที่เกี่ยวข้อง	
	A	B	E
	0.5	0.5	0.5
$\sum n$	0.33	0.33	0.33

* $\sum n$ หมายถึง การ Normalization ปรับค่าน้ำหนักคำให้มีค่าเป็น 1

จากตารางที่ 1 กลุ่มคำค้นจาก ตัวอย่างที่ 1 S={ A, B, E} มีค่าน้ำหนัก {0.33, 0.33, 0.33} ตามลำดับ

จากกระบวนการหาน้ำหนักคำในขั้นตอนนี้ ทำให้ได้รายการคำค้นที่เกี่ยวข้อง (List of Related Keywords) ที่มีการค่าน้ำหนักคำ และได้เอกสารที่เกี่ยวข้อง เพื่อนำไปใช้ในการคำนวณวัดความคล้ายคลึงเชิงความหมายในระยะที่สาม

ขั้นตอนที่ 3 คำนวณความคล้ายระหว่างคำเชิงความหมาย งานวิจัยนี้ทำโดยใช้วิธีการหาระยะทางที่สั้นที่สุด โดยนำคำสำคัญที่เกี่ยวข้อง(List of Related Keywords) ที่มีการให้ค่าน้ำหนักคำและเอกสารที่เกี่ยวข้องมาทำการคำนวณวัดความคล้ายคลึงเชิงความหมายระหว่างคำ ซึ่งกระบวนการนี้จะอาศัยค่าความเกี่ยวข้องกันหรือสัมพันธ์กันระหว่างคำสองคำ ขั้นตอนการวัดความคล้ายคลึงระหว่างเอกสาร ดังภาพประกอบที่ 5



ภาพประกอบที่ 5 แสดงขั้นตอนการวัดความคล้ายคลึงระหว่างคำเชิงความหมาย

1. การคำนวณระยะทางที่สั้นที่สุดระหว่างคำ

จะทำการคำนวณหาระยะทางจากเส้นเชื่อม (Edge) ระหว่างจุดต่อ (Node) ที่มีผลรวมน้ำหนักน้อยที่สุด โดยใช้ Dijkstra algorithm (Dijkstra, E., 1959) มีขั้นตอนดังนี้

1.1 กำหนดค่าให้กับความสัมพันธ์ของคำในคลังคำ

กำหนดคำหลัก (skos: prefLabel) คำกว้างกว่า (skos: broader) คำแคบกว่า (skos: narrower) มีค่าเท่ากับ 1 (Plubrukarn, P., Waiyamai, K., 2001) และคำเกี่ยวข้อง (Related Term-RT) คำเหมือน (skos: altLabel) มีค่าเท่ากับ 0

1.2 คำนวณระยะทางระหว่างคำที่สั้นที่สุด

การค้นหาเส้นทางจากจุดต่อ (Node) หนึ่งไปยังจุดต่อ (Node) ที่ต้องการ ใช้วิธี Dijkstra Algorithm ดังภาพประกอบที่ 1 ซึ่งตัวอย่างการคำนวณในภาพประกอบที่ 4 เช่น กำหนดให้ “A (โปรแกรมประยุกต์)” เป็นจุดเริ่มต้น สามารถคำนวณระยะทางที่สั้นที่สุดตามความสัมพันธ์ของคำในโครงสร้าง ดังนี้

PriorityQueue={startVertex}

PQ={(โปรแกรมประยุกต์, โปรแกรมประยุกต์, 0)}

Step 1: ดึงสมาชิกตัวแรกออกมา PQ={}

Step 2: เลือก “โปรแกรมประยุกต์” ให้เป็นเส้นทางจริง PQ={}

1.3 สำหรับทุกจุดต่อ (Node) ที่ต่อกับ "โปรแกรมประยุกต์" คำนวณระยะทางแล้วนำกลับไปเก็บใน PQ

PQ={(โปรแกรมประยุกต์, โปรแกรมประยุกต์ทางด้านการเงิน,1)

กระทำวนรอบจนครบกระทั่งไม่มีสมาชิกใน PQ สิ้นสุดการคำนวณ

2. การคำนวณความคล้ายคลึงระหว่างคำเชิงความหมาย ในส่วนนี้จะนำผลลัพธ์จากการคำนวณระยะทางที่สั้นที่สุดตามความสัมพันธ์ของคำในโครงสร้างมาทำการคำนวณหาค่าความคล้ายคลึงระหว่างคำเชิงความหมาย (ศุภกฤษฎิ์ นิวัฒนากูล, (2556 ดังสมการ 2

