

ระบบผู้นำแบบหลายเกณฑ์

จากข้อมูลแบบไฮบริด

A Multi-Criteria Recommendation
System based on Hybrid Profile

รศ.ดร. นุชรี เปรมชัยสวัสดิ์*

Assoc. Prof. Dr. Nucharee Premchaiswadi*

*คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยบูรพา

*Faculty of Information Technology, Dhurakij Pundit University

บทคัดย่อ

บทความนี้นำเสนอระบบผู้แนะนำแบบหลายเกณฑ์ด้วยวิธีการรวบรวมข้อมูลความชอบของผู้ใช้โดยทางอ้อมเพื่อแก้ปัญหาที่เกิดจากการรวบรวมจากผู้ใช้โดยตรง เนื่องจากการรวบรวมข้อมูลความชอบจากผู้ใช้ที่มีต่อไอเทมโดยตรงด้วยการให้ผู้ให้คะแนนแต่ละคุณลักษณะของไอเทมนั้น ในระบบผู้แนะนำแบบหลายเกณฑ์ต้องอาศัยความร่วมมือจากผู้ใช้เป็นอย่างมากและอาจนำไปสู่ความไม่สอดคล้องกันระหว่างข้อมูลความชอบที่ผู้ให้ไม่มีต่อแต่ละคุณลักษณะของไอเทมกับความชอบโดยรวมที่มีต่อไอเทมนั้น บทความนี้นำเสนอวิธีการรวบรวมข้อมูลแบบไฮบริดจากข้อมูลเดิมที่มีอยู่แล้ว ซึ่งประกอบไปด้วย 4 ขั้นตอนคือ 1) รวมข้อมูลความชอบที่ผู้ให้ไม่มีต่อไอเทมโดยรวมเข้ากับข้อมูลคุณลักษณะของไอเทม 2) หา global criteria weights ที่อธิบายถึงความชอบของผู้ใช้ที่มีต่อคุณลักษณะโดยทั่วไปในไอเทมนั้น แทนที่จะพิจารณาจากความชอบโดยรวมเพียงอย่างเดียว 3) ทำการวิเคราะห์และสร้างเป็นข้อมูลความชอบของผู้ใช้ที่มีต่อแต่ละคุณลักษณะของไอเทม 4) ทำการวัดค่าระดับความเข้มข้นของแต่ละคุณลักษณะที่ปรากฏอยู่ในเนื้อหาของไอเทม จากนั้นนำข้อมูลเหล่านี้มาใช้ในการแนะนำไอเทมที่เหมาะสมที่สุดให้กับผู้ใช้เป็นรายเฉพาะบุคคล

คำสำคัญ : ระบบผู้แนะนำ, หลายเกณฑ์, เกณฑ์เดียว, ความชอบ,

Collaborative Filtering, Content-based, Hybrid method

Abstract

This paper proposes a “Multi-criteria Recommender” system based on explicit gathering of users’ preferences. Normally within data gathering process, a user is asked to rate different aspects of an item founded on a sliding scale explicitly. However, individual’s preferences on each aspect of item may conflict with other preferences. To overcome such conflicts and limitations, we proposed a “Hybrid Method” derived from gathering of previous data profiles. The method creates a hybrid profile through combination of user-item data and item-related data. Having the calculated aggregated data, we can allocate global criteria weights to preferences of users. Despite of other methods, we did not use overall rating method at all. Instead, we tried to localize the obtained weights for each individual user. Consequently, the system applies the allocated weights in a way to recommend appropriate calculated items for each individual user.

บทนำ

ระบบผู้แนะนำเป็นระบบที่ถูกสร้างขึ้นมาเพื่อช่วยเหลือผู้ใช้ในการคัดกรองสิ่งที่สนใจหรือพึงพอใจจากสิ่งต่างๆ ที่มีอยู่เป็นจำนวนมาก และได้มีการนำไปประยุกต์ใช้กับงานด้านต่างๆ เช่น การแนะนำหนังสือ, แนะนำบทความข่าว (Marlin, 2003; Rosset et al., 2002) เป็นต้น นอกจากนี้ยังมีการนำมาใช้ในเชิงธุรกิจ เช่น การแนะนำสินค้าต่าง ๆ ใน Amazon (Linden et al., 2003), และการแนะนำภาพยนตร์ ใน MovieLens (Miller et al., 2003) เป็นต้น ระบบผู้แนะนำแบบดั้งเดิมส่วนใหญ่จะอธิบายความพึงพอใจของผู้ใช้ที่มีต่อไอเทม (item: สิ่งใดสิ่งหนึ่ง) ด้วยค่าคะแนนความชอบโดยรวม (overall rating) ซึ่งแสดงในรูปของตัวเลขที่มีค่าอยู่ในช่วงที่แน่นอน เช่น 1 หมายถึง พึงพอใจน้อยที่สุด และ 5 หมายถึง พึงพอใจมากที่สุด เป็นต้น ระบบผู้แนะนำแบบนี้จึงถูกพิจารณาว่าเป็นระบบผู้แนะนำแบบเกณฑ์เดียว (single-criteria) ในขณะที่ระบบผู้แนะนำสมัยใหม่พยายามที่จะทำความเข้าใจและอธิบายความพึงพอใจของผู้ใช้ที่มีต่อไอเทมโดยพิจารณาจากความชอบของผู้ใช้ที่มีต่อคุณลักษณะต่างๆ ของไอเทมซึ่งโดยทั่วไปจะมีมากกว่าหนึ่งคุณลักษณะ ระบบผู้แนะนำเหล่านี้จะถูกพิจารณาว่าเป็นระบบผู้แนะนำแบบหลายเกณฑ์ (multi-criteria) อย่างไรก็ตามก่อนที่จะระบบผู้แนะนำแบบหลายเกณฑ์จะสามารถให้บริการผู้ใช้ได้ จำเป็นต้องอาศัยข้อมูลตอบกลับจากผู้ใช้ซึ่งสะท้อนให้เห็นรสนิยมความชอบของเขาที่มีต่อคุณลักษณะต่าง ๆ เพื่อนำมาทำการวิเคราะห์โดยทั่วไปข้อมูลเหล่านี้ได้มาจากการสอบถามความเห็นจากผู้ใช้โดยตรง (explicitly) ด้วยการกำหนดให้ผู้ให้คะแนนความชอบของเขาที่มีต่อแต่ละคุณลักษณะของไอเทมที่เขาเคยเห็น

หรือบริโคมมาแล้ว วิธีการดังกล่าวนำไปสู่ปัญหาที่เรียกว่า intrusiveness problem (Joseph et al. 1997, Douglas and Kim 1998; Middleton et al. 2004; Gediminas and Kwon 2007) กล่าวคือ ผู้ใช้อาจรู้สึกไม่สะดวกที่จะต้องให้คะแนนคุณลักษณะต่างๆ ของแต่ละไอเทมที่มีมากกว่าหนึ่งคุณลักษณะ (Jung 2001; Kellar et al. 2004) เนื่องจากเป็นเรื่องของความรู้สึก ยิ่งไปกว่านั้นผู้ใช้อาจให้ข้อมูลตอบกลับที่ไม่ได้สะท้อนให้เห็นถึงความชอบที่แท้จริงของพวกเขาเลยก็ได้ (Lee et al. 2008; Palanivel and Sivakumar 2010) ปัญหาต่างๆ เหล่านี้นำไปสู่ความไม่สอดคล้องกันระหว่างข้อมูลความชอบของผู้ใช้ที่มีต่อแต่ละคุณลักษณะ (individual attribute rating) กับ ความชอบโดยรวมที่มีต่อไอเทมนั้นๆ ซึ่งเป็นอุปสรรคที่สำคัญในการที่จะใช้ประโยชน์จากข้อมูลตอบกลับดังกล่าว

ดังนั้น งานวิจัยนี้จึงได้นำเสนอกรอบวิธีการรวบรวมข้อมูลความชอบของผู้ใช้ที่มีต่อคุณลักษณะต่างๆ ของไอเทมโดยทางอ้อม เพื่อสนับสนุนการพัฒนาหรือการขยายขีดความสามารถของระบบผู้แนะนำจากเดิมที่เป็นแบบเกณฑ์เดียวไปเป็นแบบหลายเกณฑ์โดยพยายามที่จะลดปัญหา intrusiveness ให้เหลือน้อยที่สุด

