FASHION RECOMMENDATION SYSTEM BASED ON PERSONAL CHARACTERISTICS

Seonghoon Lee[*], **Taehun Kim**[*]
Computer Science and Engineering
Pusan National University
Busan, Repulic of Korea
`p.plue1881@gmail.com, bigteach0508@pusan.ac.kr`

ABSTRACT

This research focuses on developing a personalized fashion recommendation system based on individual physical characteristics. In modern society, fashion serves as a crucial means of self-expression, yet finding suitable styles that complement one's physical attributes remains challenging. The system analyzes three key physical characteristics-personal color, face shape, and body type-to provide personalized fashion recommendations. The methodology includes image analysis using deep learning techniques to automatically extract users' physical characteristics from uploaded photos. For personal color determination, the system analyzes skin color values using K-means clustering to classify users into four seasonal types (Spring, Summer, Autumn, Winter). Face shape classification uses an EfficientNetB4-based model trained on a data set of 5000 images, achieving 75. 83% test precision in identifying five face shapes. Body type measurement utilizes MediaPipe's Pose Landmark feature to extract key points and calculate proportions between different body parts. Two recommendation approaches were implemented: a Random Forest-based system that predicts ratings based on clothing attributes and extracted physical characteristics (achieving test RMSE of 0.759 for men and 0.773 for women), and a content-based collaborative filtering system using cosine similarity to recommend similar items based on keyword of clothes. The Random Forest-based system also incorporates user feedback to continuously improve recommendation quality. The research demonstrates the feasibility and effectiveness of personalized fashion recommendations based on physical characteristics, contributing to the growing demand for individualized services. Future work includes enhancing the system with more diverse data, real-time feedback incorporation, and integration with e-commerce platforms. We upload the source code on `https://github.com/pnucse-capstone-2024/Capstone-2024-team-24`

***K*eywords** Personal color · Recommendation System · Machine learning

1 Introduction

Fashion serves as an important means of expressing individual personality in modern society. However, finding suitable fashion for everyone is not easy. In particular, personalized fashion recommendation systems based on physical characteristics such as personal color, face shape, and body type are not yet common, which makes it difficult for users to easily find fashion that suits them. Recently, demand for personalized services has been increasing, and based on this, it has become important to develop fashion recommendation systems that consider physical characteristics.

Existing fashion recommendation systems often make recommendations based mainly on trends or popular items, and frequently do not sufficiently consider individual physical characteristics. This fails to provide optimal recommendations to users, and users are likely to receive recommendations for clothes that do not suit them.

The goal of this research is to develop a fashion recommendation system based on users' physical characteristics such as personal color, face shape, and body type. Through this, the objective is to reduce the difficulty of fashion selection and increase user satisfaction by recommending customized fashion to users.

[*]Equal contribution.

2 Background

2.1 The Importance of Personal Color, Face Shape, and Body Type

Personal color refers to an individual's innate physical coloration (skin, hair, eyes, etc.) and is an important factor that influences one's overall appearance image. According to the personal color system proposed by Carol Jackson, human coloration is broadly categorized into four types: Spring, Summer, Autumn, and Winter. When wearing clothing that matches one's personal color, the individual's skin appears more vibrant and the overall impression becomes bright and attractive. Conversely, wearing colors that don't suit well can make the face appear pale or the impression dull, making it important to choose fashion based on personal color [1].

Facial shape varies according to each person's unique bone structure, skin tissue, and facial muscles. It is commonly classified into heart-shaped, long, oval, round, square, and other types. Selecting clothing, accessories, and hairstyles that suit one's facial shape is very important for balancing facial features and maximizing attractiveness. For example, for long facial shapes, wide lapels or oversized clothing are more suitable, while round facial shapes benefit from longer flowing styles or V-neck designs that effectively improve one's impression [2]. Therefore, a system that accurately analyzes facial shape and recommends customized fashion can be very useful.

