โครงงาน

เรื่อง Greed Route ระบบแนะนำหนังสือออนไลน์

Book Recommendation System for Greed Route Platform

เสนอ

รองศาสตราจารย์ ดร.วรสิทธิ์ ชูชัยวัฒนะ

จัดทำโดย

จิตรภานุ วุฒิกรวิภาค	6780061626
คณิศร อุ่นจิตติกุล	6780051326
นนธพงศ์ ควรสวัสดิ์	6780191526
พรเพ็ญ อาชีวะเกษะ	6780258026
ณมลพรรษ วิริยะโรจน์	6780124526
ยุทธพงษ์ มหาสิทธิวัฒน์	6780330426
นลิภรณ์ คลังเปรมจิตต์	6780203426
นวพร วิริยะปรีดา	6780206326

โครงงานนี้เป็นส่วนหนึ่งของวิชาระบบแนะนำ (2603522)

หลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาสถิติและวิทยาการข้อมูล

คณะพาณิชยศาสตร์และการบัญชี

จุฬาลงกรณ์มหาวิทยาลัย

ภาคเรียนที่ 2 ปีการศึกษา 2568

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ในยุคที่ข้อมูลหนังสือมีมากมายบนแพลตฟอร์มออนไลน์ ผู้อ่านมักประสบปัญหาในการเลือกหนังสือที่ตรงกับสิ่งที่สนใจ (Information Overload) อีกทั้งการเข้าถึงเพียงคำโปรยหรือเนื้อหาบางส่วนของหนังสือยังทำให้การตัดสินใจไม่แน่นอน และอาจทำให้ต้องใช้เวลาไปกับหนังสือที่ไม่ตรงกับความชื่นชอบ ซึ่งนำไปสู่ความไม่พึงพอใจหลังการซื้อ

ปัญหานี้ไม่ได้เกิดกับผู้อ่านเพียงฝ่ายเดียว แต่ยังส่งผลต่อผู้มีส่วนได้ส่วนเสียอื่น ๆ ในระบบนิเวศของหนังสือ เช่น สำนักพิมพ์ที่ขาดข้อมูลในการวางแผนและคาดการณ์ความต้องการ และร้านหนังสือที่ประสบปัญหาในการบริหารคลังสินค้า รวมถึงความท้าทายในการเข้าถึงกลุ่มลูกค้าเป้าหมาย

โครงงานนี้นำเสนอการออกแบบระบบแนะนำหนังสือสำหรับแพลตฟอร์ม “Greed Route” ซึ่งเชื่อมต่อผู้อ่าน สำนักพิมพ์ และร้านหนังสือ (Multi-sided Platform) โดยมีเป้าหมายเพื่อมอบประสบการณ์การแนะนำแบบเฉพาะบุคคล (Personalized Recommendation) และเปิดโอกาสให้ผู้ใช้ค้นพบสิ่งใหม่ผ่านระบบ Surprise Recommendation โดยมีองค์ประกอบหลัก ได้แก่

- ระบบแนะนำส่วนบุคคล (Recommended for You) เพื่อแนะนำหนังสือตามพฤติกรรมและความชื่นชอบของผู้ใช้
- ระบบสำรวจ (Explore, New Releases, Trending) เพื่อช่วยให้ผู้ใช้ค้นพบหนังสือใหม่
- ชุมชน (Groups, Discussions, Quotes) เพื่อส่งเสริมปฏิสัมพันธ์ระหว่างผู้อ่าน
- ชั้นหนังสือส่วนตัว (My Books) สำหรับติดตามและจัดการรายการหนังสือของผู้ใช้
- Choice Awards สำหรับโหวตหนังสือยอดเยี่ยมประจำปี

ระบบแนะนำถูกออกแบบ (Recommender System) โดยใช้ข้อมูลพฤติกรรมผู้ใช้ (User Interaction Events) ร่วมกับข้อมูลหนังสือ (Metadata) และข้อมูลรีวิว (Review Messages) เพื่อพัฒนาระบบที่สามารถแนะนำหนังสือได้อย่างมีประสิทธิภาพและสอดคล้องกับความต้องการของผู้ใช้ โดยใช้แนวทางแบบผสม (Hybrid Approach) ซึ่งผสานหลายเทคนิคเข้าด้วยกัน ได้แก่ Content-Based Filtering, Collaborative Filtering, Commercial Strategies และ Surprise Recommendation นอกจากนี้โครงงานนี้ยังมีการจำลองข้อมูล (Simulated Data) ด้วย gemini-2.5-flash-lite เพื่อสะท้อนภาพรวมและสถานการณ์การใช้งานจริงของระบบ

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อออกแบบระบบแนะนำหนังสือเพื่อช่วยให้ผู้ใช้งานสามารถค้นหาหนังสือที่สอดคล้องกับความสนใจและพฤติกรรมการอ่านของตนเองได้ง่ายขึ้น ลดปัญหาในการตัดสินใจเลือกหนังสือ และเพิ่มความพึงพอใจในการใช้งาน
2. เพื่อศึกษาและเปรียบเทียบโมเดลการแนะนำหนังสือ ได้แก่ Content-based, Collaborative Filtering และ Hybrid Model รวมถึงประยุกต์ใช้กลยุทธ์เชิงพาณิชย์ (Commercial strategies)
3. เพื่อออกแบบและประเมินคุณภาพของระบบทั้งแบบ Offline และ Online พร้อมทั้งเปรียบเทียบผลลัพธ์เพื่อเลือกโมเดลที่เหมาะสมที่สุด โดยใช้ตัวชี้วัด เช่น Hit Rate@K, NDCG@K, Click-through Rate (CTR) และ Conversion
4. พัฒนาแพลตฟอร์มต้นแบบ (Prototype Platform) ในรูปแบบเว็บไซต์ สำหรับให้บริการ

1.3 ขอบเขตของโครงงาน

โครงงานนี้มุ่งเน้นการออกแบบและพัฒนาระบบแนะนำหนังสือในระดับต้นแบบ โดยเน้นการกำหนดโครงสร้างระบบ (Blueprint) และแนวทางการประยุกต์ใช้โมเดลที่มีอยู่ ทั้งนี้ มีการพัฒนาเว็บไซต์ต้นแบบเพื่อสาธิตการทำงานของระบบ และรองรับการแสดงผลรายการแนะนำในลักษณะการใช้งานจริง

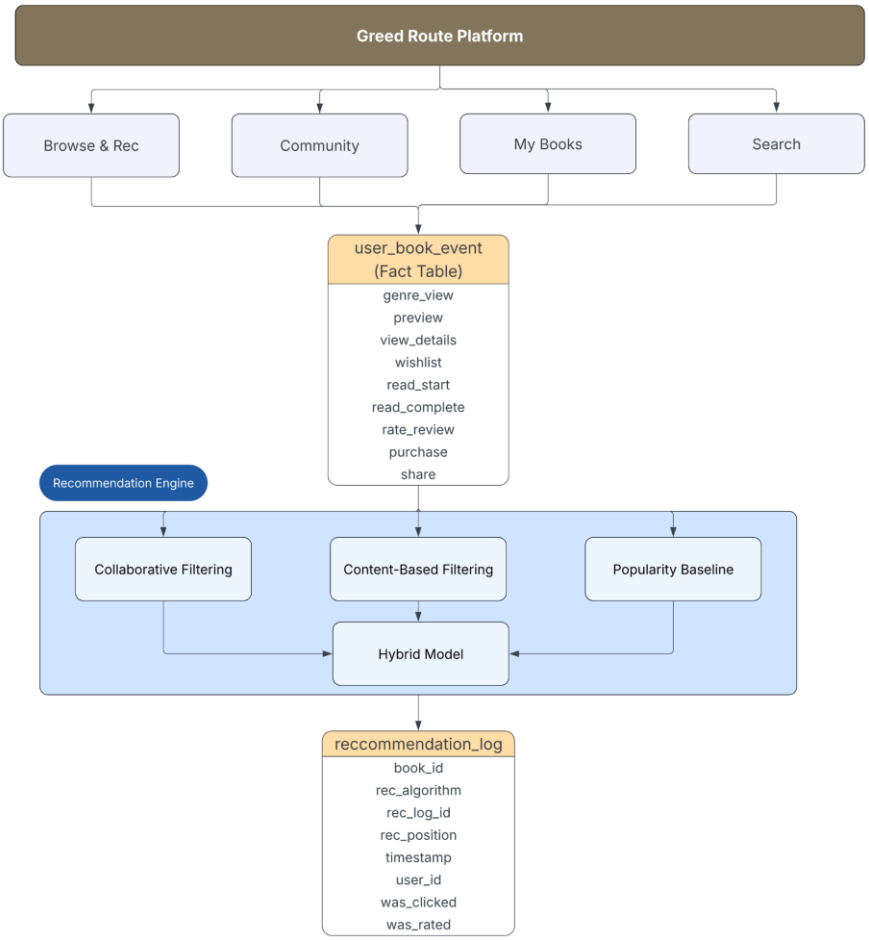
ข้อมูลที่ใช้ในโครงงานนี้ถูกจำลองขึ้นเพื่อสะท้อนพฤติกรรมการใช้งานจริงของผู้ใช้ โดยมีการสร้างข้อมูลในรูปแบบเหตุการณ์ (Event Transaction) ที่ครอบคลุมพฤติกรรมต่าง ๆ เช่น การคลิกดูรายละเอียด การเริ่มอ่าน การให้คะแนน และการซื้อหนังสือ เพื่อให้สามารถนำไปใช้ในการพัฒนาและประเมินระบบแนะนำได้อย่างมีประสิทธิภาพ

บทที่ 2
ลักษณะข้อมูล

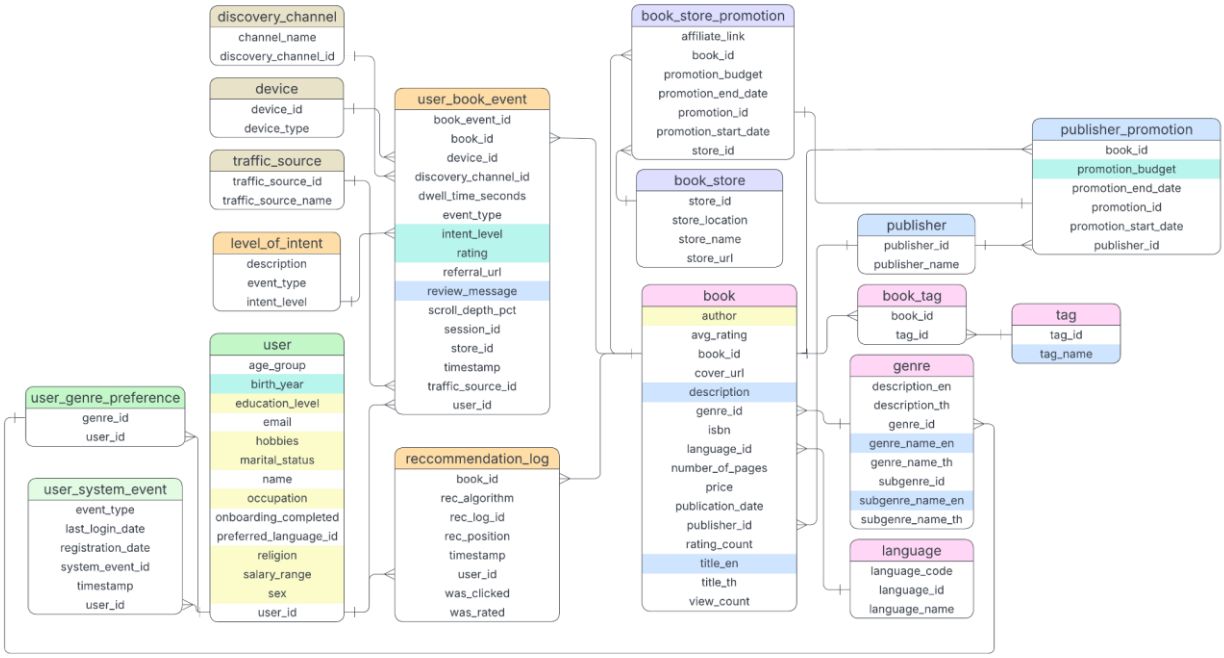
บทนี้นำเสนอภาพรวมสถาปัตยกรรมระบบแนะนำและอธิบายโครงสร้างและความสัมพันธ์ของข้อมูล รวมถึงกระบวนการจัดการและเตรียมข้อมูลเพื่อใช้ในการพัฒนาโมเดล

2.1 ภาพรวมสถาปัตยกรรมระบบ

ภาพรวมของระบบ Greed Route รองรับการใช้งานหลัก ได้แก่ การค้นหา การสำรวจ การแนะนำหนังสือ ชุมชน และชั้นหนังสือส่วนตัว โดยข้อมูลพฤติกรรมของผู้ใช้จะถูกบันทึกในรูปแบบเหตุการณ์ (Event Transaction) เพื่อนำไปใช้ในระบบแนะนำหนังสือ และมีการจัดเก็บผลลัพธ์เพื่อใช้ในการประเมินและพัฒนาปรับปรุงระบบต่อไป



2.2 การออกแบบโครงสร้างฐานข้อมูล



ฐานข้อมูลออกแบบตาม Star Schema เพื่อให้รองรับระบบแนะนำ โดยมีตาราง user_book_event เป็นจุดศูนย์กลาง ล้อมรอบด้วย 17 ตาราง แบ่งเป็น 4 กลุ่มตามหน้าที่ในระบบ