ระบบผู้แนะนำ (Recommender Systems)

ระบบผู้แนะนำเกี่ยวข้องกับการพยายามที่จะทำนายว่าผู้ใช้จะให้คะแนนความชอบต่อสิ่งที่ถูกแนะนำนั้นๆ อย่างไร ซึ่งกระบวนการทำงานดังกล่าวได้ถูกจำแนกออกเป็นสองขั้นตอน (Burke, 2002) คือขั้นตอนการทำนาย (prediction phase) และขั้นตอนการแนะนำ (recommendation phase) ในขั้นตอนการ

ทำนาย ข้อมูลเกี่ยวกับความชอบของผู้ใช้ที่มีต่อไอเทมที่ผู้ใช้เคยเห็นหรือเคยใช้ และข้อมูลเกี่ยวกับคุณลักษณะของไอเทมเองที่มีอยู่ในระบบ จะถูกนำมาวิเคราะห์เพื่อทำนายค่าคะแนนความชอบของผู้ใช้ที่จะมีต่อไอเทมอื่นๆ ที่ผู้ใช้ยังไม่เคยเห็นหรือใช้มาก่อน ผลลัพธ์ที่ได้จากขั้นตอนการทำนายจะถูกนำมาใช้ในขั้นตอนการแนะนำ โดยจะมีการจัดลำดับความสำคัญของสิ่งที่จะแนะนำตามค่าที่ทำนายได้จากมากไปหาน้อย และสร้างรายการของไอเทมที่จะแนะนำให้กับผู้ใช้เพื่อให้ผู้ใช้ทำการตัดสินใจเลือกต่อไป ซึ่งการนำเสนอเช่นนี้สามารถทำได้หลายรูปแบบ เช่น นำเสนอเฉพาะสิ่งที่มีค่าทำนายสูงๆ (Top-N recommendations) เป็นต้น

โดยทั่วไปวิธีการแนะนำที่ใช้ในระบบผู้แนะนำสามารถจำแนกได้เป็นสองวิธีพื้นฐานคือ Content-based และ Collaborative Filtering สำหรับวิธี Content-based จะเลือกแนะนำโดยพิจารณาจากคุณลักษณะของไอเทมที่จะแนะนำว่ามีความคล้ายคลึงกับคุณลักษณะของไอเทมที่ผู้ใช้ปัจจุบันเคยเห็นหรือเคยบริโภคมาก่อน และผู้ใช้ได้ให้ข้อมูลตอบกลับเป็นคะแนนความชอบที่ค่อนข้างสูง โดยทั่วไปวิธีการแนะนำแบบ content-based จะเลือกแนะนำไอเทมที่มีคุณลักษณะต่างๆ คล้ายกับไอเทมที่ผู้ใช้ปัจจุบันเคยชอบมาแล้ว วิธีการนี้ต้องการข้อมูลที่ค่อนข้างสมบูรณ์จำนวนมากที่สามารถใช้ในการอธิบายคุณลักษณะของไอเทมได้ เป็นที่น่าสังเกตว่าวิธีการดังกล่าวประสบผลสำเร็จอย่างดีในระบบแนะนำที่ไอเทมต่างๆ ที่ถูกอธิบายด้วยข้อมูลที่เป็นข้อความ (textual information) เนื่องจากระบบจะสามารถตีความหรือเข้าใจได้อย่างอัตโนมัติ เช่น ระบบผู้แนะนำหนังสือ หรือแนะนำหน้า web เป็นต้น อย่างไรก็ตาม วิธี

การ content-based นั้นไม่เหมาะสมกับระบบที่ไอเทมที่ถูกอธิบายด้วยข้อมูลที่ไม่ได้อยู่ในรูปข้อความ เช่น เพลง, วิดีโอ และ ภาพยนตร์ เป็นต้น

ส่วนวิธี Collaborative Filtering จะเลือกแนะนำสิ่งที่ผู้ใช้คนอื่นๆ ในระบบที่มีลักษณะความชอบคล้ายคลึงกับผู้ใช้ปัจจุบันมากๆ และผู้ใช้เหล่านี้ได้ให้คะแนนความชอบต่อไอเทมนั้นไว้ค่อนข้างสูง ในขณะที่ผู้ใช้ปัจจุบันยังไม่เคยใช้ไอเทมนั้นมาก่อน โดยทั่วไปในวิธีแนะนำแบบ collaborative filtering แบบดั้งเดิมนั้น จะพิจารณาความชอบที่คล้ายคลึงกันระหว่างผู้ใช้จากค่าความชอบโดยรวมที่ผู้ใช้กำหนดให้กับไอเทมต่างๆ ร่วมกัน ยกตัวอย่างเช่น ในระบบแนะนำภาพยนตร์ เป็นต้น โดยทั่วไปแล้วจำนวนของภาพยนตร์จะมีปริมาณสูงกว่าจำนวนภาพยนตร์ที่ผู้ใช้เคยดูอยู่ค่อนข้างมาก เมื่อเทียบกันทำให้ในบางครั้งมีความเป็นไปได้มากที่ระบบจะไม่สามารถคำนวณค่าความคล้ายคลึงกันระหว่างผู้ใช้ได้ โดยเฉพาะอย่างยิ่งในกรณีที่ผู้ใช้ปัจจุบันเลือกดูแต่ภาพยนตร์ที่ไม่ค่อยมีหรือไม่เคยมีผู้ใช้คนอื่น ๆ เลือกดูมาก่อนเลยในระบบ ส่งผลให้ระบบไม่สามารถแนะนำภาพยนตร์ให้แก่ผู้ใช้ปัจจุบันคนนั้นได้ หรือหากแนะนำได้ ความถูกต้องก็จะไม่ดีนัก ปัญหาดังกล่าวถูกเรียกว่า sparsity problem ซึ่งถือเป็นปัญหาหลักของระบบผู้แนะนำที่ใช้วิธีการแนะนำแบบ Collaborative filtering

นอกจากสองวิธีพื้นฐานที่กล่าวมาแล้ว ยังมีวิธีการแนะนำแบบอื่นที่ถูกพัฒนาขึ้นอีกหลายวิธี เช่น Demographic based (Pazzani, 1999) ซึ่งถูกจัดอยู่ในประเภทเดียวกันกับวิธีการพื้นฐาน Collaborative Filtering เนื่องจากใช้หลักการเดียวกัน แต่จะต่างกันที่

Demographic based นั้นเปรียบเทียบความคล้ายคลึงกันของผู้ใช้โดยพิจารณาจากข้อมูลส่วนตัวพื้นฐาน เช่น อายุ, เพศ, และ อาชีพ เป็นต้น ในขณะที่ Collaborative Filtering พิจารณาจากข้อมูลพฤติกรรมการแสดงความรักพอใจต่อไอเทมต่างๆ ที่ผู้ใช้เคยบริโภค ส่วนวิธีการแนะนำแบบ Knowledge based (Burke, 2002) นั้นจะคล้ายกันกับ Content-based แต่จะพิจารณาจับคู่ระหว่างข้อมูลเกี่ยวกับคุณลักษณะของไอเทมและลักษณะของผู้ใช้ปัจจุบันประกอบกัน

อย่างไรก็ตาม วิธีการแนะนำแบบต่างๆ ที่กล่าวมาแล้วมีข้อดีและข้อเสียต่างกันไป ดังนั้นนักวิจัยหลายท่านได้พยายามพัฒนาวิธีการใหม่ๆ โดยการนำเอาเทคนิควิธีการต่างๆ เหล่านี้มาผสมผสานกันเพื่อให้ได้ข้อดีและกำจัดข้อเสียบางอย่างออกไป วิธีการแบบผสมผสานเหล่านี้เรียกว่า Hybrid approaches รายละเอียดเกี่ยวกับวิธีการผสมผสานนี้ได้มีการศึกษาเชิงสำรวจและอธิบายอย่างมากในการศึกษาของ Burke (2002,2007)