$$D_k = \frac{\sum \exp(-dis(d_j, q_i))}{N_k} \quad (2)$$

D_k แทน ค่าน้ำหนักของเอกสารต่อคำค้นที่ k

$dis(d_j,q_i)$ แทน ค่าระยะทางระหว่างคำในเอกสาร d_j กับคำค้น q_i

N_k แทน จำนวนคำในเอกสาร

ซึ่งค่าที่ได้จากสมการที่ 2 จะนำไปใช้สำหรับการคำนวณความคล้ายคลึงระหว่างเอกสารกับคำค้น ดังสมการที่ 3

$$Sim_k = \sum (Q_i * D_k) \quad (3)$$

Sim_k แทน คะแนนความคล้ายเชิงความหมายของเอกสารที่ k

Q_i แทน ค่าน้ำหนักของคำค้นที่ i

D_k แทน ค่าน้ำหนักของเอกสารต่อคำค้นที่ k

4. การประเมินประสิทธิภาพ งานวิจัยนี้ทำการประเมินการเพิ่มประสิทธิภาพการค้นคืนเอกสารด้วยวิธีการวัดความคล้ายคลึงระหว่างคำเชิงความหมาย ด้วยค่าความแม่นยำ (Precision) ค่าความระลึกหรือค่าความถูกต้อง (Recall) และค่าความแม่นยำเฉลี่ย (Mean Reciprocal Rank: MRR) เนื่องจากเป็นวิธีการที่ได้มาตรฐานและเป็นที่นิยม Frankes, William, Baeza-Yates and Ricardo, (1992) โดย Miao, Duan, Zhang, and Jiao (2009) ได้อธิบายเกี่ยวกับการประเมินประสิทธิภาพการสืบค้นไว้ ดังนี้

ค่าความแม่นยำ (Precision) คือ การพิจารณาความถูกต้องของข้อมูลที่สืบค้นได้จากอัตราส่วนระหว่างจำนวนเอกสารที่ค้นคืนถูกต้องกับจำนวนเอกสารที่ค้นคืนมาได้ทั้งหมด สูตรการคำนวณ ดังสมการ 4

$$Precision = \frac{TP}{TP} \times \%100 \quad (4)$$

เมื่อ

TP (True Prositive) แทน เอกสารที่เลือกถูกเลือกและถูกเลือกโดยผู้เชี่ยวชาญ

FP (False Negative) แทน เอกสารที่ไม่ถูกเลือกและถูกเลือกโดยผู้เชี่ยวชาญ

ค่าความระลึกหรือค่าความถูกต้อง (Recall) คือ การพิจารณาความครบถ้วนของข้อมูลเมื่อเทียบกับข้อมูลที่ควรได้ทั้งหมด จากอัตราส่วนระหว่างจำนวนเอกสารที่ค้นคืนถูกต้องกับจำนวนเอกสารที่ถูกต้องทั้งหมดของระบบ (เน้นคำตอบที่ถูกต้องที่แสดงออกมามากที่สุด) สูตรการคำนวณ ดังสมการ 5

$$Recall = \frac{TP}{TP + FN} \times \%100 \quad (5)$$

เมื่อ

TP (True Prositive) แทน เอกสารที่เลือกถูกเลือกและถูกเลือกโดยผู้เชี่ยวชาญ

FP (False Negative) แทน เอกสารที่ไม่ถูกเลือกและถูกเลือกโดยผู้เชี่ยวชาญ

ค่าความถ่วงดุล (F-Measure) คือ ค่าเฉลี่ยที่ให้ความสำคัญกับความแม่นยำและความครบถ้วนเท่า ๆ กัน โดยเป็นการเปลี่ยนค่าความถูกต้อง และค่าความแม่นยำ มารวมเป็นหนึ่งเดียว สูตรการคำนวณ ดังสมการ 6

$$F = \left(2 \times \frac{Precision \times Recall}{Precision + Recall}\right) \quad (6)$$

ค่าความแม่นยำเฉลี่ย (Mean Reciprocal Rank: MRR) โดยวัดประสิทธิภาพของการจัดอันดับ (Craswell and Hawking, 2002) สูตรคำนวณ ดังสมการ 7

$$MRR = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{r_i} \quad (7)$$

N คือ จำนวนคำค้นทั้งหมด

r_i คือ ตำแหน่งของผลลัพธ์ที่ค้นพบอย่างถูกต้องของจำนวนคำค้นที่ i

4. ผลการวิจัยและการอภิปรายผล

สำหรับการวิจัยครั้งนี้ ได้กำหนดเกณฑ์การเลือกเอกสารด้วยการกำหนดเกณฑ์ตามศุภกฤษฎิ์ นิวัฒนากูล (2556) กำหนดเกณฑ์ที่ 0.55 ตัวอย่างผลการค้นคืน 10 คำค้น จากเอกสารจำนวน 50 รายการ ดังตารางที่ 2