1. **Fact / Event Tables:** ประกอบด้วยตารางที่ใช้เก็บข้อมูลพฤติกรรมของผู้ใช้และผลลัพธ์จากระบบแนะนำ ได้แก่
    - user_book_event: ข้อมูลการใช้งานของผู้ใช้กับหนังสือ
    - recommendation_log: ข้อมูลการแนะนำ เช่น รายการที่แสดง ลำดับ และการคลิก
2. **Core Entity Tables:** ประกอบไปด้วยตารางที่ใช้เก็บข้อมูลซึ่งถูกอ้างอิงจากตาราง fact ได้แก่
    - user: ข้อมูลผู้ใช้
    - book: ข้อมูล metadata ของหนังสือ
    - genre: ข้อมูลประเภทและหมวดหมู่ของหนังสือ
    - publisher: ข้อมูลสำนักพิมพ์
3. **Dimension Tables:** ประกอบไปด้วยตารางอ้างอิงความหมายจาก fact table เช่น language, device, level_of_intent, discovery_channel
4. **Bridge / Junction Tables:** ประกอบไปด้วยตารางที่ primary key จาก parent table หนึ่งรายการสามารถปรากฏได้หลาย record ใน junction table ตัวอย่าง เช่น
    - user_genre_preference: ข้อมูลความชอบ genre ของผู้ใช้
    - book_tag: tag เนื้อหาของหนังสือ

- publisher_promotion: ข้อมูลโปรโมชันของสำนักพิมพ์

แหล่งข้อมูลและเอกสารประกอบ

- https://www.kaggle.com/datasets/khalam/book-recommendation-system/data
- https://khala1391.github.io/100_FinalProject_RecSys/document/index.html

ตาราง **user_book_event** ใช้บันทึกพฤติกรรมของผู้ใช้ที่เกิดขึ้นกับหนังสือ เช่น การคลิกดูรายละเอียด การเริ่มอ่าน การให้คะแนน และการซื้อหนังสือ เพื่อสะท้อนระดับความต้องการของผู้ใช้จากพฤติกรรมการใช้งาน (Level of Intent) ซึ่งฟีเจอร์นี้ช่วยให้ Greed Route เข้าใจความสนใจของผู้ใช้ได้ละเอียดกว่าการใช้เพียง rating เนื่องจากครอบคลุมพฤติกรรมที่หลากหลาย รวมถึงกรณีที่ผู้ใช้ไม่ได้ให้คะแนนโดยตรง โดยระบบแบ่ง Level of Intent ออกเป็น 9 ระดับ เรียงจากความต้องการสูงไปต่ำ และสามารถนำไปใช้เป็น implicit feedback สำหรับพัฒนาระบบแนะนำได้

ระดับความต้องการของผู้ใช้จากพฤติกรรมการใช้งาน (Level of Intent)

Intent Level	Event Type	**ตัวอย่างบน Greed Route**
0	*genre_view*	ผู้ใช้เลื่อนดูหมวด “แฟนตาซี”
1	*preview*	hover เห็นปกหนังสือ
2	*view_details*	คลิกดูรายละเอียดหนังสือ
3	*add_to_wishlist*	กดบันทึกเข้า “อยากอ่าน”
4	*purchase*	ซื้อหนังสือผ่านร้านค้าที่ลิงก์
5	*read_start*	เริ่มอ่านหนังสือ
6	*read_complete*	อ่านจบ
7	*rate_review*	ให้คะแนน + เขียนรีวิว
8	*share*	แชร์หนังสือให้เพื่อน

2.3 การเตรียมข้อมูลและการสร้างตัวแปร

ก่อนนำข้อมูลเข้าสู่แต่ละโมเดล ต้องแปลง raw data ให้เป็น feature vectors ที่เหมาะสมกับการคำนวณ similarity ระบบ Greed Route ออกแบบ pipeline ที่รองรับข้อมูล 3 ประเภทหลัก ได้แก่ Nominal, Ordinal และ Text แต่ละประเภทมีวิธีการแทนค่าที่แตกต่างกันก่อนนำไปใช้คำนวณ similarity

2.3.1 Nominal Feature Preparation

ในการเตรียม nominal features ซึ่งเป็น feature เชิงหมวดหมู่ที่ไม่มีลำดับ ก่อนนำไปเป็น input คำนวณหา similarity ในโมเดล ระบบใช้วิธี encoding หลัก 2 รูปแบบ ได้แก่ one-hot encoding และ multi-hot encoding โดยเลือกใช้ตามลักษณะย่อยของ feature ดังนี้

- **One-hot encoding** ใช้กับ feature ที่มีค่าเดียวต่อหนึ่งรายการ (single-valued feature) จะใช้ เช่น user.marital_status, user.occupation, user.religion, user.sex, user.education_level, user.salary_range
- **Multi-hot encoding** ใช้กับ feature ที่สามารถมีได้หลายค่าต่อหนึ่งรายการ (multi-valued feature) จะใช้ เพื่อให้สามารถแทนหลายหมวดหมู่ได้พร้อมกัน ได้แก่ user.hobbies, book.author, book.genre และ book.subgenre สำหรับฟีเจอร์ที่เก็บในรูปแบบ list จำเป็นต้องแยกค่าออกจากกัน (list explosion) ก่อนทำ multi-hot encoding เพื่อให้แต่ละหมวดหมู่ถูกแทนเป็นคอลัมน์แยกกัน

2.3.2 Ordinal Feature Preparation

ในการเตรียม ordinal features ซึ่งเป็น feature ที่มีลำดับหรือมีค่าเชิงตัวเลข ระบบใช้วิธี normalization แบบต่าง ๆ ตามลักษณะของ feature:

- **Min-Max Normalization [0,1]:** ใช้กับ promotion_budget เพื่อปรับค่าให้อยู่ในช่วง 0–1 หลังจากคัดเลือกหนังสือ Top 100 จากโมเดล Content-based และ Collaborative Filtering แล้ว เพื่อทำให้สามารถใช้เป็น commercial score และคำนวณร่วมกับคะแนนจากโมเดลอื่นได้ โดยคำนวณจาก (x – min) / (max – min)
- **User-Mean Normalization:** ใช้กับ user_book_event.rating และ intent_level เพื่อลดอคติรายบุคคลของผู้ใช้ โดยเฉพาะผู้ใช้ที่มีแนวโน้มให้คะแนนสูงหรือต่ำกว่าปกติ ก่อนคำนวณ similarity ค่า rating จะถูกปรับจาก r_{u,i} เป็น r_{u,i} – r̄_u เพื่อให้ similarity สะท้อนความแตกต่างเชิงสัมพันธ์มากกว่าค่า rating แบบ raw
- **Mean Normalization:** ใช้กับ user.birth_year เพื่อแปลงค่าให้อยู่ในรูปแบบเชิงสัมพันธ์ และช่วยให้สามารถเปรียบเทียบความแตกต่างระหว่างผู้ใช้ได้ง่ายขึ้น
  โดยคำนวณจาก x – mean(x)

2.3.3 Text Features — TF-IDF Weighted Keyword Embedding

ข้อมูลข้อความจาก Review Messages และ metadata ของหนังสือ เป็นแหล่งข้อมูลสำคัญที่สะท้อนทั้งความหมายเชิงเนื้อหา (semantic) และความรู้สึกของผู้ใช้ ระบบจึงแปลงข้อความเหล่านี้ให้อยู่ในรูป semantic vector ขนาด 768 มิติ โดยใช้ TF-IDF Weighted Keyword Embedding

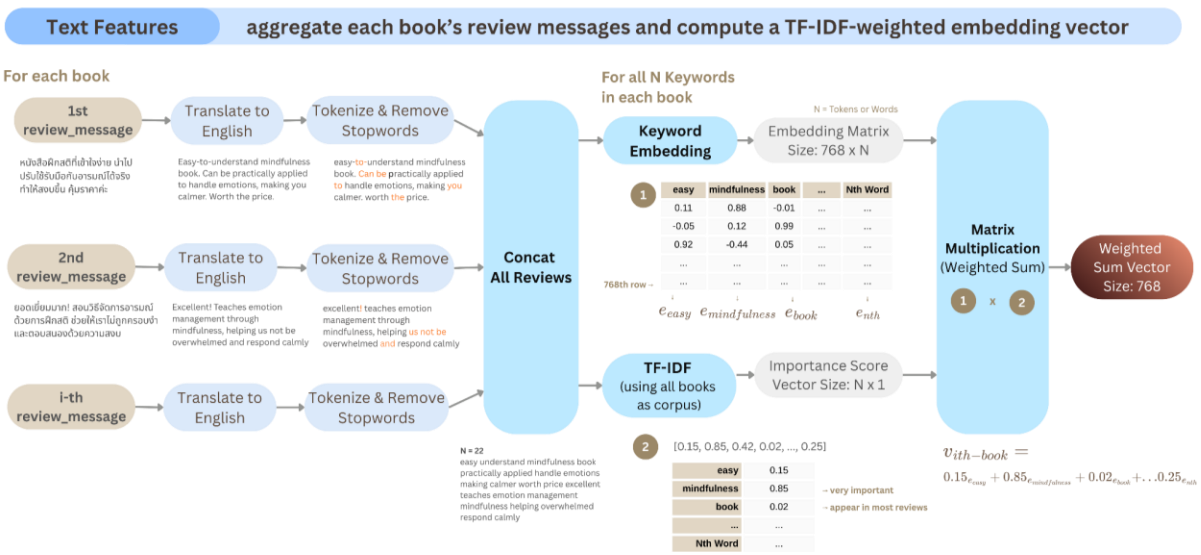
2.3.3.1 Review Messages Preparation

สำหรับรีวิวของแต่ละหนังสือ ระบบจะรวมข้อความรีวิวทั้งหมดของหนังสือแต่ละเล่มเป็นเวกเตอร์เดียว vector (768 มิติ) ผ่านขั้นตอน ดังนี้

1. **Translation:** แปลรีวิวจากภาษาไทยเป็นภาษาอังกฤษ เพื่อให้สามารถใช้ pretrained English embedding ได้อย่างมีประสิทธิภาพ
2. **Tokenization & Stopword Removal:** ตัดคำและลบ stopwords เพื่อลดสัญญาณรบกวน
3. **Review Aggregation:** รวมรีวิวทั้งหมดของหนังสือโดยการ concatenation เดียวกันเป็น document เดียว
4. **Keyword Embedding:** แปลงคำแต่ละคำเป็น embedding vector และสร้าง embedding matrix ขนาด 768 × N (N คือจำนวนคำ)
5. **TF-IDF Weighting:** คำนวณ TF-IDF โดยใช้หนังสือทุกเล่มเป็น corpus เพื่อสร้าง importance score vector ขนาด N × 1
6. **Weighted Sum:** คำนวณ weighted sum ระหว่าง embedding matrix และ TF-IDF vector ได้ vector ขนาด 768 × 1 เป็นตัวแทนเชิงความหมายของหนังสือ

vector ขนาด 768 × 1 ที่ได้เป็นตัวแทนเชิงความหมายของหนังสือ โดยรวมทั้งความหมายเชิง semantic และความสำคัญเชิงสถิติของคำ

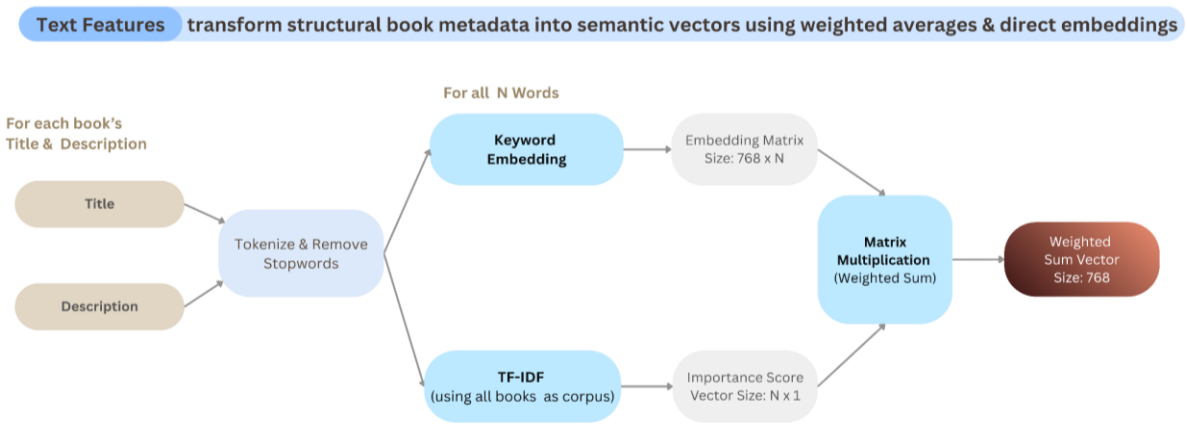
Review Messages Preparation



2.3.3.2 Book Metadata (Title & Description) Preparation

สำหรับข้อมูล metadata ของหนังสือ ได้แก่ title และ description ระบบใช้กระบวนการเดียวกัน โดยนำข้อความทั้งสองส่วนมารวมกันก่อนผ่านขั้นตอน tokenization และ embedding pipeline โดยผลลัพธ์คือเวกเตอร์ขนาด 768 มิติต่อหนังสือ ซึ่งสามารถนำไปใช้คำนวณ similarity เช่น cosine similarity ได้โดยตรงในโมเดล