ระบบผู้แนะนำแบบเกณฑ์เดียวและหลายเกณฑ์

ระบบผู้แนะนำที่ใช้ค่าคะแนนความชอบโดยรวมเพียงอย่างเดียวเรียกว่าระบบผู้แนะนำแบบเกณฑ์เดียว ส่วนระบบผู้แนะนำที่พยายามอธิบายความพึงพอใจของผู้ใช้ที่มีต่อไอเทมโดยพิจารณาจากความชอบที่ผู้ใช้มีต่อคุณลักษณะต่างๆ ของไอเทมที่มีมากกว่าหนึ่งคุณลักษณะเรียกว่าระบบผู้แนะนำแบบหลายเกณฑ์ (multi-criteria) นักวิจัยหลายท่านเชื่อว่าการพิจารณาเพียงแค่ค่าความชอบโดยรวมของผู้ใช้ที่มีต่อไอเทมเพียงอย่างเดียว นั้น ไม่เหมาะสมในการทำนายอีกต่อไป (Alexander, 2005;

Gediminas and Youngok, 2007) เนื่องจากว่าในความเป็นจริงแล้ว การพิจารณาตัดสินใจเลือกไอเทมที่เหมาะสมของผู้ใช้ มักจะพิจารณาจากข้อมูลคุณลักษณะต่างๆ ของแต่ละไอเทม ดังนั้น การที่ระบบได้เรียนรู้และทำความเข้าใจกับข้อมูลเหล่านี้จะช่วยให้สามารถทำนายและแนะนำได้อย่างมีประสิทธิภาพมากขึ้น (Lee et al., 2002; Gediminas et al., 2011)

แม้ว่าวิธีการแนะนำแบบหลายเกณฑ์จะได้มีการศึกษาและใช้งานกันมาบ้างแล้ว แต่ส่วนใหญ่เป็นเพียงการค้นหาไอเทมที่เหมาะสมที่สุดสำหรับคนทั่วไป ไม่ใช่สิ่งที่เหมาะสมที่สุดเฉพาะบุคคล ตัวอย่างเช่น ระบบผู้แนะนำการท่องเที่ยวเฉพาะบุคคลที่ถูกพัฒนาโดยใช้เทคนิค case-based reasoning ที่แม้ว่าจะมีการจัดลำดับความสำคัญของคุณลักษณะต่างๆ เช่น สถานที่ (location), กิจกรรม (activities) และ บริการต่างๆ (services) แต่ก็เป็นการจัดลำดับความสำคัญตามความนิยมของนักท่องเที่ยวทั่วไป โดยอาศัยข้อมูลการท่องเที่ยวที่ผ่านมา (Ricci and Werthner, 2002) คุณลักษณะต่างๆ ของไอเทม ไม่ได้ถูกพิจารณาว่าเป็นเกณฑ์ที่ผู้ใช้ๆ ในการตัดสินใจ และถึงแม้ว่าจะมีหลายๆ ระบบที่พยายามเปรียบเทียบ ไอเทมต่างๆ โดยพิจารณาการจัดลำดับความสำคัญหรือให้ค่าน้ำหนักของแต่ละคุณลักษณะของไอเทม (Lee et al., 2002) อย่างไรก็ตามค่าน้ำหนักเหล่านั้นเป็นค่าที่ผู้ใช้ทั้งหมดได้ร่วมกันกำหนดให้กับระบบเพื่อนำไปจัดลำดับไอเทมที่น่าสนใจเท่านั้น ดังนั้นการแนะนำสิ่งต่างๆ ให้กับผู้ใช้แต่ละคนจึงไม่แตกต่างกัน เนื่องจากเป็นการแนะนำสิ่งที่ดีที่สุดตามความเห็นของคนทั่วไป แทนที่จะแนะนำตามความชอบส่วนตัวของแต่ละบุคคล

Adomavicius et, al (2005) ได้เสนอวิธีการเปรียบเทียบความคล้ายคลึงกันระหว่างผู้ใช้สองวิธีคือ Aggregating Similarity และ การใช้ multidimensional distance metrics โดยพิจารณาเปรียบเทียบจากข้อมูลความชอบของผู้ใช้ที่มีต่อคุณลักษณะต่างๆ ของไอเทม แทนที่จะพิจารณาแค่เพียงความชอบโดยรวมของไอเทมเหมือนวิธีดั้งเดิม สำหรับวิธีแรก ความชอบของผู้ใช้ที่มีต่อคุณลักษณะของไอเทม จะถูกแยกออกมาคำนวณเป็นค่าความคล้ายคลึงกันระหว่างผู้ใช้บนพื้นฐานของคุณลักษณะนั้นๆ จากนั้นความคล้ายคลึงกันทั้งหมดที่คำนวณได้บนพื้นฐานของแต่ละคุณลักษณะเหล่านั้น จะถูกนำมาคำนวณรวมกัน (aggregation) อีกครั้ง เพื่อหาความคล้ายคลึงกันของผู้ใช้ที่เป็นผลลัพธ์สุดท้าย (Final similarity) สำหรับวิธีที่สองนั้น ค่าความชอบของผู้ใช้ที่มีต่อแต่ละคุณลักษณะของแต่ละไอเทมจะถูกแสดงเป็นแต่ละจุดใน $k+1$ มิติ (dimension space) และ ค่าทุกค่าใน $K+1$ มิติจะถูกคำนวณเป็นค่าระยะห่างระหว่างผู้ใช้ด้วยวิธีการคำนวณระยะห่างด้วยวิธีการต่างๆ เช่น Manhattan, Euclidean และอื่นๆ เป็นต้น จากนั้นก็แปลงกลับไปเป็นค่าความคล้ายคลึงกันระหว่างผู้ใช้

จากที่ยกตัวอย่างมาข้างต้นจะเห็นได้ว่าในสภาพแวดล้อมของระบบผู้แนะนำแบบหลายเกณฑ์นั้น จำเป็นต้องรวบรวมข้อมูลความชอบของผู้ใช้ที่มีต่อคุณลักษณะต่างๆ ของไอเทม โดยระบบส่วนใหญ่จะอนุญาตให้ผู้ใช้สามารถกำหนดค่าคะแนนความชอบต่างๆ เหล่านั้นโดยตรง (explicitly) เพื่อเก็บเป็นข้อมูลที่จะนำมาใช้ในการทำนายความพึงพอใจของผู้ใช้ในภายหลัง อย่างไรก็ตามการรวบรวมข้อมูลด้วยวิธีการดังกล่าวมีผลกระทบต่อผู้ใช้ โดยผู้ใช้งานจะต้องประเมินความชอบของตนเอง และต้องใช้เวลาใน

การให้ข้อมูลกับระบบเพิ่มมากขึ้น จนอาจทำให้เกิดปัญหาที่เรียกว่า intrusiveness problem เมื่อเร็วๆ นี้ นักวิจัยกลุ่มหนึ่ง (Palanivel and Sivakumar, 2010) ได้เสนอวิธีการที่จะแก้ปัญหาดังกล่าว โดยนำข้อมูลความชอบของผู้ใช้ที่มีต่อคุณลักษณะต่าง ๆ ของไอเทม ที่ได้มีการรวบรวมไว้แล้วมาวิเคราะห์เพื่อคำนวณเป็นข้อมูลตอบกลับทางอ้อม (implicit relevance feedback) เพื่อที่จะสร้างเป็น user profile ใหม่ และนำมาใช้ในการแนะนำแบบหลายเกณฑ์ อย่างไรก็ตามข้อมูลที่ไดมาจากระบบดังกล่าว สนับสนุนเฉพาะวิธีการแนะนำแบบ Collaborative filtering เท่านั้น ดังนั้น งานวิจัยนี้ จึงไม่ได้มุ่งเน้นเพียงแค่การแก้ปัญหา intrusiveness problem เท่านั้น แต่ยังมุ่งเน้นการเพิ่มความสามารถของระบบที่จะสามารถสนับสนุนวิธีการแนะนำในแบบอื่น (Knowledge-based) ด้วย

การสร้างข้อมูลโดยรวมแบบไฮบริด (HYBRID PROFILING)

งานวิจัยได้เสนอวิธีการรวบรวมข้อมูลความชอบของผู้ใช้โดยทางอ้อมที่มีแนวคิดจากงานของพิทยาภิวิเชียร (Pittaya and Wichian, 2010) เพื่อแก้ปัญหาของวิธีการต่างๆ ที่ได้กล่าวมาแล้ว แบบไฮบริดซึ่งประกอบไปด้วย 4 ขั้นตอนคือ