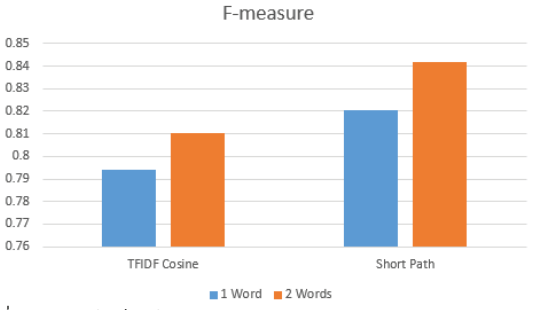
ตารางที่ 2 ผลการค้นคืนเอกสาร

ลำดับ	คำค้น	เอกสารที่เกี่ยวข้อง	วิธี	จำนวนเอกสารที่พบ	F-measure	MRR
1	อินเทอร์เน็ต	30	A	17	0.70	0.72
			B	29	0.92	0.94
2	เทคโนโลยีสารสนเทศ	43	A	30	0.76	0.77
			B	43	0.99	0.99
3	การสื่อสาร, เทคโนโลยี	25	A	20	0.80	0.96
			B	23	0.86	0.97
4	การค้นหา, ฐานข้อมูล	36	A	28	0.84	0.83
			B	33	0.86	0.83
5	สารสนเทศ, อินเทอร์เน็ต	42	A	24	0.84	0.90
			B	30	0.89	0.93
6	การสื่อสารข้อมูล	27	A	23	0.86	0.91
			B	25	0.89	0.93
7	ทฤษฎีรหัส, เครือข่าย	21	A	15	0.86	0.83
			B	19	0.87	0.95
8	วิทยาการคอมพิวเตอร์	34	A	29	0.70	0.72
			B	31	0.76	0.77
9	Programming, ภาษา	35	A	25	0.76	0.90
			B	30	0.77	0.93
10	Algorithm	26	A	19	0.69	0.74
			B	23	0.89	0.91

**วิธี A คือ TF-IDF* Cosine, *B คือ Shortest Path*

ตารางที่ 3 ผลการวัดประสิทธิภาพความแม่นยำและความถูกต้องโดยเปรียบเทียบแต่ละเทคนิควิธีด้วย F-measure

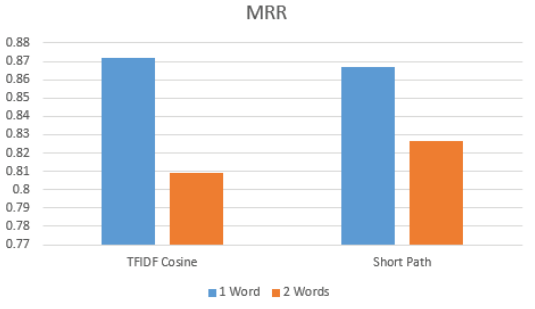
F-Measure	TF-IDF Cosine	Short Path
1 Word	0.7939672	0.8205257
2 Words	0.8103818	0.8419307



ภาพประกอบที่ 6 แสดงผลลัพธ์การวัดประสิทธิภาพความแม่นยำและความถูกต้องด้วย F-measure

ตารางที่ 4 ผลการวัดประสิทธิภาพความแม่นยำเฉลี่ยโดยเปรียบเทียบแต่ละเทคนิควิธีด้วย MRR

MRR	TF-IDF Cosine	Short Path
1 Word	0.8721751	0.8672143
2 Words	0.8094314	0.8267424