Book Metadata (Title & Description) Preparation

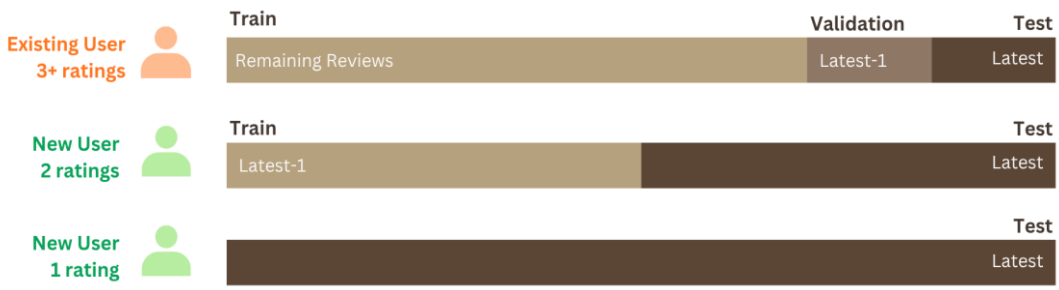


2.4 Data Splitting

เนื่องจากข้อมูล rating ของผู้ใช้มีลักษณะเป็นลำดับตามเวลา ระบบจึงใช้การแบ่งข้อมูลแบบ temporal split แทนการสุ่ม เพื่อจำลองสถานการณ์จริงที่ใช้พฤติกรรมในอดีตทำนายพฤติกรรมถัดไปของผู้ใช้ พร้อมทั้งแบ่งผู้ใช้ตามจำนวน rating วิธีการนี้ช่วยให้การประเมินสะท้อนการทำงานของระบบได้ใกล้เคียงความเป็นจริงมากขึ้น และลดปัญหา data leakage จากการใช้ข้อมูลในอนาคตในการฝึกโมเดล รายละเอียดดังนี้

- **Existing users (3+ ratings):** ใช้รายการล่าสุด (latest) เป็น test set, รายการรองล่าสุด (latest-1) เป็น validation set และที่เหลือเป็น train
- **New users (2 ratings):** ใช้รายการล่าสุด (latest) เป็น test set และ ใช้ rating ก่อนหน้าเป็น train set
- **New users (1 rating):** ใช้เป็น test set เท่านั้น เนื่องจากข้อมูลไม่พอสำหรับการ train หรือ weight tuning

Temporal Data Splitting



บทที่ 3

Model

บทนี้นำเสนอภาพรวมของโมเดล รายละเอียดของโมเดลย่อยทั้ง 8 โมเดล และกระบวนการรวมโมเดลทั้งหมดเพื่อสร้าง Hybrid Recommender System

3.1 ภาพรวมของโมเดล

ระบบแนะนำของ Greed Route แบ่งเป็นการ Recommendation เป็น 2 ประเภท ดังนี้

- **Personalized Recommendation** ใช้ Hybrid Recommender ที่ผสม 4 แนวทางหลักเพื่อรองรับทั้งผู้ใช้ใหม่และผู้ใช้เดิม พร้อมทั้งสนับสนุน commercial recommendation และ serendipity ซึ่งระบบประกอบไปด้วย 8 โมเดลที่ทำงานคู่ขนานกันและรวมคะแนนโดย Hybrid Ensemble โดยแบ่งเป็นโมเดลที่คิดคะแนน 7 โมเดล และอีก 1 โมเดล เป็น surprise recommendation ซึ่งใช้การสุ่มแบบมีหลักการ เพื่อเพิ่มความหลากหลายของคำแนะนำ
- **Non-Personalized Recommendation** ใช้โมเดล Trending สำหรับแนะนำหนังสือยอดนิยมโดยไม่อิงข้อมูล user profile

แนวทางหลักสำหรับ Recommendation

แนวทาง	วิธีการ	จุดแข็ง	จุดอ่อน
Content-Based Filtering	ใช้ features ของ item (genre, tags, description) และ user (education, hobby etc.) เปรียบเทียบกับสิ่งที่ผู้ใช้ชอบและค้นหาผู้ใช้งานที่เหมือนกัน	ทำงานได้กับ user ใหม่ และหนังสือใหม่ที่ไม่มี rating	จำกัดอยู่ใน content ที่คล้ายของเดิม ขาดการค้นพบสิ่งใหม่
Collaborative Filtering	ใช้ rating/behavior patterns ของผู้ใช้หรือ item ที่คล้ายกัน	ค้นพบ item ใหม่ที่ไม่เคยเห็น	Cold-start problem
Hybrid	รวม Collaborative + Content-Based + Commercial ผ่าน weighted ensemble	ลดจุดอ่อนของแต่ละแนวทาง ครอบคลุมหลายสถานการณ์	ซับซ้อน ต้องมีการปรับน้ำหนักและคำนึงถึงพารามิเตอร์หลายค่า
Non-Personalized / Popularity	แนะนำรายการยอดนิยมหรือรายการใหม่โดยไม่อิงกับ user profile	ทำงานได้ทันทีแม้ไม่มีประวัติผู้ใช้	ไม่ได้แนะนำรายการเฉพาะสำหรับ user

3.2 Content-Based Filtering

Content-Based Filtering จะคำนวณหา similarity ระหว่าง item หรือ user จาก feature vectors แล้วทำการแนะนำ item ที่มี features profile คล้ายกับที่ user เคยแสดงความสนใจ ในระบบ Greed Route วิธีนี้ถูกนำไปใช้ใน 3 มุมมอง ได้แก่

- Content-Based by Demographic (Model 1) ใช้ feature ของผู้ใช้ เช่น birth_year, education_level, hobbies, marital_status, occupation, religion, salary_range, sex คำนวณ similarity ระหว่างผู้ใช้เพื่อแนะนำหนังสือที่ผู้ใช้คล้ายกับที่ผู้ใช้ชอบ
- Content-Based by Metadata (Model 2) ใช้ feature ของหนังสือ เช่น genre, subgenre, title, author, description คำนวณ similarity ระหว่างหนังสือ เพื่อแนะนำเล่มที่คล้ายกับเล่มที่ผู้ใช้ชอบ
- Content-Based by behaviour data (Model 3) ใช้ feature คะแนนความสนใจ (IE) ที่ได้จาก intent level และการถ่วงน้ำหนักตามเวลา (time decay)

3.2.1 Similarity Measure: Cosine Similarity

ใช้ Cosine Similarity เป็นตัววัดความคล้ายระหว่างเวกเตอร์ โดยจะไม่ขึ้นกับขนาดของเวกเตอร์ โดยผลลัพทธ์ออกมา จะอยู่ในช่วง [−1, 1] เมื่อเวกเตอร์ชี้ไปในทิศทางเดียวกันค่า similarity จะเข้าใกล้ 1 และเมื่อชี้ตรงข้ามจะเข้าใกล้ -1 ซึ่งเหมาะกับ feature vectors ที่ผสมกันทั้ง one-hot, ordinal และ embedding ซึ่งมี scale ต่างกันมาก

$$sim(A,B) = \frac{(A \cdot B)}{(\|A\| \cdot \|B\|)}$$

โดย

- $sim(A,B)$ คือ ค่า Cosine Similarity ระหว่างหนังสือ A และหนังสือ B

3.2.1 Prediction Formula

Predicted Score ของหนังสือ i สำหรับแต่ละผู้ใช้ u สามารถคำนวณเป็น weighted average ของ similarity กับหนังสือที่ผู้ใช้เคยให้ rating ไว้ ได้เป็นสูตรดังนี้

$$score(u,i) = \frac{\Sigma\, sim(i,j) \times r(u,j)}{\Sigma\, sim(i,j)}$$

โดย

- $score(u,i)$ คือ $Predicted\ Score$ ของหนังสือ i สำหรับแต่ละผู้ใช้ u
- $sim(i,j)$ คือ ค่า $similarity$ ระหว่างหนังสือ i และหนังสือ j
- $r(u,j)$ คือ $rating$ ของหนังสือ j ที่ผู้ใช้ u เคยให้คะแนนไว้

จากสูตรนี้ จะทำให้หนังสือที่ผู้ใช้ชื่นชอบมากกว่า มีน้ำหนักในการดึง item ใกล้เคียงมากกว่า

3.3 Collaborative Filtering (CF)

Collaborative Filtering ใช้ข้อมูลของผู้ใช้หลายคนในการทำนายความชอบของผู้ใช้เป้าหมาย โดยไม่จำเป็นต้องรู้ content ของ item โดยตรง โดยแบ่งเป็น 2 รูปแบบตาม dimension ของ similarity matrix

3.3.1 รูปแบบที่ 1: Item-Based Collaborative Filtering

คำนวณความคล้ายระหว่าง items จาก rating vectors (จากข้อมูล rating, event และข้อมูลอื่นๆที่ embedding เรียบร้อยแล้ว) แล้วแนะนำ items ที่คล้ายกับสิ่งที่ผู้ใช้เคย Interact โดยสูตรการคำนวณ prediction score เป็นดังนี้

$$Pred(u,i) = \tilde{r}_{\mathrm{u}} \frac{\Sigma_{\mathrm{j}} \in S_{\mathrm{i}}\, sim(i,j) \times r_{u,j}}{\Sigma_{\mathrm{j}} \in S_{\mathrm{i}}\, sim(i,j)}$$

โดย

- $\tilde{r}_{\mathrm{u}}$ คือค่า rating เฉลี่ยของผู้ใช้ u โดยผ่านการ normalization เพื่อลด bias
- S_{i} คือเซ็ตของหนังสือใน neighborhood ที่ผู้ใช้เคยให้ rating

ในระบบ Greed Route นี้ Collaborative Filtering จะถูกใช้ใน 2 โมเดล ได้แก่ Model 4 และ 5 โดยแตกต่างกันที่ feature vector ที่ใช้คำนวณ similarity:

- Model 4: ใช้ keyword embedding จากรีวิวของผู้ใช้งานระบบ โดยใช้ TF-IDF และทำการ weighted average ซึ่งจะมีขนาด 768 dimensions
- Model 5: ใช้ user-mean normalized rating vector

3.3.2 รูปแบบที่ 2: User-Based Collaborative Filtering

ในรูปแบบนี้ระบบจะคำนวณและหาผู้ใช้ที่มี rating pattern คล้ายกับผู้ใช้เป้าหมาย โดยคำนวณค่า similarity ระหว่างผู้ใช้จาก rating vectors แล้วแนะนำหนังสือที่ผู้ใช้ที่มีลักษณะคล้ายกันชอบ แต่ผู้ใช้เป้าหมายยังไม่เคยอ่าน โดยรูปแบบนี้จะถูกใช้ใน Model 6

Similarity measure: Cosine Similarity

$$sim(u,v) = \frac{\Sigma_i \in I_{uv}\, r_{ui} \cdot r_{vi}}{\sqrt{\Sigma_i \in I_{uv} r_{ui}^2} \cdot \sqrt{\Sigma_i \in I_{uv}\, r_{vi}^2}}$$

โดย

- I_{uv} คือ set ของ items ที่ทั้ง user u และ v เคย rate
- r_{ui} คือ rating ที่ user u ให้กับ item i

3.3.3 Cold-Start Problem และแนวทางแก้ไขใน Greed Route

ข้อจำกัดสำคัญของ Collaborative Filtering คือ cold start problem ซึ่งหมายถึงปัญหาที่ระบบไม่สามารถแนะนำได้อย่างแม่นยำเมื่อผู้ใช้ใหม่หรือหนังสือใหม่ยังไม่มี interaction history เพียงพอ การใช้วิธี Content-Based เข้ามาร่วมกัน ก็จะสามารถแก้ไขปัญหานี้ได้ โดย

- New User ไม่มี rating history ใช้ Content-Based by Demographic (M1)
- New Item ไม่มี ratings จากผู้ใช้ ใช้ Content-Based โดยคำนวณ similarity จาก genre, tags, description embedding + Popularity boost

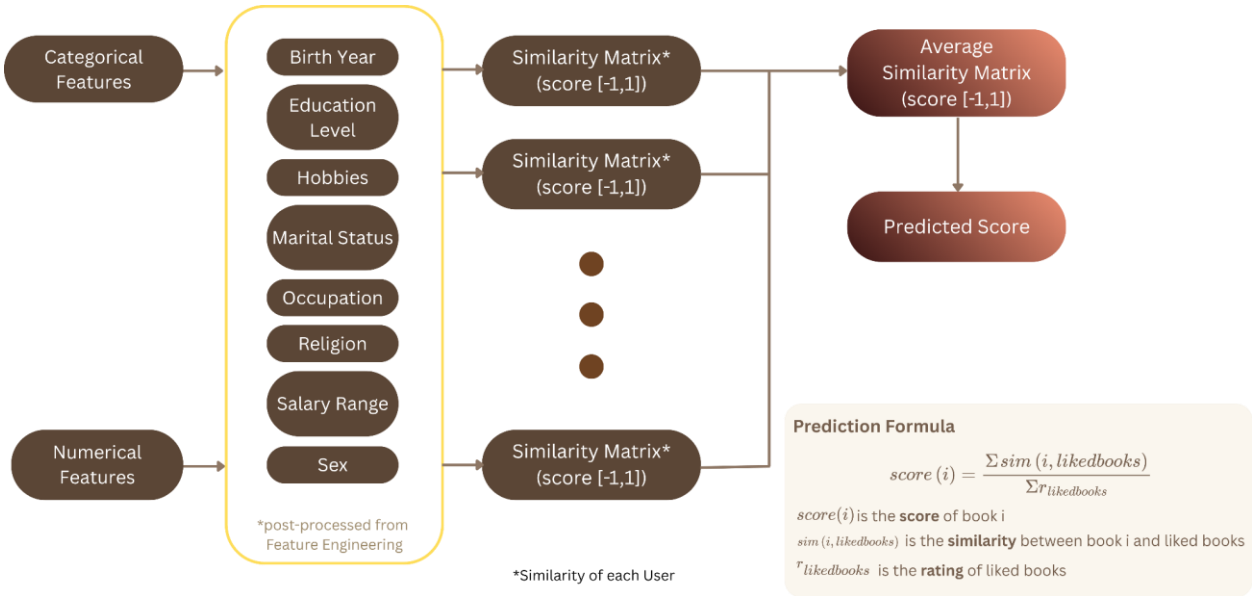
3.4 โมเดลของระบบ Greed Route

Greed Route ถูกออกแบบเป็น Hybrid Recommender ที่ใช้ 8 โมเดลทำงานร่วมกัน โดยอาศัยข้อมูลหลายประเภท เช่น demographic, content, behavior, rating และ commercial signal ซึ่งถูกแปลงเป็น feature ที่เหมาะสมกับแต่ละโมเดล เพื่อให้คะแนนหนังสือจากหลายมุมมอง ก่อนรวมคะแนนผ่านวิธี Hybrid Ensemble เป็น Final Score เดียว