1.รวมข้อมูลความชอบของผู้ใช้เข้ากับข้อมูลคุณลักษณะต่าง ๆ ของไอเทม

ขั้นตอนนี้เป็นการผนวกเอาข้อมูลความชอบโดยรวมของผู้ใช้เข้าที่มีต่อไอเทม (User-Item Rating: UIR) เข้ากับข้อมูลคุณลักษณะต่างๆ ของไอเทม (Item-Attributes Binary Values: IAB) ดังแสดงในตารางที่ 1 และ 2 ตามลำดับ ผลจากการผนวก UIR และ IAB เข้าด้วยกันจะได้เป็น

เมทริกซ์ใหม่เรียกว่า User-Item-Attribute Binary Value and Ratings (UIABR) ดังแสดงในตารางที่ 3

ตารางที่ 1 User-Item Ratings (UIR)

	m1	m2	m3	m4	m5	m6	m7
u1	4	0	0	0	2	0	3
u2	0	2	0	4	0	4	0
u3	0	3	3	0	0	2	0
u4	0	3	0	4	0	0	3
u5	0	3	0	3	0	2	0

ตารางที่ 2 Item-Attribute values (IAB)

	Action	Advent	Drama	Horror	Sci-fi	War	Western
m1	1	1	0	0	0	0	1
m2	0	0	1	0	0	0	0
m3	0	0	0	1	1	0	0
m4	1	1	1	0	0	1	0
m5	1	0	0	0	0	0	0
m6	0	1	1	0	0	0	0
m7	1	0	0	0	0	1	1

จากตารางที่ 1 สมมติว่าในระบบผู้แนะนำภาพยนตร์มีผู้ใช้จำนวน 5 คน ($U = \{u1, u2, u3, u4, u5\}$) และมีภาพยนตร์จำนวน 7 เรื่อง ($I = \{m1, m2, m3, m4, m5, m6, m7\}$) ผู้ใช้สามารถที่จะแสดงความชอบโดยรวมที่มีต่อภาพยนตร์แต่ละเรื่องที่เขาเคยดูมาแล้วในลักษณะของค่าคะแนน ซึ่งมีค่าอยู่ในช่วง 1=ชอบน้อยสุด ถึง 5=ชอบมากที่สุด (0 หมายถึงผู้ใช้นั้นยังไม่เคยดูภาพยนตร์เรื่องนั้นมาก่อน) ดังแสดงในตารางที่ 1 ส่วนตารางที่ 2 อธิบายเกี่ยวกับเนื้อหาของภาพยนตร์ด้วยประเภทของภาพยนตร์ ในที่นี้สมมติว่าในระบบผู้แนะนำภาพยนตร์นี้มีประเภทภาพยนตร์ 7 ประเภท ($A = \{action,$

adventure, drama, horror, sci-fi, war, western\}) ภาพยนตร์แต่ละเรื่องจะถูกอธิบายด้วยคุณลักษณะของประเภทภาพยนตร์ ซึ่งในความเป็นจริงแล้วภาพยนตร์แต่ละเรื่องอาจประกอบไปด้วยหลายประเภทภาพยนตร์

ตารางที่ 3 User-Item-Attribute Binary Value and Ratings (UIABR)

		Action	Advent	Drama	Horror	Sci-fi	War	Western	Overall
u1	m1	1	1	0	0	0	0	1	4
	m5	1	0	0	0	0	0	0	2
	m7	1	0	0	0	0	1	1	3
u2	m2	0	0	1	0	0	0	0	2
	m4	1	1	1	0	0	1	0	4
	m6	0	1	1	0	0	0	0	4
u3	m2	0	0	1	0	0	0	0	3
	m3	0	0	0	1	1	0	0	3
	m6	0	1	1	0	0	0	0	2
u4	m2	0	0	1	0	0	0	0	3
	m4	1	1	1	0	0	1	0	4
	m5	1	0	0	0	0	0	0	3
	m7	1	0	0	0	0	1	1	3
u5	m2	0	0	1	0	0	0	0	3
	m4	1	1	1	0	0	1	0	3
	m6	0	1	1	0	0	0	0	2

ตัวอย่างเช่น ภาพยนตร์เรื่องที่ 6 ถูกอธิบายด้วยประเภทภาพยนตร์ 2 ประเภทคือ adventure และ drama โดยทั่วไปข้อมูลเกี่ยวกับประเภทของภาพยนตร์นี้จะถูกแทนด้วยค่า 0=ไม่มีคุณลักษณะ และ 1=มีคุณลักษณะ ถ้าสังเกตให้ดีจะพบว่าใน UIR และ IAB จะมีข้อมูลเกี่ยวกับ id ของภาพยนตร์เหมือนกัน ดังนั้น เราจึงสามารถนำข้อมูลจากทั้งสองตาราง มารวมเข้าด้วยกันโดยอาศัย id ของภาพยนตร์ ดังแสดงในตารางที่ 3

2. คำนวณหาค่า global criteria weights ของผู้ใช้

ขั้นตอนนี้เป็นกรนำเอาข้อมูลผลลัพธ์ที่ได้จากขั้นตอนที่ 1 มาทำการวิเคราะห์เพื่อคำนวณหาค่า global criteria weights ของผู้ใช้แต่ละคน ค่า global criteria weights ของผู้ใช้เป็นค่าที่แสดงความชอบเกี่ยวกับคุณลักษณะทั่วไป (เช่น ประเภทภาพยนตร์) ของภาพยนตร์ที่ผู้ใช้ชอบ เป็นความชอบที่ไม่ได้เฉพาะเจาะจง

ว่าจะดูภาพยนตร์เรื่องไหน ตัวอย่างเช่น ถ้าถามว่าคุณชอบดูภาพยนตร์แนวไหน เราอาจจะตอบว่าแนว action กับ adventure โดยที่ไม่ได้ระบุว่า เป็นภาพยนตร์เรื่องใด เป็นต้น เนื่องจากว่าโดยปกติถ้าคนเราชอบดูภาพยนตร์แนวไหนก็มักจะเลือกดูแนวที่ตัวเองชอบนั้นบ่อยๆ ดังนั้นเราสามารถระบุได้ว่าเราสนใจภาพยนตร์แนวไหนเป็นพิเศษ โดยพิจารณาจากค่าความถี่ในการเลือกดูภาพยนตร์แต่ละประเภคนั้นเอง ตัวอย่างเช่น จากข้อมูลของผู้ใช้คนที่ 4 ในตารางที่ 3 จะเห็นว่าเราได้เลือกชมภาพยนตร์ทั้งหมด 4 เรื่องคือ เรื่องที่ 2, 4, 5 และ 7 จากการสังเกตข้อมูลการชมภาพยนตร์ของเขา พบว่าภาพยนตร์ที่เขาเลือกชมบ่อยที่สุดจะเป็นแนว action ดังนั้นเราจึงพอสรุปในเบื้องต้นได้ว่าเขาชอบดูภาพยนตร์แนว action มากที่สุดเป็นต้น

โดยสรุป ค่า global criteria weights ของผู้ใช้สามารถคำนวณได้โดยนับจำนวนความถี่ของประเภทภาพยนตร์ที่ผู้ใช้เกี่ยวข้อง จากนั้น