Body shape refers to the overall structure and proportions of the body, which varies greatly according to each individual's skeletal frame, muscle distribution, and fat ratio. Men are mainly classified into triangular, inverted triangular, rectangular, oval, trapezoidal, and other body shapes, while women are classified into hourglass, inverted triangular, rectangular, triangular, round, and other shapes. Selecting clothing that suits one's body shape enables body shape enhancement and can boost confidence in styling. For example, for inverted triangular body shapes, designs that emphasize the waist are suitable, while round body shapes benefit from flare skirts or V-neck tops that help complement the body shape [3]. A system that recommends clothing suited to body shape can significantly increase customer satisfaction.

Thus, a system that recommends fashion based on personal color, facial shape, and body shape can be a powerful tool that goes beyond mere external beauty to enhance individual self-esteem and boost confidence. Particularly, a system that properly reflects these elements can provide high satisfaction by suggesting customized fashion styles to users.

2.2 Increasing Demand for Personalized Services and Limitations of Fashion Recommendation Systems

In recent years, the global demand for personalized services has surged. Consumers now seek not just to purchase products, but to find items and experiences tailored precisely to their preferences. This trend is clearly reflected in the fashion industry, where many brands are focusing on providing personalized shopping experiences. According to the 2023 Consumer Trend Report, consumers are highly interested in expressing their individuality and finding products that suit them perfectly, with particularly strong demand for personalized services among consumers in their 20s and 30s [4].

However, current fashion recommendation systems often rely on popular trends or users' purchase histories. These systems typically do not adequately consider individual physical characteristics—such as personal color, face shape, or body type—which limits their ability to recommend truly suitable fashion items. Additionally, many of these systems fail to effectively incorporate user feedback. Even when users provide ratings or reviews, this information is often not properly utilized to improve recommendation quality. Continuously integrating user feedback is essential for enhancing algorithm accuracy and is a key factor in increasing both user trust and satisfaction.

To address these limitations, this study aims to develop a personalized fashion recommendation system that reflects users' physical characteristics. By allowing users to upload photos or input their physical information, the system will automatically analyze this data and recommend fashion items tailored to each individual's traits, thereby overcoming the shortcomings of existing systems.

3 Architecture

3.1 System Architecture

3.1.1 User input and data processing

Users can provide their physical characteristics through two methods. First, they can upload photos of their face or full body, and second, they can directly input their physical information. When using the photo upload method, the user's physical characteristics (personal color, face shape, body type) are automatically extracted through a deep learning-based image analysis model.

During this process, the user's photos are transmitted to the backend server via HTTP protocol. The data is transmitted in JSON format or image file format, and the photos uploaded by users are not stored separately. This enhances the protection of users' personal information.

3.1.2 Data preprocessing and analyzing

The photos uploaded by users first undergo an image preprocessing process. In the preprocessing stage, unnecessary parts of the face or body are removed, and only the necessary parts are extracted for analysis. For this purpose, We use OpenCV, dlib, and MediaPipe to accurately detect key parts of the face and body type, and based on this, skin area, face shape, and body type information are extracted.

3.2 Personal Color Measurement

3.2.1 Data Preprocessing

We use Korean facial images from AIHub[5], consisting of approximately 30,000 images. Among these, only frontal photos were used, utilizing about 5,000 images. The skin regions of faces were extracted from each image to conduct accurate personal color measurements.

To extract only the skin regions from faces, deep learning-based face detection and segmentation models were applied. We use the retinaface/mobilenet model from the facer library to detect faces, followed by the farl/lapa/448 model to segment the faces. In this process, to accurately distinguish skin regions, each facial part (eyes, nose, mouth, forehead, skin, etc.) was segmented into probability maps, and only the skin portions were selected.