ในโครงสร้างนี้ Model 1 – Model 6 เป็น scoring models ที่ทำงานขนานกันและให้คะแนนอย่างอิสระ ส่วน Model 7 จะทำหน้าที่คำนวณคะแนนกับ commercial score เพื่อให้ได้อันดับในการแนะนำหนังสือในอันดับ 1-4 และ 6-9 นอกจากนี้ Model 8 จะแนะนำหนังสือในอันดับ 5 และ 10 สำหรับ serendipity เพื่อเพิ่มความหลากหลายโดยไม่ต้องแข่งขันกับโมเดลให้คะแนนหลัก ในขณะที่ Trending ทำงานแยกเป็น non-personalized feed นอก Hybrid pipeline ส่งผลให้ระบบสามารถสร้าง recommendation ที่ครอบคลุม signal หลายมิติและมีความสมดุลทั้งด้านความแม่นยำและความหลากหลาย

โมเดล	รายละเอียดโมเดล	ประเภท	Output
M1	CB Demographic	Content-Based (User features)	Predicted score [−1, 1] ต่อผู้ใช้ต่อหนังสือ
M2	CB Metadata	Content-Based (Book Metadata features)	Predicted score [−1, 1] ต่อผู้ใช้ต่อหนังสือ
M3	CB Behavior Data	Collaborative (Item-Item)	Predicted score จาก intent level (- ∞, ∞) ต่อผู้ใช้ต่อหนังสือ
M4	CF Item × Item with Keyword Embedding	Collaborative (Item-Item) Embedded Review text	Predicted score [1, 5] ต่อผู้ใช้ต่อหนังสือ
M5	CF Item × Item with Rating	Collaborative (Item-Item)	Predicted score [1, 5] ต่อผู้ใช้ต่อหนังสือ
M6	CF User × User with Rating	Collaborative (User-User)	Predicted score [1, 5] ต่อผู้ใช้ต่อหนังสือ
M7	Non-Personalized Commercial	Commercial Blend (promotion_budget)	Min-Max Normalised score [0, commercial score weight] ต่อหนังสือ
M8	Surprise Recommendation	Serendipity	สุ่มหนังสือ 2 เล่มจาก percentile 40-60
Trending	Weighted Popularity	Non-Personalized (Rating aggregates (v, m, R, C))	Ranking สำหรับ Trending feed

3.4.1 Model 1: Content-Based by Demographic

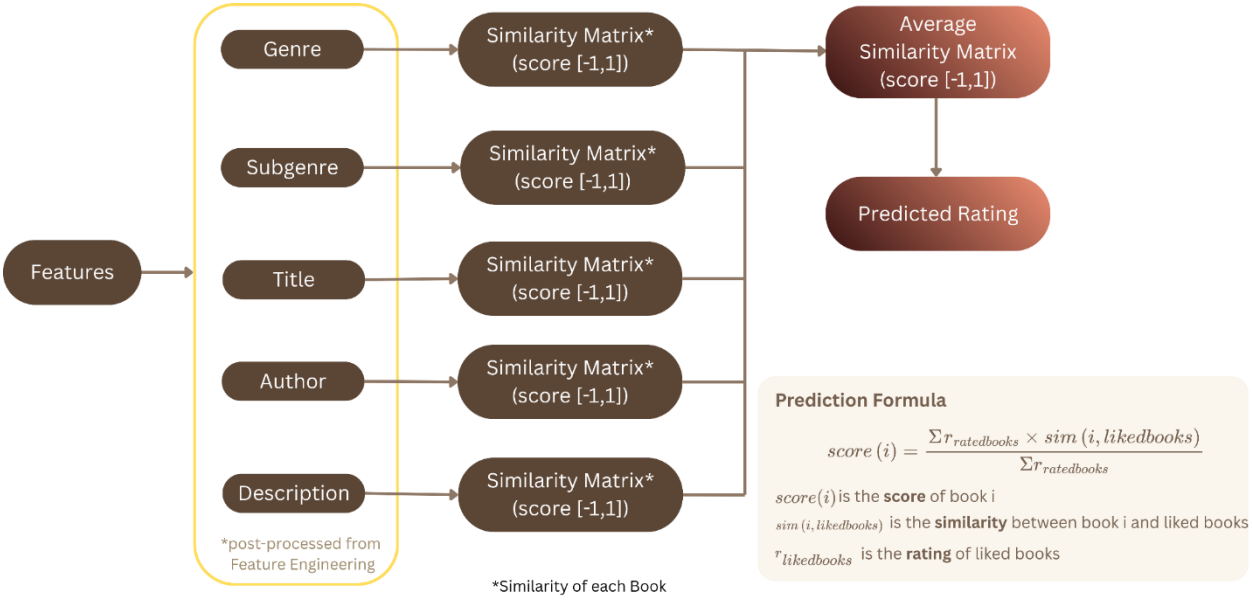


<u>Model 1</u> จะใช้ feature ของผู้ใช้ที่ผ่านการ encoding (nominal one/multi-hot และ ordinal normalization) เพื่อคำนวณค่า similarity ระหว่างผู้ใช้ หลังจากนั้นจึงรวมผลจากทุก feature เป็น Average Similarity Matrix แล้วใช้สูตรทำนายคะแนน โดย Model 1 จะเป็นกรณีพิเศษกว่าโมเดลอื่นด้วย เพราะถูกใช้เป็น cold-start เพียง model เดียวใน Model 1-6 (มี weight 50% สำหรับ cold-start สำหรับผู้ใช้ใหม่ที่มี rating 0-2 ครั้ง)

Feature: One/Multi-hot + User-mean + Min-max normalize

Similarity / Method: Average cosine similarity across features (user × user)

3.4.2 Model 2: Content-Based by Metadata

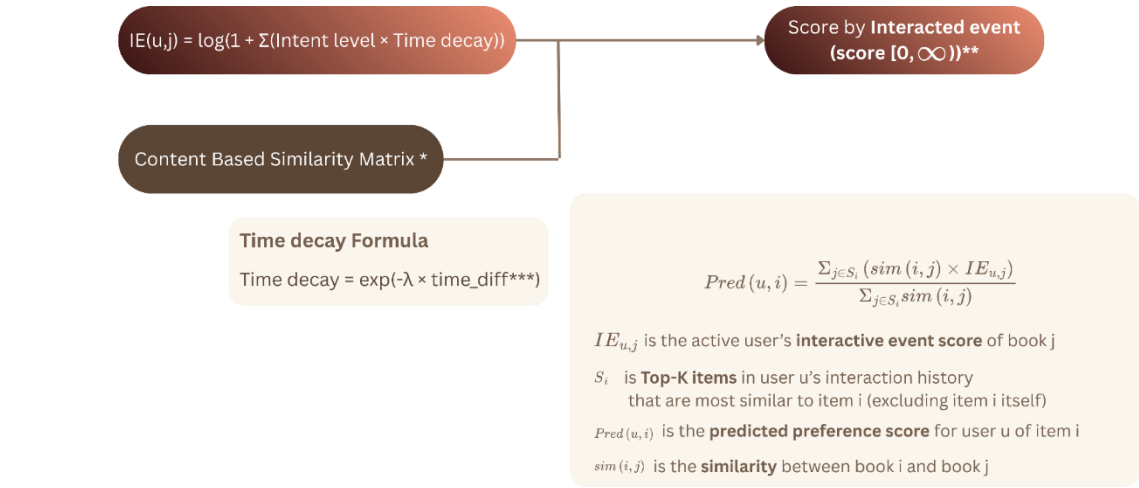


Model 2 จะใช้ feature ของหนังสือ ได้แก่ genre, subgenre, title embedding, author และ description ที่ผ่านการ embedding โดยคำนวณค่า similarity ระหว่างหนังสือ โดยใช้แนวทางเดียวกับ Model 1 แต่ข้อมูลจะเป็นหนังสือ ไม่ใช่ข้อมูลผู้ใช้

Feature: Multi-hot (genre, subgenre) + TF-IDF Keyword Embedding 768-dim (title + description)

Similarity / Method: Cosine similarity (book × book)

3.4.3 Model 3: Content-based Item × Item with Behaviour Data



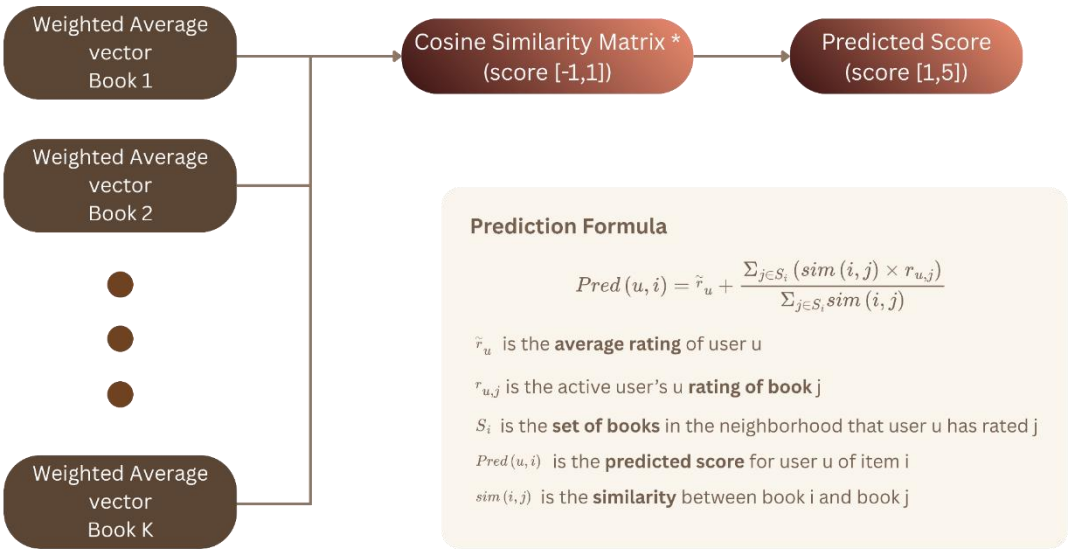
* Similarity of each book
** after log transform, typically ~0–5
*** time_diff in days

Model 3 คำนึงถึงระดับความสนใจของผู้ใช้ต่อหนังสือแต่ละเล่ม โดยใช้ Cosine-based similarity matrix จากโมเดลที่ 2 คูณถ่วงน้ำหนักร่วมกับ interaction-based score (IE) ซึ่งคำนวณมาจาก intent level ของพฤติกรรมผู้ใช้ร่วมกับ time decay โดยผลลัพธ์ที่ได้คือ predicted preference score ของผู้ใช้ต่อหนังสือแต่ละเล่ม

Feature: Interaction score (IE) จาก intent level และ time decay

Similarity / Method: ใช้ similarity matrix ของหนังสือร่วมกับการถ่วงน้ำหนักด้วย IE (similarity-weighted aggregation)

3.4.4 Model 4: CF Item × Item with Keyword Embedding from review text



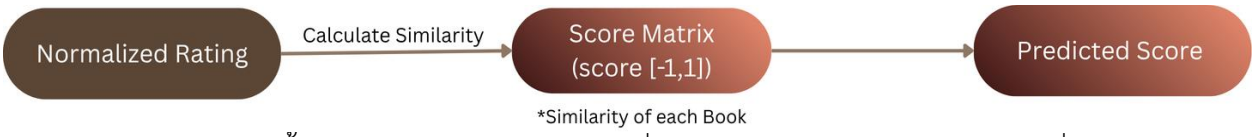
*Similarity of each book

Model 4 จะใช้ weighted average vector 768 dimensions ของแต่ละหนังสือ (จาก Model 2) มาคำนวณเป็น cosine similarity matrix ระหว่างหนังสือทุกคู่ จากนั้นจึงทำนาย predicted score ของผู้ใช้

Feature: TF-IDF-weighted Keyword Embedding 768-dim ต่อหนังสือ

Similarity / Method: Cosine similarity (book × book) + CF prediction formula

3.4.5 Model 5: CF Item × Item with Rating



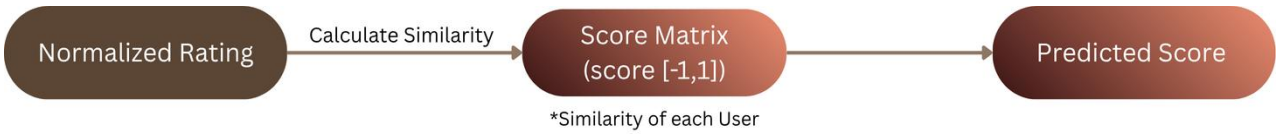
*Similarity of each Book

Model 5 จะใช้ normalized rating matrix (ผ่าน user-mean normalization ในช่วง [-1, 1]) คำนวณหาค่า similarity ระหว่างแต่ละหนังสือ จากเวกเตอร์ rating แล้ว Predicted Score