จึงปรับเปลี่ยนค่าความถี่ของแต่ละประเภทของภาพยนตร์ให้อยู่ในช่วงค่าคะแนนระหว่าง 0 ถึง 1 ซึ่งสามารถเขียนเป็นสมการได้ ดังสมการที่ 8

$$g_{x,a} = h \sum_{i \in I_x} b_{i,a} \quad (8)$$

เมื่อกำหนดให้ I_x แทนเซตของภาพยนตร์ที่ถูกกำหนดค่าความชอบโดยผู้ใช้ x และให้ $b_{i,a}$ แทนค่าของแต่ละประเภทภาพยนตร์ของภาพยนตร์เรื่องที่ i โดยที่ $1 =$ เป็นภาพยนตร์ประเภทนั้น, $0 =$ ไม่ใช่ภาพยนตร์ประเภทนั้น. สำหรับตัวคูณ h เป็นค่า normalization factor ซึ่งถูกนิยามไว้ดังสมการที่ 9 โดยที่ A_i จะแทนเซตของคุณลักษณะ (ประเภทภาพยนตร์) ของภาพยนตร์เรื่องที่ i

$$h = 1 / \max_{a \in A_i} \left(\sum_{i \in I_x} b_{i,a} \right) \quad (9)$$

ตัวอย่าง การคำนวณ global criteria weigh ของผู้ใช้คนที่ 4

$$g_{u4,action} = 3/3 = 1.0, g_{u4,adventure} = 1/3 = 0.333, g_{u4,drama} = 2/3 = 0.667$$

$$g_{u4,war} = 2/3 = 0.667, g_{u4,western} = 1/3 = 0.333$$

ตารางที่ 4 แสดงค่า global criteria weights ของผู้ใช้ทั้งหมดที่ได้จากการคำนวณ ซึ่งตารางนี้ถูกเรียกว่า User-Attribute-Weights (UAW) จากตัวอย่างที่นำเสนอในข้างต้น เราสามารถอธิบายคุณลักษณะของผู้ใช้คนที่สองได้ด้วย global criteria weights ของเขา โดยสามารถเขียนในรูปของ vector ได้ ดังนี้ $g_{u4} = (1.0, 0.333, 0.667, 0, 0, 0.667, 0.333)$ ข้อมูลดังกล่าวอธิบายโดยสรุปได้ว่าเขาชอบภาพยนตร์ประเภท action มากที่สุด รองลงมาคือ drama ซึ่งสามารถวัดค่าออกมาเปรียบเทียบกับได้เป็นตัวเลขที่แสดงถึงระดับความสนใจของเขา

ตารางที่ 4 user-attribute-weights (UAW)

	Action	Advent	Drama	Horror	Sci-fi	War	Western
u1	1	0.333	0	0	0	0.333	0.667
u2	0.333	0.667	1	0	0	0.333	0
u3	0	0.5	1	0.5	0.5	0	0
u4	1	0.333	0.667	0	0	0.667	0.333
u5	0.333	0.667	1	0	0	0.333	0

3. คำนวณหาระดับความสนใจหรือค่าน้ำหนักที่ผู้ใช้แต่ละคนกำหนดให้กับแต่ละคุณลักษณะของไอเทม

ขั้นตอนนี้เป็นกรนำเอาข้อมูลที่ได้จาก ขั้นตอนที่ 1 และ 2 มาทำการวิเคราะห์เพื่อคำนวณหาระดับความสนใจหรือค่าน้ำหนักที่ผู้ใช้แต่ละคนกำหนดให้กับคุณลักษณะต่างๆ (ประเภทภาพยนตร์) ที่มีอยู่ในเนื้อหาของแต่ละไอเทม (ภาพยนตร์) ที่เกี่ยวข้อง โดยแนวคิดคือต้องการที่จะคำนวณหาความเห็นหรือความชอบของผู้ใช้แต่ละคนที่มีต่อแต่ละคุณลักษณะ (attribute rating) ของแต่ละไอเทม ในรูปของค่าคะแนน จากนั้นจึงทำการ normalize ค่าคะแนนเหล่านี้ให้อยู่ใน scale ระหว่าง 0 ถึง 1 เพื่อให้สามารถนำไปเปรียบเทียบกับค่า global criteria weights ของผู้ใช้ได้ในภายหลัง

ในขั้นต้นจะทำการสำเนาข้อมูลทั้งหมดใน UIABR ออกมาเป็นข้อมูลอีกชุดหนึ่งจากนั้นจะแทนค่าไบนารีที่เป็น 1 ในเมทริกซ์ดังกล่าวทั้งหมดด้วยค่า global criteria weights ของผู้ใช้แต่ละคนที่สอดคล้องตรงกัน โดยการนำเอาแต่ละค่าของ global criteria weights คูณกับค่าไบนารีที่ตรงกัน จากนั้นแปลงค่าที่คำนวณได้แต่ละค่าให้อยู่ในช่วง 0 ถึง 1 ด้วยการหารด้วยค่าคะแนนของประเภทภาพยนตร์ที่มีค่าสูงสุดดังแสดงในสมการที่ 10 โดยที่ $s_{x,i,a}$ แทนค่าระดับความสนใจ (weight) ของผู้ใช้ x ที่มีต่อคุณลักษณะ a ของไอเทมที่ i

$$S_{x,i,a} = \frac{b_{i,a} \cdot g_{x,a}}{\max_{a \in A_i}(b_{i,a} \cdot g_{x,a})} \quad (10)$$

จากนั้นเราก็สามารถคำนวณค่าความชอบของผู้ใช้ x ที่มีต่อคุณลักษณะ a ของไอเทมที่ i ด้วยการนำค่าที่ได้จากการคำนวณด้วยสมการที่ 10 มาคูณกับค่าความชอบโดยรวมที่ผู้ใช้งานหนดให้กับไอเทมนั้น ดังแสดงในสมการที่ 11 โดยที่ $rop_{x,i,a}$ เป็นค่าความชอบที่ผู้ใช้ x กำหนดให้กับคุณลักษณะ a ของไอเทมที่ i และ $r_{x,i}$ แทนค่าความชอบโดยรวมที่ผู้ใช้ x ที่ได้กำหนดให้กับไอเทม i

$$rop_{x,i,a} = s_{x,i,a} \cdot r_{x,i} \quad (11)$$

ตัวอย่างเช่น ผู้ใช้คนที่ 2 เลือกชมภาพยนตร์เรื่องที่ 6 ซึ่งประเภทของภาพยนตร์เป็นแนว drama และ adventure หลังจากที่เราแทนค่า binary ที่เป็น 1 ในข้อมูลของภาพยนตร์เรื่องที่ 6 ด้วยค่า global criteria weight ที่ตรงกันของผู้ใช้คนที่ 2 โดยใช้สมการที่ 10 แล้ว ก็สามารถที่จะคำนวณค่าความชอบที่ผู้ใช้คนที่ 2 จะกำหนดให้กับประเภทภาพยนตร์ต่างๆ (drama และ adventure) ซึ่งเป็นคุณลักษณะของ ภาพยนตร์เรื่องที่ 6 ได้โดยใช้สมการที่ 11 ดังนี้

$$\begin{aligned} rop_{u2,m6,drama} &= s_{u2,m6,drama} \cdot r_{u2,m6} = (1) \cdot (4) = 4 \\ rop_{u2,m6,adventure} &= s_{u2,m6,adventure} \cdot r_{u2,m6} = (0.667) \cdot (4) = 2.668 \end{aligned}$$

อย่างไรก็ตาม เพื่อให้ค่าที่คำนวณได้ดังกล่าวเป็นอิสระโดยไม่ขึ้นต่อ rating scale ของค่าความชอบที่กำหนดในระบบผู้แนะนำ เราจึงจำเป็นต้องเปลี่ยนจากค่าความชอบที่มีค่าอยู่ในช่วงระหว่าง 1 ถึง 5 ไปเป็นค่าที่อยู่ในช่วงระหว่าง 0 ถึง 1 ซึ่งสามารถทำได้ด้วยสองขั้นตอน ขั้นตอนแรก เปลี่ยนค่าความชอบโดยรวม

ให้อยู่ในช่วง 0 ถึง 1 โดยการหารค่าดังกล่าวด้วยค่าสูงสุดของ rating scale (ในที่นี้คือ 5) ดังแสดงด้วยสมการที่ 12 จากนั้นนำค่าที่ได้ซึ่งเป็นค่าน้ำหนักหรือความสำคัญโดยรวม ไปคูณกับค่าน้ำหนัก หรือระดับความสนใจของผู้ใช้ x ที่มีต่อลักษณะ a ซึ่งเป็นส่วนประกอบของ item ที่ i ที่คำนวณมาจากสมการที่ 10 ดังแสดงด้วยสมการที่ 13

$$overall\ weight = r_{x,i,o} = r_{x,i} / maximum\ rating\ scale \quad (12)$$

$$wop_{x,i,a} = s_{x,i,a} \cdot overall\ weight \quad (13)$$

โดยที่ $wop_{x,i,a}$ คือค่าระดับความสนใจหรือน้ำหนักที่ผู้ใช้ x กำหนดให้กับคุณลักษณะที่ a ของ item i