3.2.2 Skin Region RGB Value Extraction

A binary mask corresponding to the skin portion was generated from the segmented probability maps. This mask enables the selection of only skin region pixels, through which RGB values of the skin were extracted from the images. To remove noise and minimize data distortion, the extracted RGB values were filtered to use only values between the first quartile (Q1) and third quartile (Q3), deriving a filtered average value.

3.2.3 Color Space Conversion and Personal Color Classification

The extracted average RGB values were converted to HSV and Lab color spaces. Specifically, the V (Value) and S (Saturation) values from the HSV color space and the b-channel values from the Lab color space were utilized for analysis.

Subsequently, We use a K-means clustering algorithm to build a model that predicts users' personal colors based on the extracted skin tones. This model was designed with reference to [6], using the method of categorizing into spring, summer, autumn, and winter based on the brightness, saturation, and yellowness of representative skin colors for each season as presented in that paper.

Table 1: Skin Color Characteristics by Personal Color Type

Skin Color			
Spring	Summer	Autumn	Winter
High brightness, High saturation, High yellowness	High brightness, Low saturation, Low yellowness	Low brightness, Low saturation, High yellowness	Low brightness, High saturation, Low yellowness

3.3 Face shape Measurement

3.3.1 Data Preprocessing

We use the Face Shape Dataset from Kaggle[7], consisting of a total of 5,000 images. It includes 800 training images and 200 test images for each face shape (heart, long, oval, round, square). These images were processed to extract only the facial regions using the CascadeClassifier provided by OpenCV, and the image size was adjusted to 380x380. This is because EfficientNetB4 has an input shape of 380x380. Since 5,000 images were deemed insufficient for training, data augmentation was performed by adding horizontally flipped versions of the original photos and images rotated within a (-20, 20) degree range.

Additionally, since the pixel values of the photos are in the 0 255 range, all images were normalized based on the R, G, B mean values and standard deviations as follows:

$$p_i = \frac{p_i - mean_i}{std_i}, i \in R, G, B$$

where p_i, $mean_i$, std_i are one of the i-th color pixels, the mean of all photos for the i-th color, and the standard deviation of all photos for the i-th color, respectively.

3.3.2 Model Architecture

The model configuration was based on the EfficientNetB4 network pre-trained on ImageNet data. GlobalAveragePooling2D was applied and set to output from EfficientNetB4, and the output values from EfficientNetB4 were classified into 5 face shapes through Dense layers.

3.4 Body Shape Measurement

3.4.1 Extracting Keypoint

Body measurement is performed by extracting KeyPoint coordinates of body parts using the Pose Landmark feature of the MediaPipe library. In this process, coordinates of shoulders, hips, knees, and other parts are extracted, and the waist position is predicted by calculating ratios for each body part. Using Korean body data, waist-to-height ratios relative to each body measurement were calculated, and based on this, the waist position was estimated.

3.4.2 Waist Position Estimation

We use data from Size Korea[8], which compiled Korean body data. This data contains information on shoulder height, hip height, waist height, knee height, etc., organized by age and gender of Koreans. Since the data is organized by body type, the average shoulder height, hip height, waist height, and knee height for each body type were calculated. Various ratios were computed including waist height relative to shoulder height, waist height relative to hip height, waist height relative to knee height, waist height relative to (shoulder height + hip height), and waist height relative to (shoulder height + hip height + knee height) to determine which method is most suitable for calculating waist height.