Feature: User-mean normalized rating matrix

Similarity / Method: Item-based Cosine Similarity (on normalized ratings)

3.4.6 Model 6: CF User × User with Rating

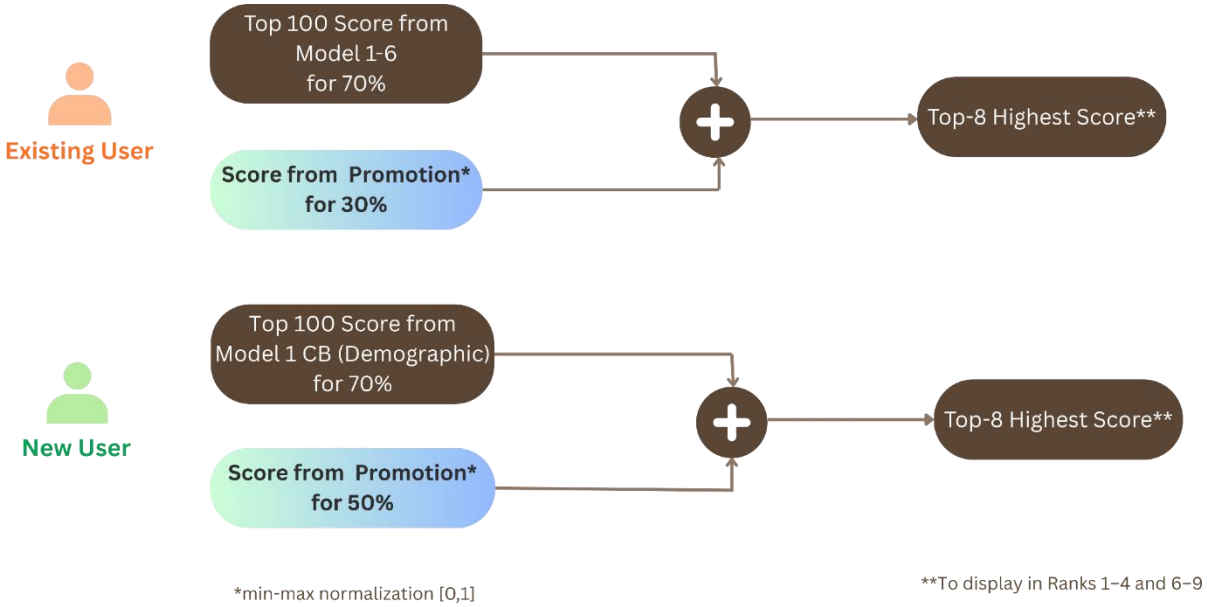


Model 6 จะมีโครงสร้างเดียวกับ Model 5 แต่ในการคำนวณ similarity จะใช้การคำนวณระหว่างผู้ใช้แทนระหว่างหนังสือ ใช้ normalized rating matrix ในช่วง [-1, 1] คำนวณหาค่า similarity ระหว่างแต่ละผู้ใช้ แล้วทำนาย Predicted Score

Feature: User-mean normalized rating matrix

Similarity / Method: User-based Cosine Similarity (on normalized ratings)

3.4.7 Model 7: Non-Personalized for Commercial



เพื่อรองรับโมเดลธุรกิจของแพลตฟอร์ม ระบบเปิดโอกาสให้สำนักพิมพ์และร้านหนังสือสามารถโปรโมทหนังสือได้โดยไม่กระทบต่อประสบการณ์ของผู้ใช้ ดังนั้น Model 7 จึงถูกออกแบบให้ผสานคะแนน Personal Score จาก Model 1–6 กับ Promotion Score โดยกำหนดสัดส่วนเป็น 70:30 สำหรับผู้ใช้เดิม และ 50:50 สำหรับผู้ใช้ใหม่ (ทั้งนี้ Predicted Score ของผู้ใช้ใหม่มาจาก Model 1 เพียงโมเดลเดียว)

การคำนวณจะดำเนินการในขั้นตอน Re-ranking จากรายการ 100 อันดับแรก ซึ่งเป็นผลลัพธ์จาก Model 1–6 เท่านั้น โดยออกแบบให้ทำงานหลังจากขั้นตอนการคัดกรองหลักเสร็จสมบูรณ์ เพื่อให้มั่นใจว่าหนังสือที่ถูกนำมาจัดอันดับใหม่ล้วนผ่านการประเมินแล้วว่ามีความสอดคล้องกับความสนใจของผู้ใช้

โดย Score คำนวณจาก promotion_budget ของแต่ละหนังสือ หลังผ่าน min-max normalization ให้อยู่ในช่วง [0, 1] การแยกคะแนนนี้ออกจากคะแนนหลัก ทำให้สามารถควบคุมได้ว่า promotion_budget ส่งผลต่ออันดับในการแนะนำหนังสืออย่างไรบ้าง

ในการทำงานของ Model 7 ไม่ใช่โมเดล stand-alone แต่เป็นกลไกผสมคะแนน Personal Score กับ Promotion Score เพื่อสร้าง revenue stream ให้แพลตฟอร์ม โดยแยกเป็น 2 แบบ

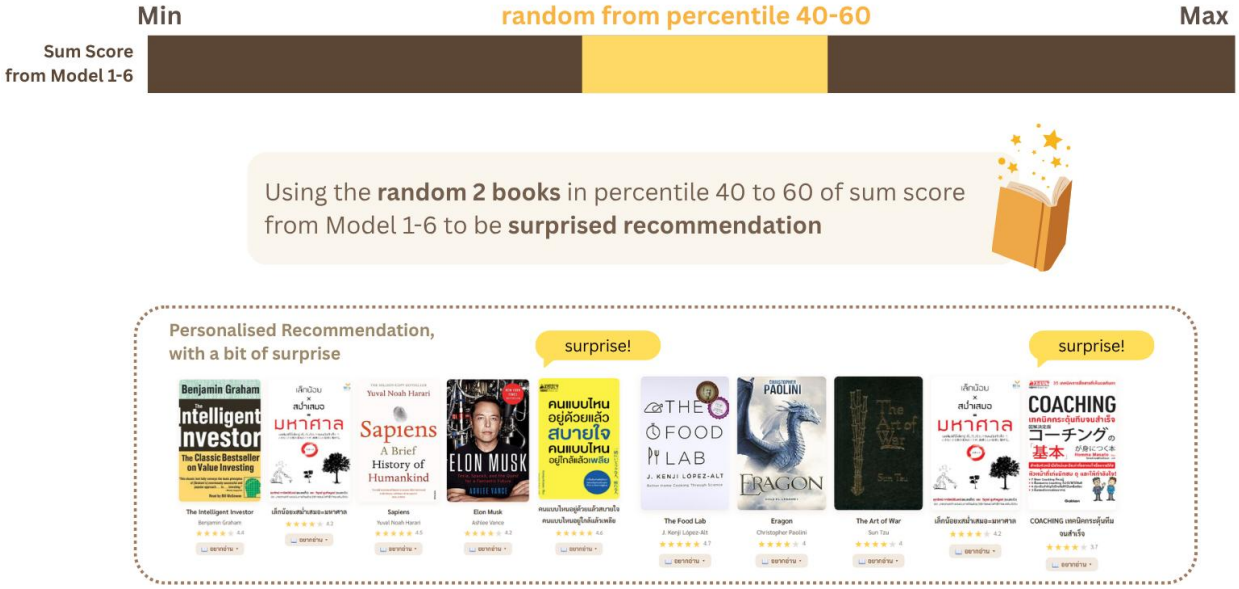
ประเภทผู้ใช้งาน	โมเดล	Weighted Personal score	Weighted Promotion score	Ranking result
Existing User	Model 1-6 (รวมกัน)	70%	30%	Rank 1-4 & 6-9
New User	Model 1 เท่านั้น (CB Demographic)	50%	50%	Rank 1-4 & 6-9

Weighted Promotion Score ที่สูงขึ้นสำหรับผู้ใช้ใหม่ สะท้อนให้เห็นว่าเมื่อข้อมูลความชอบของผู้ใช้ยังมีจำกัด การให้ความสำคัญกับรายการที่ได้รับการโปรโมทจะช่วยเพิ่มโอกาสในการเกิด Conversion และเพิ่มการเก็บข้อมูลพฤติกรรมที่มากขึ้น เพื่อนำไปใช้แก้ปัญหา Cold Start ในอนาคต

Feature: Min-max normalize promotion_budget; Top-100 Book's score

Similarity / Method: Weighted blend: 70% personal + 30/50% promotion

3.4.8 Model 8: Surprise Recommendation (Rank 5 & 10)



เพื่อเพิ่มความแปลกใหม่ในระบบแนะนำ Greed Route ได้ออกแบบ slot สำหรับ Surprise Recommendation โดยใช้คะแนนรวม (Sum Score) จาก Model 1–6 มาคำนวณ percentile ของหนังสือทั้งหมด และสุ่มเลือกหนังสือ 2 เล่มจากช่วง percentile 40–60 ซึ่งเป็นช่วงคะแนนระดับกลาง เพื่อหลีกเลี่ยงทั้งหนังสือยอดนิยมที่คาดเดาได้และหนังสือที่มีความเกี่ยวข้องต่ำ ทั้งนี้ หนังสือที่สุ่มได้จะถูกแทรกในอันดับที่ 5 และ 10 ของรายการแนะนำ Top-10

Feature: Percentile ranking

Similarity / Method: Random sampling ในช่วง percentile 40-60

3.4.9 Trending Recommendation (สำหรับผู้ใช้ทุกคน)

สำหรับ Non-personalized feed ที่ใช้ Weighted Rating Formula สำหรับจัดอันดับหนังสือยอดฮิตและหนังสือใหม่ ระบบแนะนำแบบไม่ขึ้นกับผู้ใช้ (Trending Recommendation) จะใช้ Weighted Rating ตามสูตรของ IMDB Top 250 เพื่อหลีกเลี่ยง bias จากหนังสือที่มี rating สูงแต่มีจำนวน rating น้อย คือ

$$WR = (\frac{v}{v+m} xR) + (\frac{m}{v+m} xC)$$

โดย

- v คือ จำนวน rating ของหนังสือเล่มนั้น
- m คือ จำนวน rating ขั้นต่ำที่กำหนดให้เข้ารอบ Top Chart
- R คือ ค่าเฉลี่ย rating ของหนังสือเล่มนั้น
- C คือ ค่าเฉลี่ย rating ของหนังสือทั้งหมดใน dataset

จากสมการนี้ หนังสือที่มีคะแนนดีและมีจำนวนรีวิวเพียงพอผ่าน threshold จะถือว่าถูก verified

Trending Recommendation จะถูกวางในหน้าแรก (ใน tab 'สำรวจ') และหน้า browse-explore เพื่อช่วยให้ผู้ใช้ค้นพบหนังสือใหม่ๆ ที่ไม่ได้ขึ้นมาจาก personalized feed

Feature: v, m, R, C scalars

Similarity / Method: Weighted Rating Formula (IMDB-style)

3.5 Hybrid Recommendation

Hybrid Recommendation เป็นการรวมคะแนนจากหลายโมเดลเพื่อใช้จุดแข็งของแต่ละวิธีร่วมกัน และเพิ่มความหลากหลายของข้อมูลที่ใช้ในการแนะนำ

ระบบของ Greed Route ใช้การรวมคะแนนจากแต่ละโมเดลแบบ Weighted Ensemble หรือการรวมคะแนนแบบถ่วงน้ำหนัก โดยแต่ละโมเดลสามารถมีน้ำหนักแตกต่างกันได้ ซึ่งในที่นี้ คะแนนจากโมเดลหลักทั้ง 6 โมเดล (M1–M6) จะถูกนำมารวมเป็น Personal Score ก่อนเพิ่ม Commercial Score จาก Model 7 เพื่อเลือกหนังสือที่มีคะแนนสูงสุด 8 อันดับ และแนะนำให้แก่ผู้ใช้

ขั้นตอนการคำนวณ Hybrid Score มีดังนี้

1. รับคะแนนจากโมเดลหลักทั้ง 6 โมเดล (M1–M6)
2. ปรับคะแนนแต่ละโมเดลด้วย Min-Max Normalization ให้คะแนนอยู่ในช่วง [0,1]
3. คูณคะแนนจาก (2) ด้วยน้ำหนักของแต่ละโมเดล
4. รวมคะแนนถ่วงน้ำหนักทั้งหมดเป็น Personal Score
5. เพิ่ม Commercial Score จาก Model 7 ตามสัดส่วนที่กำหนด
6. จัดอันดับหนังสือและเลือก Top-8 ที่ได้ NDCG@K สูงสุดเป็นผลลัพธ์สุดท้าย

โดยขั้นตอนที่ 3–4 เป็นกระบวนการ Weighted Ensemble ซึ่งเป็นการรวมคะแนนจากหลายโมเดลโดยการถ่วงน้ำหนักและรวมเป็นคะแนนเดียว