ตัวอย่างเช่น สมมติว่าในระบบผู้แนะนำภาพยนตร์กำหนดให้ใช้ rating scale อยู่ในช่วง 1 ถึง 5 ค่าระดับความสนใจหรือค่าน้ำหนักที่ผู้ใช้คนที่ 2 กำหนดให้กับแต่ละคุณลักษณะ (drama และ adventure) ของภาพยนตร์เรื่องที่ 6 สามารถคำนวณได้ดังนี้

$$wop_{u2,m6,drama} = (1) \cdot (4/5) = 0.8$$

$$wop_{u2,m6,adventure} = (0.667) \cdot (4/5) = 0.534$$

ตารางที่ 5 แสดงให้เห็นระดับความสนใจหรือน้ำหนักทั้งหมดที่ผู้ใช้แต่ละคนกำหนดให้กับคุณลักษณะต่างๆ ของภาพยนตร์แต่ละเรื่องที่เขาเคยชมแล้ว

ตารางที่ 5 user-item-attribute weights (UIAW)

		Action	Advent	Drama	Horror	Sci-fi	War	Western	Overall
u1	m1	0.800	0.266	0	0	0	0	0.534	0.8
	m5	0.400	0	0	0	0	0	0	0.4
	m7	0.600	0	0	0	0	0.200	0.400	0.6
u2	m2	0	0	0.400	0	0	0	0	0.4
	m4	0.266	0.534	0.800	0	0	0.266	0	0.8
	m6	0	0.534	0.800	0	0	0	0	0.8
u3	m2	0	0	0.600	0	0	0	0	0.6
	m3	0	0	0	0.300	0.300	0	0	0.6
	m6	0	0.200	0.400	0	0	0	0	0.4
u4	m2	0	0	0.300	0	0	0	0	0.6
	m4	0.800	0.266	0.533	0	0	0.5336	0	0.8
	m5	0.600	0	0	0	0	0	0	0.6
	m7	0.600	0	0	0	0	0.4002	0.333	0.6
u5	m2	0	0	0.600	0	0	0	0	0.6
	m4	0.200	0.4002	0.600	0	0	0.200	0	0.6
	m6	0	0.267	0.400	0	0	0	0	0.4

4. การวัดค่าระดับความเข้มข้นของแต่ละคุณลักษณะ

ขั้นตอนสุดท้ายของกรอบวิธีการรวบรวมข้อมูล คือการวัดค่าระดับความเข้มข้นของแต่ละคุณลักษณะที่ปรากฏอยู่ในเนื้อหาของไอเทม (ภาพยนตร์) เรื่องต่างๆ เนื่องจากในตัวอย่งระบบผู้แนะนำภาพยนตร์นี้ค่าเดิมที่ใช้ในการอธิบายคุณลักษณะของภาพยนตร์มีค่าเป็นตัวเลขไบนารี (0 หรือ 1) ซึ่งใช้อธิบายได้แค่เพียงการบ่งบอกว่ามีคุณลักษณะนั้นๆ อยู่ในเนื้อหาของไอเทมหรือไม่เท่านั้น ไม่สามารถบอกค่าระดับความเข้มข้นของแต่ละคุณลักษณะที่ปรากฏอยู่ในเนื้อหาของ item ที่แท้จริงได้ ดังนั้น จึงต้องอาศัยความเห็นของผู้ใช้ที่เกี่ยวข้องร่วมกันพิจารณา (collaborate) ซึ่งวิธีที่ง่ายที่สุดคือการคำนวณค่าเฉลี่ยของความเห็นต่าง ๆ ของผู้ใช้ที่เกี่ยวข้องกับคุณลักษณะนั้นๆ (คำนวณมาได้จากสมการที่ 13) ดังแสดงได้ด้วยสมการที่ 14

$$as_{i,a} = \frac{\sum_{x \in U_i} wop_{x,i,a}}{|U_i|} \quad (14)$$

โดยที่ $as_{i,a}$ แทนระดับความเข้มข้นของคุณลักษณะ a ของไอเทมที่ i และ $|U_i|$ แทน cardinality ของเซตของผู้ใช้ที่เกี่ยวข้องกับไอเทมที่ i

ตารางที่ 6 แสดงให้เห็นค่าต่างๆที่ได้จากการคำนวณด้วยสมการที่ 14 ซึ่งถูกใช้ในการอธิบายระดับความเข้มข้นของแต่ละประเภทภาพยนตร์ ที่ปรากฏอยู่ในเนื้อหาของภาพยนตร์แต่ละเรื่อง

ตารางที่ 6 Item-Attribute (Average) Weights (IAW)

	Action	Advent	Drama	Horror	Sci-fi	War	Western	Overall
m1	0.800	0.2664	0	0	0	0	0.5336	0.800
m2	0	0	0.55	0	0	0	0	0.550
m3	0	0	0	0.3	0.3	0	0	0.600
m4	0.422067	0.400067	0.644533	0	0	0.333267	0	0.733
m5	0.5	0	0	0	0	0	0	0.500
m6	0	0.433467	0.5333	0	0	0	0	0.533
m7	0.6	0	0	0	0	0.3	0.3666	0.600

การหาความคล้ายคลึงกันของข้อมูลโดยรวมแบบไฮบริด

ถึงแม้ว่าความคล้ายคลึงกันระหว่างผู้ใช้หรือระหว่าง item ในระบบผู้แนะนำจะสามารถวัดได้จากการคำนวณด้วยวิธีที่เป็นมาตรฐานอย่าง Cosine similarity หรือ Pearson correlation อย่างไรก็ตาม Cosine ไม่สามารถใช้งานได้เมื่อประยุกต์ใช้กับชุดข้อมูลสามชุดสุดท้าย เนื่องจากว่าบางครั้ง ผู้ใช้หรือไอเทมถูกแทนด้วย vector ที่มีขนาดเล็ก ดังนั้นการเปรียบเทียบระหว่าง vector ด้วยวิธีการดังกล่าวบางครั้งอาจได้ผลลัพธ์ที่ไม่สมเหตุสมผลนัก ตัวอย่างเช่น สมมติว่าเราต้องการเปรียบเทียบระหว่างไอเทม x และ y ซึ่งถูกแทนด้วย 2 vector ดังนี้ $x = (3,3)$ และ $y = (3,3)$ จากการคำนวณด้วย cosine vector similarity เราได้ค่าเท่ากับ 1 ซึ่งแสดงว่าทั้งสองไอเทมนั้นเหมือนกันมากที่สุด แต่เมื่อเราเปลี่ยนค่าที่แทน x และ y ใหม่เป็น $x = (3,3)$ และ $y = (2,2)$ หรือ ให้

$x = (5,5)$ และ $y = (1,1)$ ผลลัพธ์ที่ได้จากการคำนวณด้วย Cosine Vector similarity ก็ยังคงได้ 1 ในทุกกรณี ส่วน Pearson Correlation นั้นถึงแม้จะได้รับการยอมรับว่าทำงานได้ดีอย่างกว้างขวาง แต่ก็จำเป็นต้องใช้ความพยายามเพิ่มขึ้นในการตีความเนื่องจากค่าที่ได้อาจเป็นไปได้ทั้งค่าบวกและค่าลบ ทำให้ไม่สะดวกในการนำไปใช้ อีกทั้งยังไม่เหมาะกับกรณีไอเทมที่จะนำมาเปรียบเทียบกันมีจำนวนคุณลักษณะที่ใช้อธิบายไอเทม นั้นๆ ค่อนข้างน้อย ดังนั้นเพื่อลดความยุ่งยากและความผิดพลาดจากปัญหาในการใช้วิธีการวัดความคล้ายคลึงที่เป็นมาตรฐานดังกล่าว จึงประยุกต์ใช้วิธีการวัดความคล้ายคลึงแบบ Tanimoto correlation เพื่อทดสอบประสิทธิภาพของวิธีการแนะนำที่ได้นำเสนอเพื่อเปรียบเทียบระหว่างผู้ใช้กับไอเทมดังแสดงในสมการที่ 15

$$sim(x,i) = \frac{\sum_{a \in G_x \cap AS_i} (g_{x,a}) \cdot (as_{i,a})}{\sum_{a \in G_x \cap AS_i} (g_{x,a})^2 + \sum_{a \in G_x \cap AS_i} (as_{i,a})^2 - \sum_{a \in G_x \cap AS_i} (g_{x,a}) \cdot (as_{i,a})} \quad (15)$$

