1 Scope

The system is aiming at providing a "Daily Ten" product recommendation in mobile stream for Gen-Z lifestyle shoppers in the Asia Pacific region. A report from Business Insider notes that although Gen Z consumers' shopping journeys often starts with Amazon they quickly drift toward Google's discovery surfaces like YouTube and Gemini, revealing that the need for concise, inspiration-driven discovery are more than the keyword search

The target user will be 18-28-year-old consumers who use Amazon more than (including) 3 times a week and browse over 20 items before making a purchase. Their general pain points are choice overload, caused by overwhelming product choices, and a lack of credible recommendations, which makes them unsure whose advice to follow.

The interaction design will focus on delivering a smooth discovery flow that displays a fresh "Daily Ten" stream every morning. Every morning in 6.A.M, the app refreshes 10 personalised items (max one per brand) and displays Pinterest-style in a masonry layout (max one per brand) on the home page of the application when the user starts the app. User just needs to scroll down to view more items and tap to open the product on amazon. Moreover, if the user takes a long press, the product card will flip and bring up quick actions like 'Save', 'Hide' etc., The explicit feedback, such as Save and clicking into product will be updated in real time to re-ranking the recommendations, while other implicit signals, like scroll dwell time, will be packed and updated in night with several batches for model updates.

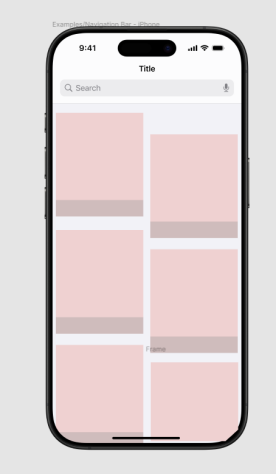

Figure 1: UI example

The cold start will be handled in two combined methods. Firstly, there will be a ten-question onboarding quiz for new users to acquire users' tastes so that the system can check and recommend what people with similar interests have bought or saved. Then, the system will rearrange the list in real time using action made by new users. Secondly, for any new items without any interactions, the system will decide where they should be recommended based on the text descriptions and photos until it has enough information on different users' feedback.

For a business model, it could refer to Amazon's affiliate marketing programs, which provide a 3% - 5% commission revenue with optional sponsored slots capped at 20 % of the grid to protect trust. For ethics and safety issues, all experimentation will follow the dataset's non-commercial licence. There will be no personally identifiable information stored, and an ethics checklist will be reviewed for bias and privacy safeguards.

2 Dataset

The selected dataset will be Amazon Reviews dataset collected in 2023(Amazon Reviews '23), the latest large-scale corpus released by McAuley Lab in March 2024. The public dump contains around 571 million user–item interactions, including ratings and text messages, from May 1996 to September 2023. It covers 54 million users, 48 million products and 33 high-level categories. Further, it contains second-level timestamps, rich item metadata and an official 5-Core train/valid/test split for RecSys benchmarking. Therefore, 110 million rows data for electronics and clothing is enough to build a good recommendation model.

Files	Key Features Kept	How to Apply
*.jsonl.gz	user_id, parent_asin, rating, reviewText, unix_timestamp	Implicit feedback matrix (if rating $\geq 4 \rightarrow$ positive recommendation). Sequence modelling for SASRec. Sentiment and topic features via BERT on reviewText.
meta_*.jsonl.gz	title, brand, description, price, image_urls, also_bought / also_viewed	Cold-start item embeddings from BERT (text) and CLIP (image). Graph edges (also_bought/also_viewed) for LightGCN regularizer. Price buckets, categorical feature, for diversity filtering.
5-Core split files	user/item index mapping	Ensures each user and item has at least 5 interactions to reduce sparsity in early-stage prototyping.

Table 1: Dataset Files and Their Applications

The mapping above shows that every element is planned to be modelled from the user sequence to cold-start item embeddings.