คำนวณ Hybrid Score

3.5.1 Weighted Ensemble

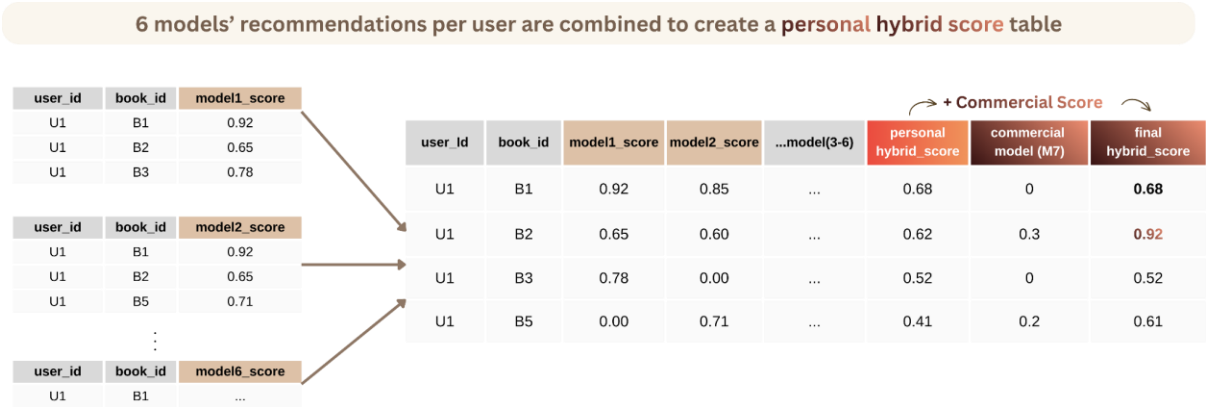
ในการคำนวณ Hybrid Score ระบบแบ่งคะแนนออกเป็น 2 ส่วนหลัก ได้แก่ Personal Score และ Commercial Score โดย Personal Score สะท้อนความสนใจของผู้ใช้แต่ละคนจากทั้ง 6 โมเดล (M1–M6) ขณะที่ Commercial Score (Model 7) สะท้อนปัจจัยเชิงธุรกิจ กล่าวคือ promotion_budget ที่สำนักพิมพ์จ่ายให้เพื่อโอกาสที่หนังสือจะถูกแนะนำ โดยการรวมคะแนน Personal Score จากทั้ง 6 โมเดลใช้แนวทาง Weighted Ensemble ซึ่งเป็นการนำคะแนนจากแต่ละโมเดลที่ผ่านการปรับให้อยู่ในช่วงเดียวกันคูณกับน้ำหนักของโมเดลนั้นๆ และรวมเป็นคะแนนเดียว

ขั้นตอนการคำนวณ Weighted Ensemble มีดังนี้

1. **Score Aggregation :** รวมคะแนนจากโมเดลทั้ง 6 (M1–M6) บน (user_id, book_id)
2. **Score Normalization:** ปรับคะแนนของแต่ละโมเดลให้อยู่ในช่วงเดียวกันด้วย Min-Max Normalization [0,1]

3. **Personal Score Computation - Weighted Ensemble (Apply Model Weights):** คูณคะแนนของแต่ละโมเดลด้วยน้ำหนักที่กำหนด เพื่อสะท้อนความสำคัญของโมเดลนั้นๆ เพื่อให้ได้ Personal Score ของแต่ละคู่ระหว่างผู้ใช้และหนังสือ (user, book)
4. **Commercial Adjustment (Model 7):** เพิ่ม Commercial Score (M7) เพื่อสะท้อนปัจจัยเชิงธุรกิจ เช่น promotion หรือความนิยม
5. **Ranking:** จัดอันดับหนังสือตาม Final Score และเลือก Top-K ในที่นี้คือ Top-8 เป็นผลลัพธ์

ตัวอย่างการรวมของผลลัพธ์ของแต่ละโมเดลและการคำนวณ hybrid score



จากขั้นตอนข้างต้น คะแนนของแต่ละโมเดลจะถูกปรับให้อยู่ในช่วงเดียวกันก่อนนำมาคูณกับน้ำหนักที่กำหนด เพื่อให้คะแนนจากโมเดลที่มีสเกลต่างกันสามารถนำมารวมกันได้อย่างเหมาะสม โดยระบบกำหนดสัดส่วนคะแนนระหว่าง Personal Score และ Commercial Score แตกต่างกันตามประเภทผู้ใช้ โดยให้ i แทนลำดับของโมเดลที่ 1-7 (i $\in$ {1,...,7}) และ $s_{i,u,b}$ แทนคะแนนจากโมเดลที่ i สำหรับผู้ใช้ u และหนังสือ b และ w_i แทนน้ำหนักของโมเดลที่ i

3.5.2 Weight Tuning ด้วย Bayesian Optimization

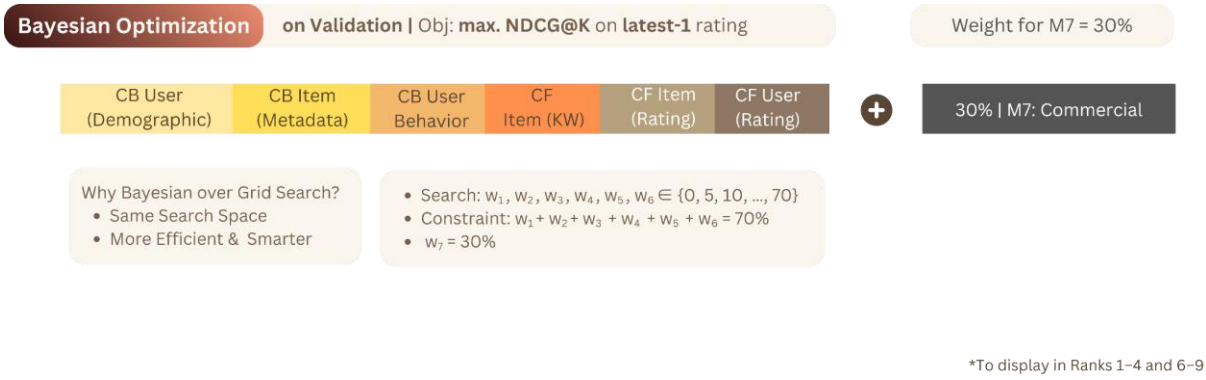
การกำหนดน้ำหนักของโมเดลในส่วน Personal Score โดยใช้วิธี Bayesian Optimization และประเมินผลบน validation set เพื่อค้นหาชุดน้ำหนักที่เหมาะสมที่สุดสำหรับ Model 1-6

โดยกำหนด objective function เป็นการ maximize ค่า NDCG@K บน latest-1 rating ของผู้ใช้แต่ละคน ซึ่งสะท้อนความสามารถของระบบในการจัดอันดับหนังสือที่ผู้ใช้ให้คะแนนล่าสุดในชุด validation สามารถเขียนได้ดังนี้

- Objective: max NDCG@K on latest-1 interaction
- Search space: w_1 , w_2 , w_3 , w_4 , w_5 , w_6 $\in$ {0, 5, 10, ..., 70}
- Constraint: $w_1 + w_2 + w_3 + w_4 + w_5 + w_6 = 70\%$
- $w_7 = 30\%$

Bayesian Optimization ถูกเลือกแทน Grid Search เนื่องจากสามารถเรียนรู้ลักษณะของ objective function ผ่าน surrogate model เช่น Gaussian Process ทำให้ค้นหาค่าที่เหมาะสมได้เร็วและมีประสิทธิภาพมากกว่า โดยไม่จำเป็นต้องทดลองทุก combination ใน search space

Weight Tuning ด้วย Bayesian Optimization



3.5.3 Hybrid Recommender Implementation

สำหรับการนำไปใช้และ Deploy จริง ค่าชุดน้ำหนักที่ดีที่สุดบน validation set ที่ได้จาก 3.5.2 จะถูกนำไปใช้กับผู้ใช้ที่มีอย่างน้อย 1 rating ในขณะที่ผู้ใช้ที่ไม่มีประวัติ rating เลย ระบบจะแนะนำหนังสือจาก Model 1 (Content-based Demographic) และ Commercial Score เท่านั้น โดยกำหนดให้มีสัดส่วนน้ำหนักโมเดลละ 50% เนื่องจากโมเดลอื่นต้องอาศัยข้อมูลพฤติกรรมผู้ใช้ ซึ่งยังไม่เพียงพอในกรณีนี้

1. **Existing Users (1+ rating(s))** สำหรับผู้ใช้ที่มีประวัติ rating อย่างน้อย 1 รายการ ระบบจะใช้คะแนนจากโมเดลหลักทั้ง 6 โมเดล (M1–M6) ในการคำนวณ Personal Score ดังนี้

$$Personal\ Score_{u,b} = \sum w_i \times normalize(s_{i,u,b})$$

โดยกำหนดให้ $\sum w_i$ = 0.7 (i ∈ {1,...,6}) ซึ่งเป็นการจำกัดค่าสูงสุดของ Personal Score ไว้ที่ 0.7 จากนั้นระบบจะนำ Personal Score มาบวกกับ Commercial Score จาก Model 7 ซึ่งมาจาก promotion factor ที่ผ่านการ normalize และถ่วงน้ำหนักแล้ว ดังนี้

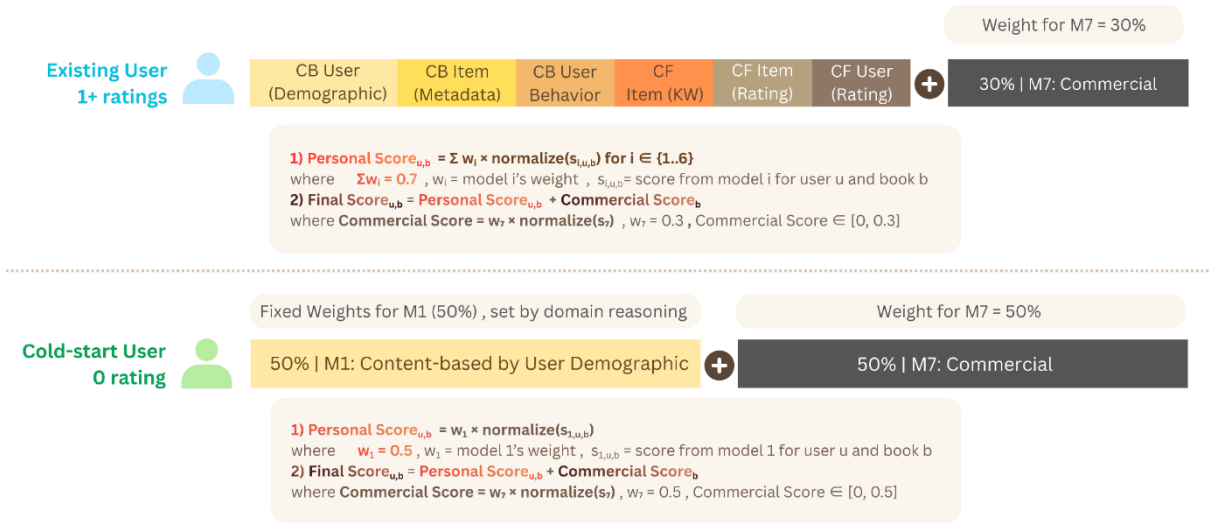
$$Final\ Score_{u,b} = Personal\ Score_{u,b} + Commercial\ Score_b$$

สำหรับ Existing Users ใช้สัดส่วน Personal Score = 0.7 และ Commercial Score = 0.3

2. **New Users (0 rating):** สำหรับผู้ใช้ใหม่ที่มีข้อมูลจำกัด ระบบจะใช้เฉพาะ Model 1 Content-based Demographic ซึ่งอิงจากข้อมูลผู้ใช้ เนื่องจากโมเดลอื่นต้องอาศัยข้อมูลพฤติกรรมผู้ใช้เพิ่มเติม ดังนั้นสัดส่วนคะแนนรวมจะเป็น Personal Score = 0.5 และ Commercial Score = 0.5

$$Final\ Score_{u,b} = Personal\ Score_{u,b} + Commercial\ Score_{b}$$

Hybrid Recommender Implementation



บทที่ 4

Evaluation

4.1 การออกแบบ Offline Evaluation

เพื่อประเมินความสามารถของระบบแนะนำหนังสือในการคัดเลือกและจัดอันดับรายการที่สอดคล้องกับความสนใจของผู้ใช้ โดยอ้างอิงจากชุดข้อมูลทดสอบ (test dataset) โดยทำการประเมินประสิทธิภาพตามกลุ่มผู้ใช้ 2 กลุ่ม ได้แก่ Cold start (rating 1 ครั้ง) และ Existing User (rating อย่างน้อย 2 ครั้ง)

โดยมีตัวชี้วัดในการประเมิน ได้แก่

4.1.1. Normalized Discounted Cumulative Gain (NDCG)

NDCG (Normalized Discounted Cumulative Gain) เป็นตัวชี้วัดที่ใช้ประเมินคุณภาพของการจัดอันดับ โดยพิจารณาว่าระบบสามารถเรียงลำดับรายการให้สอดคล้องกับความสนใจของผู้ใช้ได้ดีเพียงใด โดยรายการที่มีความเกี่ยวข้องสูงควรปรากฏในอันดับต้นของผลลัพธ์ ตัวชี้วัดนี้ให้น้ำหนักกับอันดับต้นมากกว่าอันดับล่างผ่านการถ่วงด้วยฟังก์ชันลอการิทึม และมีการปรับค่าให้อยู่ในช่วง 0 ถึง 1 โดยเทียบกับลำดับที่ดีที่สุด (ideal ranking) ทำให้สามารถเปรียบเทียบประสิทธิภาพของโมเดลได้อย่างเหมาะสม

$$NDCG = \frac{DCG}{IDCG} = \left[\sum_{i=1}^{K} \frac{2^{rel[i]} - 1}{log_2([i] + 1)}\right] / IDCG$$

โดยที่

- $rel[i]$ คือ ระดับความเกี่ยวข้อง (relevance) ของรายการในอันดับที่ i
- $IDCG$ คือ ค่า DCG ที่ได้จากการจัดลำดับที่ดีที่สุด (ideal ranking)