โดยที่ as_i เป็น เซตของคุณลักษณะต่างๆ (ถูกอธิบายด้วยค่าระดับความเข้มข้นของคุณลักษณะนั้นๆ ซึ่งแทนด้วย $as_{i,a}$ ของไอเทม i ตัวอย่าง เช่น ต้องการเปรียบเทียบระหว่างผู้ใช้กับไอเทมคือ u4 กับ m6 เนื่องจาก m6 มีสองคุณลักษณะ คือ Adventure และ Drama ดังนั้น จึงนำเอาเฉพาะค่า global criteria ของ

u4 ออกมาเฉพาะAdventure และ Drama เท่านั้น คือ $\langle 0.333, 0.667 \rangle$ เมื่อทำการ normalize ด้วยค่า max $(0.333, 0.667)$ จะได้ $\langle 0.333/0.667, 0.667/0.667 \rangle = \langle 0.49925, 1 \rangle$ ซึ่งสามารถนำไปเปรียบเทียบกับ m6 ได้ $u4 = \langle 0.49925, 1 \rangle$ และ $m6 = \langle 0.433467, 0.53333 \rangle$ เมื่อนำไปแทนค่าในสูตร tanimoto จะได้ ดังนี้

$$\text{Sim}(4,6) = \frac{(0.49925 * 0.433467) + (1 * 0.53333)}{(0.49925)^2 + (0.433467)^2 + (1)^2 + (0.53333)^2 - [(0.49925 * 0.433467) + (1 * 0.53333)]}$$

$$= 0.77145$$

สังเกตว่า ค่า similarity จะไม่เกิน 1 ถ้าเท่ากับ 1 แสดงว่าเหมือนกันทุกประการ เช่น สมมุติ

$$u4 = \langle 0.49925, 1 \rangle$$

$$m6 = \langle 0.49925, 1 \rangle$$

$$\text{Sim}(4,6) = \frac{(0.49925 * 0.49925) + (1 * 1)}{(0.49925)^2 + (0.49925)^2 + (1)^2 + (1)^2 - [(0.49925 * 0.49925) + (1 * 1)]}$$

$$= 1.0$$

การทดสอบและประเมินผล

ในการทดสอบความถูกต้องของวิธีการที่นำเสนอ งานวิจัยนี้ใช้ข้อมูลจากระบบแนะนำภาพยนตร์ MovieLens (<http://www.grouplens.org/node/73/>) ซึ่งข้อมูลดังกล่าวได้ถูกรวบรวมโดยกลุ่มผู้วิจัยของ GroupLens Research Project แห่งมหาวิทยาลัย Minnesota โดยฐานข้อมูล MovieLens ที่นำมาใช้ในการทดลองจะประกอบไปด้วยข้อมูลเกี่ยวกับภาพยนตร์ (movies) ที่ถูกอธิบายด้วยประเภทของภาพยนตร์ (genres) ต่าง ๆ เช่น action, adventure, drama เป็นต้น ข้อมูลค่าคะแนนความชอบโดยรวม (rating) ของผู้ใช้จำนวน 100,000 ค่าที่รวบรวมจากผู้ใช้งานจำนวน 943 คนที่ชมภาพยนตร์จำนวน 1682 เรื่อง ข้อมูลค่าคะแนนความชอบโดยรวม ที่ผู้ใช้กำหนดจะมีค่าอยู่ในช่วง 1 ถึง 5 (1=ไม่ชอบ และ 5=ชอบมากที่สุด) โดยผู้ใช้แต่ละคนได้กำหนดค่าคะแนนความชอบโดยรวมให้กับภาพยนตร์ไม่น้อยกว่า 20 เรื่อง โดยทดสอบความถูกต้องของการทำนายของวิธีที่นำเสนอเปรียบเทียบกับวิธีการแนะนำแบบอื่นๆ ทั้งแบบ single-criteria แบบดั้งเดิมคือ Single User-Based Collaborative Filtering และวิธีการแบบ multi-criteria ที่ประยุกต์ใช้เทคนิค aggregation approach (Adomavicius et al, 2005)

คือ Multi-attribute Aggregation for User-Based Collaborative Filtering การประเมินประสิทธิภาพความถูกต้องในการจำแนก (classification accuracy) จะกำหนด thresholds เพื่อใช้ในการจำแนกภาพยนตร์ที่ถูกแนะนำออกเป็นสองกลุ่ม (relevant และ non-relevant) โดยจะประเมินด้วยค่า threshold 2 ระดับคือ 3 and 4 (ถ้าค่าที่ทำนายมากกว่า หรือเท่ากับ Threshold ที่กำหนด ผลลัพธ์นั้นจะจัดอยู่ในกลุ่ม relevant, ถ้าน้อยกว่าจะจัดอยู่ในกลุ่ม non-relevant) เราเลือกที่จะทดลองในกรณีที่ขนาด neighbor size เท่ากับ 10 โดยจะพิจารณาสิ่งที่แนะนำในลักษณะ top N recommendations โดยที่ N คือจำนวนไอเทมหรือคำแนะนำที่เหมาะสมมากที่สุดที่ต้องการจะแนะนำให้กับผู้ใช้ สำหรับการทดลองนี้ได้กำหนดค่าของ N เป็น 1 และ 5 และได้ผลการทดลองดังแสดงในตารางที่ 7

ตารางที่ 7 ผลการทดสอบ

ค่า Thresholds	วิธีการ	Precision in Top-1(%)	Precision in Top-5(%)
3	Single User-Based Collaborative Filtering	89.2	89.1
	Multi-attribute Aggregation for User-Based Collaborative Filtering	90.0	89.7
	วิธีที่นำเสนอ	91.1	91.0
4	Single User-Based Collaborative Filtering	71.9	72.8
	Multi-attribute Aggregation for User-Based Collaborative Filtering	72.2	73.9
	วิธีที่นำเสนอ	74.1	75.8

จากข้อมูลที่แสดงในตารางที่ 7 แสดงให้เห็นว่าวิธีการที่นำเสนอมีผลการทำนายที่ดีกว่าวิธีการอื่นที่นำมาเปรียบเทียบทั้งสองวิธี เป็นการสนับสนุนว่าวิธีการรวบรวมข้อมูลที่นำเสนอนี้สามารถสร้างข้อมูลที่เป็นประโยชน์ต่อการพัฒนาระบบผู้แนะนำแบบหลายเกณฑ์ได้ นอกจากนั้น ในวิธีการที่นำเสนอนี้ยังเป็นการลดการ sparse ของเมทริกต์ไว้ในระบบด้วยการแปลงข้อมูลให้เป็นแบบไฮบริด เช่น ในกรณีของตัวอย่างที่ใช้ในการทดลอง ได้การแปลงขนาดของข้อมูล User-Item Ratings (UIR) ที่มีขนาดเท่ากับ $100,000 \times 1682$ (จำนวนผู้ใช้ที่กำหนดค่าความชอบโดยรวม $\times$ จำนวนภาพยนตร์) ให้เป็นเมทริกต์ขนาด 943×7 (จำนวนผู้ใช้ $\times$ จำนวนของประเภทภาพยนตร์) จะเห็นได้ว่าขนาดของข้อมูลเล็กลงอย่างมาก ในส่วนซึ่งข้อมูลของไอเทม (Item-Attributes Binary Values: IAB) ถึงแม้ว่าจะมีขนาดเท่าเดิมไม่เปลี่ยนแปลงในกรณีนี้เท่ากับ 1682×7 (จำนวนภาพยนตร์ $\times$ จำนวนของประเภทภาพยนตร์) แต่ค่าน้ำหนักคะแนนของแต่ละคุณลักษณะมีความชัดเจนมากขึ้น ไม่ใช่ค่าที่เป็น 0 หรือ 1 เท่านั้น ทำให้การทำนายมีความถูกต้องมากขึ้น