Table 2: Waist Height Ratios by Age Group for Male Body Measurements

Unit: mm, Mean (Standard Deviation)	**20s Male**	**30s Male**	**40s Male**	**50s Male**	**60s Male**
Mean Shoulder Height	1398.67	1385.75	1364	1348.75	1339
Mean Hip Height	849.33	834.5	819.25	805	804
Mean Waist Height	1070.33	1054.24	1033	1024.25	1019.75
Mean Knee Height	443	438.25	431.5	422.25	424.5
Waist Height/Shoulder Height	0.765 (0.00042)	0.761 (0.00305)	0.757 (0.00648)	0.759 (0.01437)	0.756 (0.01389)
Waist Height/Hip Height	1.26 (0.00393)	1.263 (0.0096)	1.261 (0.01699)	1.273 (0.04887)	1.261 (0.04218)
Waist Height/Knee Height	2.416 (0.01572)	2.406 (0.02015)	2.395 (0.0525)	2.427 (0.08395)	2.388 (0.09247)
Waist Height/(Shoulder+ Hip Height)	0.476 (0.00072)	0.475 (0.0023)	0.473 (0.00475)	0.476 (0.01182)	0.473 (0.01073)
Waist Height/(Shoulder+ Hip Height+Knee Height)	0.398 (0.00091)	0.397 (0.00202)	0.395 (0.00471)	0.398 (0.01047)	0.395 (0.00995)

It was decided to predict the waist position using the average waist height relative to (shoulder height + hip height), which had the relatively lowest standard deviation. The process of determining waist height is as follows:

1. Obtain the coordinates of the keypoints for shoulders, hips, and toes.
2. Calculate shoulder height by finding (toe keypoint's y-coordinate - shoulder keypoint's y-coordinate).
3. Calculate hip height by finding (toe keypoint's y-coordinate - hip keypoint's y-coordinate).
4. Obtain the waist y-coordinate by multiplying the ratio derived from Table 2 and Table 3 with (shoulder height + hip height).

3.4.3 Chest Position Estimation

Since there was insufficient data on the average chest height of Koreans, the chest position was estimated by moving approximately 10% of the shoulder height downward from the shoulder coordinates.

Table 3: Waist Height Ratios by Age Group for Female Body Measurements

Unit: mm, Mean (Standard Deviation)	20s Female	30s Female	40s Female	50s Female	60s Female
Mean Shoulder Height	1289.5	1268.25	1260	1249.25	1223
Mean Hip Height	778.75	756.75	748.25	744.56	729.5
Mean Waist Height	995	971.75	963.25	951.31	927.5
Mean Knee Height	407.75	398	397.25	393.81	386.5
Waist Height/Shoulder Height	0.772 (0.00317)	0.766 (0.00555)	0.764 (0.00509)	0.76 (0.00733)	0.758 (0.00971)
Waist Height/Hip Height	1.278 (0.00395)	1.284 (0.00729)	1.287 (0.00729)	1.274 (0.01698)	1.271 (0.01685)
Waist Height/Knee Height	2.44 (0.00729)	2.442 (0.0117)	2.425 (0.00835)	2.413 (0.00431)	2.4 (0.02996)
Waist Height/(Shoulder+ Hip Height)	0.481 (0.00092)	0.48 (0.00219)	0.48 (0.00236)	0.476 (0.00504)	0.475 (0.00583)
Waist Height/(Shoulder+ Hip Height+Knee Height)	0.402 (0.00071)	0.401 (0.0018)	0.4 (0.00151)	0.398 (0.00343)	0.397 (0.00482)

3.4.4 Background Removal and Length Measurement

Initially, We utilized canny Edge Detection to find the boundary between the body and background, but measurement became difficult due to patterns on clothing and other factors, so we use the Rembg background removal library to remove the background. After background removal, body measurements were taken using coordinates of body parts measured with Mediapipe. Since arm width would be included in waist and chest width measurements when arms are positioned close to the body, measurements were taken with arms spread apart as shown in Figure 2. Figure 2 utilized the [9] dataset.

Figure 1: Shoulder, chest, waist, and hip width measurements. We utilized the [9] dataset for evaluation of the body shape measurement model

3.5 Recommendation System based on RandomForest

3.5.1 Data Preprocessing

We use the 2019 dataset of the AI-Hub's "Fashion Preference Analysis and Recommendation Data by Year."[10] This dataset consists of clothing-worn images and evaluation data in JSON format where respondents assessed these images. The evaluation data is composed of respondent ID, evaluated image name, fashion style, image gender, evaluation details, and respondent information. Each user's preferred fashion styles are organized as Fashion Styles 1-5 as shown in Table 4 below, with users selecting one from each fashion style category for a total of 5 preferred fashion styles.