ในการประเมินผล ค่า NDCG@K จะถูกคำนวณแยกสำหรับผู้ใช้แต่ละราย และนำมาเฉลี่ยเพื่อสะท้อนประสิทธิภาพโดยรวมของโมเดล โดยค่าที่เข้าใกล้ 1 แสดงว่าโมเดลสามารถจัดอันดับรายการได้สอดคล้องกับความสนใจของผู้ใช้ทั้งในด้านความถูกต้องและความเหมาะสมของลำดับการจัดเรียง

<u>ตัวอย่างการประเมินผล</u>

Recommended Ranking	User A	User B
Rank 1 | Book A	Book A : Rating 5	Book B : Rating 5
...		
Rank 10 | Book J		
NDCG	NDCG = 1	NDCG = 0.63

4.1.2. Hit Rate@K

เป็นตัวชี้วัดที่ใช้ประเมินระบบแนะนำ โดยตรวจสอบว่ารายการที่ผู้ใช้สนใจปรากฏอยู่ใน Top-K หรือไม่ หากพบรายการดังกล่าวจะให้ค่าเป็น 1 และหากไม่พบจะให้ค่าเป็น 0 ในการศึกษานี้กำหนดให้ผู้ใช้แต่ละรายมีรายการที่สนใจจริงเพียง 1 รายการ (leave-one-out evaluation) ทำให้ Hit Rate@K สามารถตีความเป็นฟังก์ชันตัวบ่งชี้ (indicator function) สำหรับการวัดผลได้

$$Hit\ Rate@K\ = 1\ if\ the\ relevant\ item\ is\ in\ top\ K\ and\ 0\ otherwise$$

เมื่อทำการเฉลี่ยค่า Hit Rate@K ของผู้ใช้ทั้งหมด จะได้ค่าที่สะท้อนประสิทธิภาพโดยรวมของโมเดล โดยค่าที่ใกล้ 1 หมายถึงระบบสามารถแนะนำรายการที่ผู้ใช้สนใจได้อย่างมีประสิทธิภาพสูง

<u>ตัวอย่างการประเมินผล</u>

Recommended Ranking	User A	User B
Rank 1 | Book A	Book A : Rating 5	Book K : Rating 5
...		
Rank 10 | Book J		
Book K		
Hit Rate@10	**1**	**0**

4.2 การออกแบบ Online Evaluation

หลังจากทำการ deploy ระบบแนะนำแล้ว จะทำการประเมินประสิทธิภาพของโมเดลด้วย Online Evaluation โดยใช้ตัวชี้วัดที่สะท้อนพฤติกรรมของผู้ใช้จริง เพื่อคัดเลือกโมเดลที่มีประสิทธิภาพดีที่สุดและนำมาใช้เป็น baseline สำหรับการพัฒนาในรอบถัดไป

4.2.1 Click-Throught Rate (CTR)

ใช้ในการวัดความสามารถของระบบในการดึงดูดให้ผู้ใช้คลิกคำแนะนำ ซึ่งใช้ข้อมูลที่มาจากตาราง recommendation_log ร่วมกับข้อมูลจากตาราง book และ genre

$$CTR = \frac{\text{จำนวนครั้งที่คลิก}}{\text{จำนวนครั้งที่แสดงทั้งหมด}}$$

4.2.2 Funnel Conversion Analysis

ใช้ติดตามพฤติกรรมของผู้ใช้หลังจากการคลิก ตั้งแต่ click,read, rating, purchase เพื่อประเมินทั้งด้าน engagement และ conversion ของระบบ โดยคำนวณค่าต่อไปนี้จากข้อมูลในตาราง user_book_event

- $Purchase\ rate\ = \frac{\text{จำนวนผู้ใช้ที่ซื้อ}\ (Intent = 4)}{\text{จำนวนผู้ใช้ที่คลิก}}$
- $Read\ rate\ = \frac{\text{จำนวนผู้ใช้ที่อ่าน}\ (Intent = 5)}{\text{จำนวนผู้ใช้ที่คลิก}}$
- $Rating\ rate\ = \frac{\text{จำนวนผู้ใช้ที่รีวิว}\ (Intent = 7)}{\text{จำนวนผู้ใช้ที่คลิก}}$

4.2.3 Weighted Scoring Model

วิธีการประเมินและเปรียบเทียบทางเลือกต่าง ๆ โดยการนำหลายตัวชี้วัด (metrics) มารวมกันเป็นคะแนนเดียว (composite score) ผ่านการกำหนด น้ำหนัก (weights) ให้แต่ละตัวชี้วัดตามความสำคัญ

$$\text{Score} = 0.6\ \text{Conversion} + 0.2\ \text{CTR} + 0.2\ \text{Retention}$$

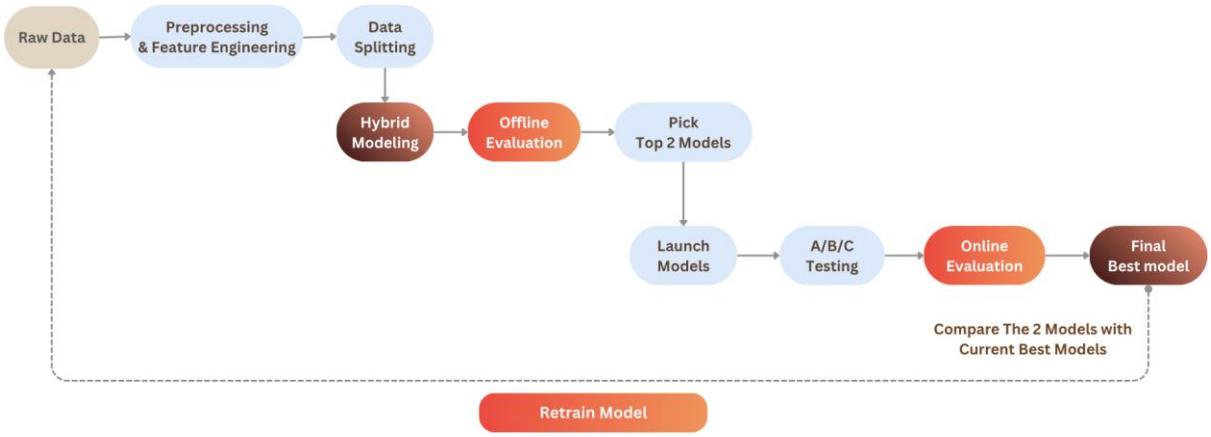
Retention ใช้ประเมินความสามารถของระบบในการรักษาผู้ใช้ในระยะยาว โดยพิจารณาจากสัดส่วนของผู้ใช้ที่กลับมาใช้งานซ้ำผ่านข้อมูล session_id และ user_id เมื่อได้โมเดลที่มีคะแนนสูงที่สุด จะทำการทดสอบนัยสำคัญทางสถิติ (Statistical Hypothesis Testing) โดยใช้ Chi-square test เพื่อเปรียบเทียบกับโมเดลเดิม หากผลการทดสอบสามารถปฏิเสธสมมติฐานหลัก (H0) ได้ จะสรุปได้ว่าโมเดลใหม่มีประสิทธิภาพดีกว่าอย่างมีนัยสำคัญทางสถิติ และจะนำโมเดลดังกล่าวมาใช้เป็น baseline สำหรับการพัฒนาระบบในรอบถัดไป

บทที่ 5
สรุปภาพรวมและหน้าเว็บไซต์ต้นแบบ

สรุปภาพรวมการออกแบบระบบแนะนำหนังสือ Greed Route และตัวอย่างเว็บไซต์ต้นแบบ

5.1 สรุปภาพรวมสถาปัตยกรรมระบบ

ภาพรวมสถาปัตยกรรมระบบ



1. **Data Collection & Intent Capture**: ระบบได้จัดเก็บข้อมูลพฤติกรรม โดยแปลงเป็นคะแนนความสนใจ 9 ระดับ (Level of Intent) ตั้งแต่การเลื่อนดู ไปจนถึงการให้คะแนนและการซื้อ รวมกับการเก็บข้อมูล Metadata ของหนังสือ และข้อความรีวิว
2. **Data Pre-Preprocessing & Feature Engineering:** ทำการเปลี่ยนข้อมูลตัวหนังสือให้อยู่ในรูป Feature Vectors เช่น
   - การทำ TF-IDF Weighted Keyword Embedding เพื่อสกัดความหมายจากข้อความรีวิว
   - การทำ Normalization เพื่อปรับคะแนนต่างๆ ให้อยู่ในมาตรฐานเดียวกัน
   - การทำ One-hot/Multi-hot Encoding สำหรับข้อมูลหมวดหมู่
3. **Data Splitting:** ทำการแบ่งข้อมูลตามลำดับตามเวลา (Temporal Split) เพื่อจำลองสถานการณ์จริงที่ระบบต้องใช้พฤติกรรมในอดีตในการทำนายพฤติกรรมในอนาคต โดยแบ่งผู้ใช้ตามจำนวน rating
4. **Personal Scoring Model (M1 - M6):** ทำการพัฒนาโมเดล ซึ่งแต่ละโมเดลจะให้คะแนนด้วยวิธีที่แตกต่างกัน ทำให้สามารถแนะนำหนังสือให้ผู้ใช้แต่ละคนได้ครอบคลุมทุกมิติ ได้แก่
   1. Content-Based*:* M1 (ข้อมูล Demographic ของผู้ใช้ที่คล้ายกัน), M2 (Metadata ของหนังสือที่คล้ายกัน), M3 (ความคล้ายคลึงกันของหนังสือ โดยใช้ข้อมูลพฤติกรรมผู้ใช้ในหนังสือเล่มนั้นๆ)

2. Collaborative Filtering: M4 (ความคล้ายคลึงจาก keyword ใน reviews), M5 (Item-based Rating), M6 (User-based Rating)

5. Hybrid Ensemble & Commercial Scoring Model (M7): ทำ Weight Tuning เพื่อหาชุดน้ำหนักที่ได้ค่า NDCG@K ที่สูงสุด 2 อันดับแรกบน validation set โดยนำคะแนนจากโมเดล M1 ถึง M6 มารวมกันด้วยเทคนิค Weighted Ensemble จากนั้นรวมกับ Commercial Score จาก M7 เพื่อจัดลำดับหนังสือที่ได้คะแนนดีที่สุด 8 อันดับแรก และแนะนำหนังสือในลำดับ 1-4 และ 6-9 ให้แก่ผู้ใช้
6. Serendipity (M8): นอกจากนี้ได้เพิ่มความหลากหลายของการแนะนำหนังสือ โดยสุ่มเลือกหนังสือใน Percentile ที่ 40-60 เป็น Surprise Recommendation แนะนำหนังสือในลำดับ 5 และ 10
7. Trending: แนะนำหนังสือที่เป็นที่นิยม เพื่อการแนะนำแบบ Non-Personalized
8. Offline Evaluation: ทำการวัดผลชุด Weight ที่ดีที่สุดบน test set โดยประเมินด้วย NDCG@K และ HitRate@K
9. Online Evaluation: ทำ A/B/C Testing โดยเปรียบเทียบ 2 โมเดลที่ได้มาจากขั้นตอนที่ 5 กับโมเดลที่ดีสุดในรอบก่อนหน้า โดยประเมินด้วย weighted score ว่าโมเดลไหนได้ผลลัพธ์ดีที่สุด แล้วทำการทดสอบโดยใช้ Chi-square test เพื่อวัดผลว่าโมเดลใหม่ดีกว่าโมเดลเดิมโดยมีนัยสำคัญทางสถิติ
10. Delivery & Feedback Loop: ระบบส่งผลลัพธ์ทั้งหมดไปแสดงบนหน้าเว็บไซต์ และเมื่อผู้ใช้มีการใช้งานข้อมูลจะถูกจัดเก็บลง user_book_event เพื่อส่งกลับไปเทรนโมเดลให้แม่นยำมากขึ้นในรอบต่อไป

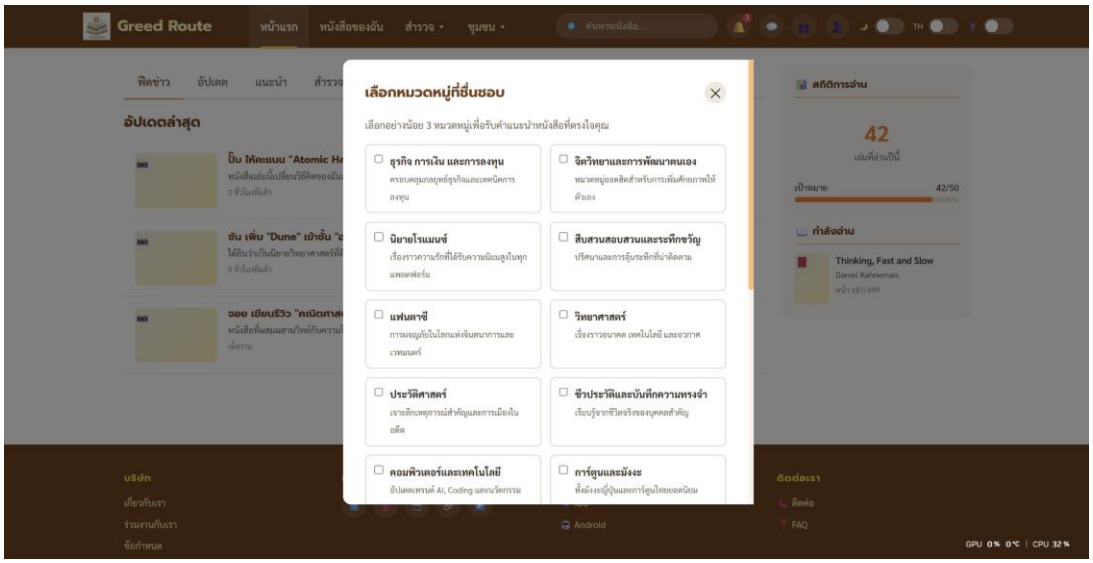
5.2 การออกแบบหน้าเว็บไซต์ต้นแบบ (Website Prototype Design)