สรุป

ในงานวิจัยนี้ได้นำเสนอระบบผู้แนะนำแบบหลายเกณฑ์ ด้วยวิธีการรวบรวมข้อมูลความชอบของผู้ใช้ทางอ้อมเพื่อลดภาระของผู้ใช้ในการที่จะต้องประเมินและให้ข้อมูลความชอบที่มีต่อคุณลักษณะต่างๆ ของไอเทม และลดปัญหาความไม่สอดคล้องกันของข้อมูลที่เกิดจากการรวบรวมจากผู้ใช้อย่างตรง โดยวิธีการที่นำเสนอจะเป็นการผนวกกันของข้อมูลความชอบของผู้ใช้เกี่ยวกับคุณลักษณะโดยรวมของไอเทมเข้ากับข้อมูลคุณลักษณะต่างๆ ของไอเทม เพื่อทำการวิเคราะห์และสร้างเป็นข้อมูลความชอบของผู้ใช้เกี่ยวกับคุณลักษณะต่างๆ ของไอเทม เพื่อประโยชน์ในการแนะนำไอเทมให้กับผู้ใช้เป็นรายเฉพาะบุคคลที่เป็นประโยชน์อย่างมากต่อการพัฒนาระบบผู้แนะนำแบบหลายเกณฑ์ เป็นประโยชน์สำหรับการเพิ่มความสามารถการทำงานจากระบบที่เคยใช้วิธีการแบบเกณฑ์เดียวไปใช้วิธีการแบบหลายเกณฑ์ที่มีประสิทธิภาพมากกว่าโดยการใช้อยู่เดิมที่มีอยู่ได้แล้ว พร้อมทั้งยังสามารถที่จะลดปัญหา intrusiveness และความไม่สอดคล้องของข้อมูลความชอบของผู้ใช้ได้อีกด้วย

เอกสารอ้างอิง

- Adomavicius, Gediminas, and Youngok Kwon. 2007. New Recommendation Techniques for Multicriteria Rating Systems. **IEEE Intelligent Systems** 22, no. 3: 48-55.
- Adomavicius and Tuzhilin. 2005. Toward the Next Generation of Recommender Systems: **A Survey of the. Knowledge and Data Engineering IEEE Transactions on** 17, no. 6: 734--749.
- Balabanovic, Marko, and Yoav Shoham. 1997. Fab : content-based , collaborative recommendation .(Special Section : Recommender Systems). **Communications of the ACM** 66, March: 1-7.
- Billsus, D., C. A. Brunk, C. Evans, B. Gladish, and M. Pazzani. Adaptive interfaces for ubiquitous web access. **Commnications of the ACM**, 45 no.5:34-38, 2002.
- Bollen, Dirk, Bart P Knijnenburg, Martijn C Willemsen, and Mark Graus. 2010. Understanding choice overload in recommender systems. **ACM International Conference on Recommender Systems RecSys**, 63-70.
- Burke, Robin. 2000. Knowledge-based recommender systems. **Encyclopedia of Library and Information Systems** 69, no. Supplement 32: 175--186.
- Burke, Robin. 2002. Hybrid Recommender Systems: **Survey and Experiments. User Modeling and User Adapted Interaction** 12, no. 4: 331-370.
- Burke, Robin. 2007. Hybrid Recommender Systems: The Adaptive Web. **Springer Berlin Heidelberg**: 377-408.
- Deshpande, Mukund, and George Karypis. 2004. Item-Based Top-N Recommendation Algorithms. **ACM Transactions on Information Systems** 22, no. 1: 143-177.
- Herlocker, J L, J A Konstan, A Borchers, and J Riedl. 1999. An algorithmic framework for performing collaborative filtering. **Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval**, 54: 230-237.
- Kellar, M, C Watters, J Duffy, and M Shepherd. 2004. Effect of Task on Time Spent Reading as an Implicit Measure of Interest. **Proc ASIS AM** 04.

- Konstan, J A, B N Miller, David Maltz, J L Herlocker, L R Gordon, and John Riedl. 1997. GroupLens: applying collaborative filtering to Usenet news. **Communications of the ACM 40, no. 3**: 77-87.
- Lee, T, and Y Park. 2008. A time-based approach to effective recommender systems using implicit feedback. **Expert Systems with Applications 34, no. 4**: 3055-3062.
- Lee, Wei-Po, Chih-Hung Liu, and Cheng-Che Lu. 2002. Intelligent agent-based systems for personalized recommendations in Internet commerce. **Expert Systems with Applications 22, no. 4**: 275-284.
- Linden, G., B. Smith, and J. York. Amazon.com Recommendations: Item-to-Item Collaborative Filtering. **IEEE Internet Computing, Jan.-Feb, 2003**.
- Manouselis, Nikos, and Constantina Costopoulou. 2007. Analysis and Classification of Multi-Criteria Recommender Systems. **World Wide Web Internet and Web Information Systems 10, no. 4**: 415-441.
- Manouselis, Nikos. 2007. Experimental Analysis of Design Choices in Multiattribute Utility Collaborative Filtering. **International Journal of Pattern Recognition and Artificial Intelligence 21, no. 2**: 311.
- Middleton, Stuart E, N R Shadbolt, and D C De Roure. 2004. Ontological User Profiling in Recommender Systems. **ACM Transactions on Information Systems 22, no.1**: 54-88.
- Miller, Bradley N, Istvan Albert, Shyong K Lam, Joseph A Konstan, and John Riedl. 2003. MovieLens unplugged: experiences with an occasionally connected recommender system. **Proceedings of the 8th international conference on Intelligent user interfaces**: 263-266.
- Marlin, B. Modeling User Rating Profiles for Collaborative Filtering. **Proceedings of the 17th Annual Conference on Neural Information Processing Systems (NIPS'03), 2003**.
- MovieLens, Retrieved January 29, 2012 from <http://www.grouplens.org/node/73/>.
- Oard, D., & Kim, J. (1998). Implicit Feedback for Recommender Systems. **Proceedings of the AAAI Workshop on Recommender Systems**: 81-83.
- O'Donovan, John, and Barry Smyth. 2005. Trust in recommender systems. **Proceedings of the 10th international conference on Intelligent user interfaces IUI 05 15, June**: 167.

- Palanivel K. and Sivakumar R. 2010. A Study On Implicit Feedback in Multicriteria E-Commerce Recommender System. **Journal of Electronic Commerce Research 11, no. 2:** 140-156.
- Pazzani, Michael J. 1999. A Framework for Collaborative, Content-Based and Demographic Filtering. **Artificial Intelligence Review 13, no. 5:** 393-408.
- Pittaya Poompiang, Wichian Premchaiswadi. 2010. User and Item Pattern Matching in Multi-criteria Recommender Systems. **11th ACIS International Conference on Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing, SNPD 2010, London, England, UK, June 9-11, 2010:** 20-25.
- Recker, Mimi M, and Andrew Walker. 2003. Supporting “word-of-mouth” social networks through collaborative information filtering. **Journal Of Interactive Learning Research 14, no. 1.**
- Ricci, F, and H Werthner. 2002. Case base querying for travel planning recommendation. **Tourism 3, no. 4:** 215-226.
- Rosset, S., E. Neumann, U. Eick, N. Vatnik, and Y. Idan. Customer Lifetime Value Modeling and Its Use for Customer Retention Planning. **Proceeding of the 8th ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining (KDD-2002), July 2002.**
- Schafer, J Ben, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative Filtering Recommender Systems. **International Journal of Electronic Business 2, no. 1:** 77.
- Schafer, J Ben, Joseph A Konstan, and John Riedl. 2001. E-Commerce Recommendation Applications. **Data Mining and Knowledge Discovery 5, no. 1:** 115-153.
- Schafer, J Ben, Joseph Konstan, and John Riedi. 1999. Recommender systems in e-commerce. **Proceedings of the 1st ACM conference on Electronic commerce EC 99:** 158-166.
- Takács, Gábor, István Pilászy, Botyán Németh, and Domonkos Tikk. 2008. Matrix factorization and neighbor based algorithms for the netflix prize problem. **Proceedings of the 2008 ACM conference on Recommender systems RecSys 08:** 267.