Table 4: Respondents' Preferred Fashion Style Selection

Fashion Style	**Fashion Style 1**	**Fashion Style 2**	**Fashion Style 3**	**Fashion Style 4**	**Fashion Style 5**
1	Flashy and Unique	Masculine/Feminine	Traditional	Formal	Active
2	Modest and Normal	Androgynous	Trendy	Casual	Calm
Note	**1, 2 Choose 1**	**1, 2 Choose 1**	**1, 2 Choose 1**	**1, 2 Choose 1**	**1, 2 Choose 1**

During the data preprocessing phase, JSON data was converted to CSV format, separating user information, clothing information, and evaluation information. When multiple users evaluated the same clothing item, the final evaluation for that item was determined by adopting the mode (most frequent value). Subsequently, personal color, face shape, and body type were inferred based on users' preferred fashion styles. Using the table 5 and 6, personal color and physical characteristics were assigned for each style, and the characteristics with the highest frequency were selected as the final choice.

Table 5: Assignment of Personal Color, Face Shape, and Body Type by Preferred Fashion Style for Male Respondents

Preferred Fashion Style	**Personal Color**	**Face Shape**	**Body Type**
Flashy and Unique	Spring, Winter	Oval, Heart	Inverted Triangle, Triangle
Modest and Normal	Summer, Autumn	Round, Square	Round, Rectangle
Masculine	Autumn, Winter	Oblong, Square	Inverted Triangle, Triangle
Androgynous	Spring, Summer	Oval	Rectangle
Traditional	Autumn, Winter	Round, Rectangle, Oblong	Triangle, Oval
Trendy	Spring, Summer	Heart, Oval	Rectangle, Round
Formal	Autumn, Winter	Oblong, Square	Inverted Triangle, Triangle
Casual	Spring, Summer	Round, Heart	Round, Rectangle
Active	Spring, Summer	Round, Heart	Rectangle, Inverted Triangle
Calm	Autumn, Winter	Square, Oval, Oblong	Triangle, Rectangle

Table 6: Assignment of Personal Color, Face Shape, and Body Type by Preferred Fashion Style for Female Respondents

Preferred Fashion Style	**Personal Color**	**Face Shape**	**Body Type**
Flashy and Unique	Spring, Winter	Oval, Heart	Inverted Triangle, Hourglass
Modest and Normal	Summer, Autumn	Round, Square	Round, Rectangle
Masculine	Autumn, Winter	Oblong, Square	Hourglass
Androgynous	Spring, Summer	Oval	Rectangle
Traditional	Autumn, Winter	Round, Square, Oblong	Rectangle, Round
Trendy	Spring, Summer	Heart, Oval	Inverted Triangle, Triangle
Formal	Autumn, Winter	Oblong, Square	Rectangle, Hourglass
Casual	Spring, Summer	Round, Heart	Round, Triangle
Active	Spring, Summer	Round, Heart	Triangle, Inverted Triangle
Calm	Autumn, Winter	Square, Oval, Oblong	Hourglass, Rectangle

3.5.2 Machine Learning using RandomForest

The data used for training consisted of user characteristics (personal color, face shape, body type) and clothing characteristics as features, with ratings as the target variable. The data was preprocessed using One-Hot Encoding, and trained using Scikit-Learn's RandomForestRegressor. The RMSE results showed that for male data, the training data achieved 0.304 and test data achieved 0.759, while for female data, the training data achieved 0.312 and test data achieved 0.773.

Table 7: Training Results Using RandomForest

Gender/Data	Training Data	Test Data
Male	0.304	0.759
Female	0.312	0.773

Using the created model to predict ratings for user characteristics - clothing characteristics and make recommendations in order of highest ratings yields the results shown in Table 8.