For feasibility, it will be too large to process the full 140 GB corpus for this project. Therefore, there are two popular lifestyles with target users will be picked, Electronics (around 44 million rows) and Clothing (around 66 million rows). After the conversion, it will be small enough to train and test comfortably in Colab

There will be a few noticeable limitations for this dataset. Firstly, a subset bias leading to long-tail items should be considered, since the dataset only keeps the reviews users chose to post and is heavy on best-sellers. It will therefore receive far fewer interactions, which can distort both training and evaluation. Secondly, the lack of 1-star and 2-star reviews inflates headline accuracy numbers and makes the system harder to distinguish true dissatisfaction from missing data. Thirdly, the objectionable or spam reviews are filtered out, and it ignores the diversity situation that real-world systems need to handle. Lastly, the fixed benchmarking split are static, which might lead to over-optimising to a snapshot

3 Methods

To fitting the modelling power and time duration, the methods are set to three stages that become more complicated along with different ones. Each stage will improve the capability demanded by the 'Daily-Ten' scenario.
For each stage:

- **Stage 1: Baseline model**
  For stage 1, the implicit-feedback matrix-factorisation model is chosen as the project baseline model. This is a classic collaborative filtering approach, which will record all ratings of four or five stars as 'like' feedback. The model will factorise the gained matrix into 128-dimensional latent vectors and learns them with Bayesian Personalised Ranking (BPR) loss. This model is easy to train on the 110 million rows acquired from the dataset and provides a clear starting point to compare with next stage's model. It will work properly since the dataset is large enough while each user will only click on a few things. Moreover, our target user would like to do more simple 'like or unlike' rather than giving star ratings.

- **Stage 2: Sequential recommender**
  For stage 2, the model will replace plain MF with a lightweight self-attention model(a SASRec-style architecture). It will scan what users have clicked in the past 3 months from the provided exact time stamps in the dataset and pays more attention to their latest taps, which will then lead to further prediction. As long as there is enough shopping data from target users, the next day's Daily Ten should work properly.

- **Stage 3: Combined**
  For the final stage, the model will focus on the cold-start problem for new products and new users. The model will mix what each product looks like and says. It will read the product's review text and picture, turn them into numbers, and stick them together. Then, the model will mix users' recent browsing patterns from Stage 2 with their long-term tastes from the baseline model. That way, even when a brand-new product with zero reviews or a new user without any interactions, the system still has reasonable recommendations of whether it belongs in next day's "Daily Ten".

4 Evaluation

The evaluation metric will be separated into two parts: metrics for models and metrics for the recommender system.

- Model metrics
  - **Recall@10(most important)** It will measure how often the true-positive item appears in the Daily Ten recommendation stream. It will represent whether the recommendation fit the user's taste/need.

  - **Coverage@10** It will check if the model pushed enough different products to different users. To prevent the model from recommending a long-tailed product to most users. If it has a higher number, this means that more unpopular products are presented to users instead of always presenting the popular ones.
  - **NDCG@10** It will measure not only if the user like the product, but also reward the prediction arrange the correct item near the top, not just anywhere in the list.

- System-level metrics
  - **Click-through rate(most important)** It will calculate the percentage of users click the product from the daily ten products. It directly measure whether the recommendation is attractive to the user.
  - **Save rate** It will measure the percentage of products that users saved from the daily ten products
  - **Average dwell time** It will calculate the average time the user will stay on the product lists.

The user study to evaluate the recommender system:

- 30-35 volunteers aged from 18-28 should be recruited on campus
- Two mocked app platform page, one is from the baseline model, while another one is from the combined model.
- Each participant should complete two 5-minute browsing tasks (electronics, then clothing).
- Taking the questionnaire after experiencing two models' recommendations, It includes
  - "How do these recommended products fit your taste?"
  - "How do you feel about the variety of the recommended products? Is it interesting?"
  - "How do you trust the recommended products?"
  - "Did you feel more stressed or relaxed when you read the recommendation list?"
  - "Overall, how do you feel about this recommendation? Do you like it?"