แพลตฟอร์ม Greed Route ได้จำลองการทำงานผ่าน Website Prototype โดยออกแบบ User Interface (UI) ให้สอดคล้องและทำงานร่วมกับผลลัพธ์จากอัลกอริทึมทั้ง 8 โมเดลโดยตรง เพื่อให้เกิดเป็นระบบ Hybrid Recommender System ที่สมบูรณ์ โดยสามารถแบ่งออกเป็น 4 ส่วนหลัก ดังนี้

1. หน้าลงทะเบียน

สำหรับ User Onboarding & Genre Selection เมื่อผู้ใช้ลงทะเบียนครั้งแรก ระบบจะเก็บข้อมูล Demographic เช่น hobbies, education และแสดง Genre ให้เลือก 2–5 หมวดหมู่ของหนังสือที่สนใจ ซึ่งจะถูกเก็บใน user_genre_preference table และใช้เป็น cold-start signal สำหรับ Content-Based Filtering (M1 - User Demographic)

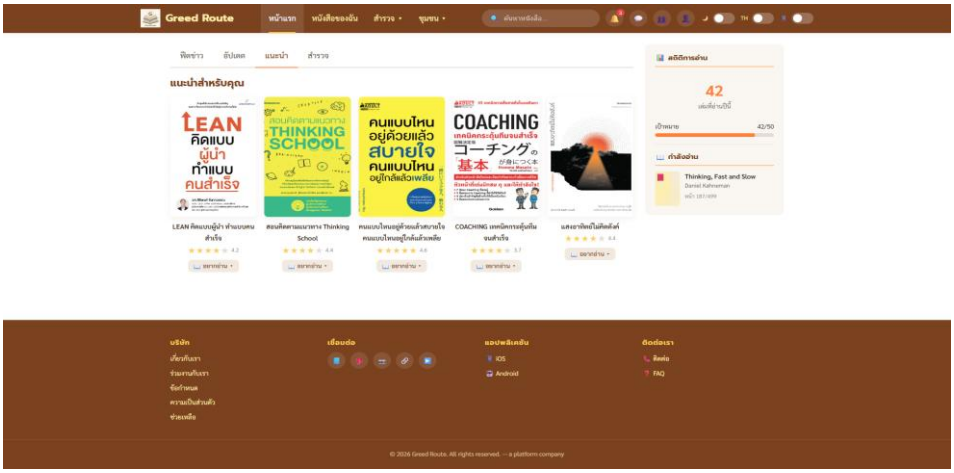
หน้าลงทะเบียน



2. หน้า "Recommended for You"

เป็นพื้นที่หลักสำหรับแสดงผลลัพธ์จาก Hybrid Models (M1-M6) ที่ผสมกับโมเดล M7 (Commercial) ซึ่งมาจากการรวม Top ranking กับสัดส่วนเชิงธุรกิจ (Promotion Budget) เพื่อให้เกิดความสมดุลระหว่างสิ่งที่ผู้ใช้ชอบและสิ่งที่แพลตฟอร์มต้องการนำเสนอ รวมถึงในตำแหน่งที่ 5 และ 10 ระบบจะใช้ โมเดล Surprise Recommendation (M8) ในการสุ่มหนังสือในกลุ่ม Percentile กลางๆ (40-60) เพื่อสร้างความแปลกใหม่และความสนใจใหม่ๆ

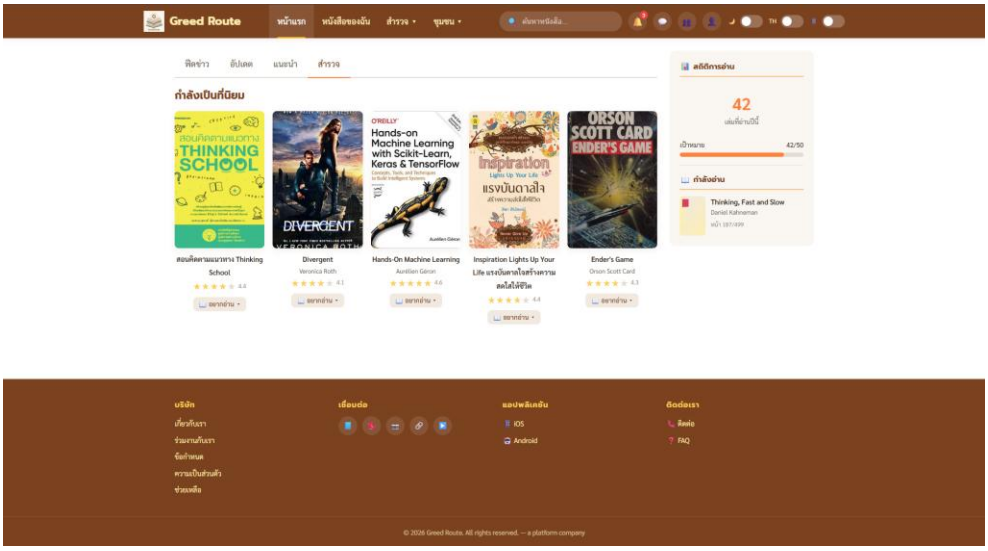
หน้า "Recommended for You"



3. หน้า "Trending Now"

หน้าสำหรับการแนะนำหนังสือที่เป็นที่นิยมโดยไม่อิงกับโปรไฟล์ส่วนตัวของผู้ใช้

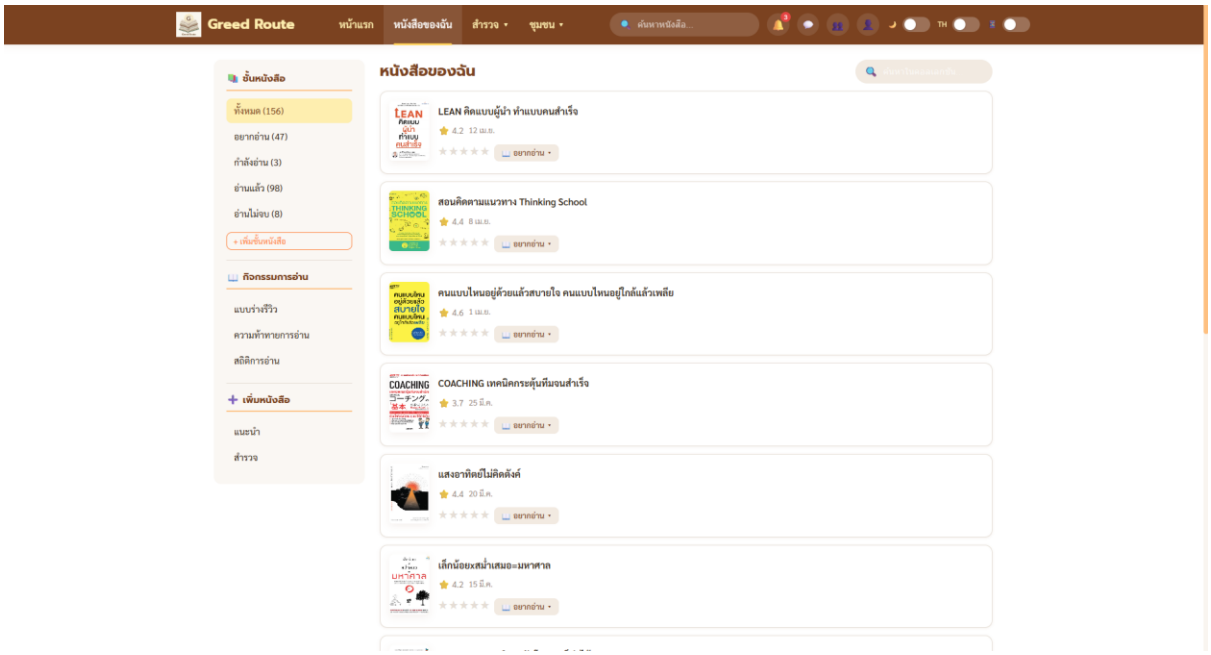
หน้า "Trending Now"



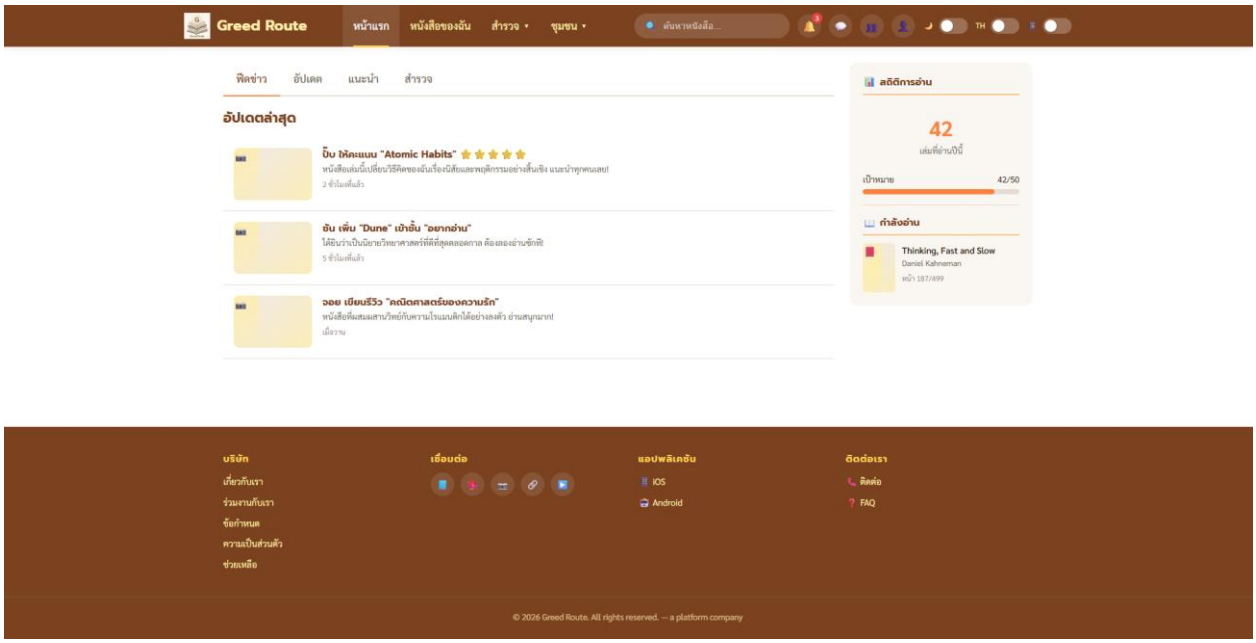
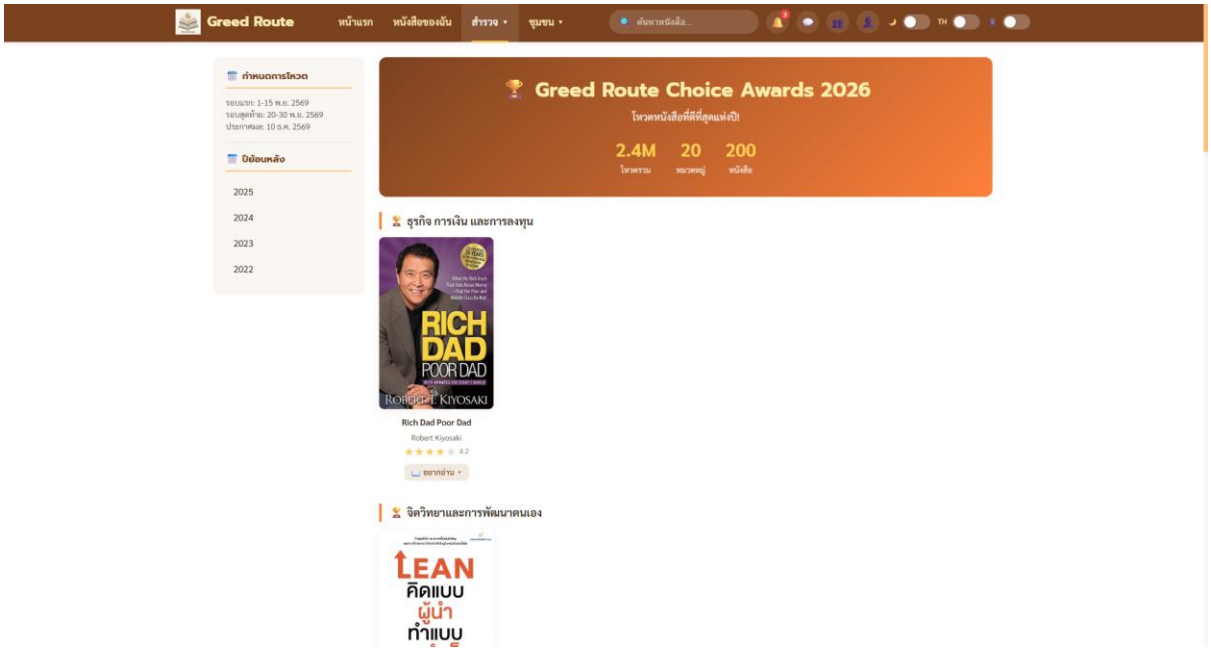
4. หน้าอื่นๆ

นอกจากหน้าหลักที่มี Recommender System เพื่อแนะนำหนังสือแล้ว ยังมีหน้าอื่นๆที่สร้างประสบการณ์ที่ดีให้แก่ผู้ใช้ เช่น

หน้า “หนังสือของฉัน”



หน้า “ฟีดข่าว”



หน้า “ชุมชน” และ “Community Choice Awards”