ภาพประกอบที่ 7 แสดงผลลัพธ์การวัดประสิทธิภาพความแม่นยำเฉลี่ยด้วย MRR

4. สรุปผลการวิจัย

ระบบการค้นคืนเอกสารที่มีประสิทธิภาพต้องให้ผลลัพธ์ที่ตรงกับความต้องการของผู้ใช้มากที่สุด โดยผลลัพธ์ที่ได้ต้องคำนึงถึงความหมายของคำหรือคำกำกวมของคำค้นจากผู้ใช้ได้ (Niwattanakul, S., 2013) ซึ่งการวัดความคล้ายคลึงระหว่างเอกสารเป็นขั้นตอนที่สำคัญในระบบค้นคืนเอกสารอันจะส่งผลให้ผลลัพธ์ตรงกับความต้องการของผู้ใช้ และการวัดความคล้ายคลึงเชิงมุม (Cosine Similarity) (G. Salton, M. J. McGill, 1983) เป็นวิธีการที่นิยมใช้ในปัจจุบัน (Jaswinder, S., Parvinder, S., Yogesh, C., 2015) ดังนั้นงานวิจัยนี้จึงได้นำเสนอเทคนิควิธีการวัดความคล้ายคลึงเชิงความหมายระหว่างคำโดยใช้วิธีการหาระยะทางที่สั้นที่สุดระหว่างคำหนึ่งไปยังคำที่ต้องการที่เก็บไว้ในคลังคำ ซึ่งคำมีความเกี่ยวข้องกัน หรือสัมพันธ์กันระหว่างคำโดยใช้ Dijkstra Algorithm ซึ่งจากการวิจัยพบว่าวิธีการที่นำเสนอให้ประสิทธิภาพความแม่นยำและความถูกต้องของการค้นคืนด้วยค่า F-measure ที่ดีกว่าการวัดความคล้ายคลึงเชิงมุม (Cosine Similarity) อาจเป็นเพราะวิธีการวัดความคล้ายคลึงที่ผู้วิจัยนำเสนอสามารถค้นพบเอกสารที่เกี่ยวข้องได้ดีกว่า เนื่องจากเอกสารแต่ละเอกสารอาจมีคำที่เขียนแตกต่างกันแต่มีความหมายเหมือนกันปรากฏอยู่ในเอกสารนั้นๆ แต่วิธีการวัดความคล้ายคลึงเชิงมุม (Cosine Similarity) ก็ยังให้ค่า MRR ที่ดีกว่าวิธีการวัดความคล้ายคลึงที่ผู้วิจัยนำเสนอในแง่การใช้คำค้นเพียง 1 คำค้น ทั้งนี้อาจเป็นเพราะเอกสารที่ถูกค้นพบมากขึ้นส่งผลให้ตำแหน่งของผลลัพธ์ที่ค้นพบมากขึ้นตามไปด้วย อย่างไรก็ตามวิธีการวัดความคล้ายคลึงระหว่างเอกสารที่นำเสนอสามารถนำไปประยุกต์ใช้สำหรับการค้นคืนเชิงความหมายใน Domain อื่นๆ ได้ แต่จะต้องมีการทำความรู้ให้เป็นระบบโดยเฉพาะประเภทรายการความสัมพันธ์ (Relation Lists) ที่เป็น Thesaurus, Semantic Network และ Ontology

งานวิจัย ในอนาคตอาจมีการปรับเปลี่ยนการถ่วงค่าน้ำหนักระหว่างความสัมพันธ์ของคำ

หรือคิดค้น ปรับปรุงเทคนิควิธีการค้นหาระยะทางที่สั้นที่สุดสำหรับวิธีการวัดความคล้ายคลึงเชิงความหมายระหว่างคำให้ได้ผลลัพธ์ที่มีประสิทธิภาพสูงขึ้น และที่สำคัญควรทดสอบกับเอกสารที่จำนวนเพิ่มขึ้น เพิ่มคำในคลังคำให้ครอบคลุม และทดสอบการเปลี่ยนข้อมูลเอกสาร และอภิธานศัพท์ใน Domain อื่น ๆ เช่น Domain ทางการเกษตร ด้านปศุสัตว์ การประมง และการแพทย์ เป็นต้น

เอกสารอ้างอิง

พิลาวัณย์ พลับรู้การ และกฤษณะ ไวยมัย. (2544). **รายงานการวิจัยเรื่องการวัดความคล้ายคลึงระหว่าง เอกสารโดยใช้แนวทางด้านความหมาย**. รายงานวิจัยระบบคอมพิวเตอร์และเครื่อข่ายสื่อสาร คณะ วิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์.

ศุภกฤษฎิ์ นิวัฒนากูล. (2556). **การเข้าถึงความรู้ ทางการเกษตรด้วยเทคโนโลยีเว็บเชิงความหมาย**. รายงานการวิจัย สาขาวิชาเทคโนโลยี สารสนเทศ มหาวิทยาลัยเทคโนโลยีสุรนารี.

Craswell, N. and Hawking, D. (2002). **Overview of the TREC-2002 Web Track. Technical report In Text Retrieval Conference**. Gaithersburg, Maryland.

Dijkstra, E. W. (1959). "A Note on Two Problems in Connexion with Graphs". **Numerische Mathematik, 1**: 269–271

Lee, J. H. (1995). Combining Multiple Evidence from Different Properties of Weighting Schemes, In **Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval** (pp.180-188).

McConnell, Jeffrey J. (2001). **Analysis of Algorithms: An Active Learning Approach**. Canada: Jones and Bartlett, pp. 163-168.

Sahami,M.,Yusufali, S. And Baldonado,M.Q.W. (1998). SONIA: A Service for Organizing Networked Information Autonomously. In **Proceedings of The Digital Libraries**.

Salton, G. (1989). **Automatic Text Processing: the Transformation, Analysis and Retrieval of Information by Computer**. Addison-Wesley Publishing.

Shatkay, H. and Wibur, W.J. (2000). Finding Themes in Medline Documents Probabilistic Similarity Search, In **IEEE, Advances in Digital Libraries**.

Strasberg, H.R., Manning, C.D., Rindfleisch, T.C. and Melmon, K.L. (2000). What's Related? Generalizing Approaches to Related Articles in Medicine. In **Proceedings AMIA Symp**.