Table 8: Rating Prediction Results for 20s Male, Spring Personal Color, Round Face Shape, and Round Body Type

Clothing Item	Predicted Rating (Out of 4)
W_81578_19_normcore_M.jpg	3.780000
W_63901_19_normcore_M.jpg	3.770000
W_54264_19_normcore_M.jpg	3.760000
T_17451_19_normcore_M.jpg	3.760000
W_81468_19_normcore_M.jpg	3.750000
W_01418_19_normcore_M.jpg	3.743333
T_19532_19_normcore_M.jpg	3.740000
T_17134_19_normcore_M.jpg	3.700000
W_31826_19_normcore_M.jpg	3.700000
T_17087_19_normcore_M.jpg	3.683333

3.5.3 Feedback on Recommendation Quality

We implemented a system to provide feedback on recommendation quality as described in the advisory report from Naver Webtoon. The implementation method was constructed as follows using the RandomForest model from section 3.5.2:

1. When making recommendations, the final rating is calculated by combining the model's predicted rating with the actual user's evaluation rating. This approach reflects 70% of the model's predicted rating and 30% of the actual user's rating.

2. When a user provides a star rating, that data is added to the existing training data, and a new RandomForest model is trained. The ratings given by users are recorded individually for each user and reflected in the actual user ratings.

When the results from Table 8 are combined with actual user evaluation ratings to calculate the final ratings, it becomes Table 9.

Table 9: Fashion Recommendation based on Rating Prediction Results and Actual users' rating for 20s Male, Spring Personal Color, Round Face Shape, and Round Body Type

Clothing Item	Predicted Rating	User Rating	Final Rating
W_81578_19_normcore_M.jpg	3.78	4	3.846
W_63901_19_normcore_M.jpg	3.77	4	3.839
T_17451_19_normcore_M.jpg	3.76	4	3.832
W_54264_19_normcore_M.jpg	3.76	4	3.832
W_81468_19_normcore_M.jpg	3.75	4	3.825
T_19532_19_normcore_M.jpg	3.74	4	3.818
W_31826_19_normcore_M.jpg	3.7	4	3.79
T_17134_19_normcore_M.jpg	3.7	4	3.79
T_17087_19_normcore_M.jpg	3.6833	4	3.778
T_19542_19_normcore_M.jpg	3.68	4	3.776

Now, when the clothing item most recommended by the recommendation system in Table 9 is rated with 1 point and recommendations are made again, it becomes Table 10.

Table 10: Recommendation Results After Feedback for 20s Male, Spring Personal Color, Round Face Shape, and Round Body Type

Clothing Item	Predicted Rating	User Rating	Final Rating
W_54264_19_normcore_M.jpg	3.77	4	3.839
T_17451_19_normcore_M.jpg	3.77	4	3.839
W_63901_19_normcore_M.jpg	3.76	4	3.832
W_81468_19_normcore_M.jpg	3.76	4	3.832
W_31826_19_normcore_M.jpg	3.705	4	3.7935
T_17134_19_normcore_M.jpg	3.7	4	3.79
W_17682_19_normcore_M.jpg	3.68	4	3.776
W_52075_19_normcore_M.jpg	3.67	4	3.769
W_25216_19_normcore_M.jpg	3.67	4	3.769
W_92952_19_normcore_M.jpg	3.67	4	3.769

3.6 Recommendation System based on Content-based Collaborative Filtering

3.6.1 Data Preprocessing

We use K-Fashion images dataset from AI-hub[11]. This dataset consists of 1.2 million clothing photos and their corresponding characteristics organized in JSON files. These were parsed to categorize into tops, bottoms, outerwear, dresses, and other categories, and organized into CSV files.

3.6.2 TF-IDF Vectorizing

TF-IDF Vectorizing is a technique that represents the importance of keywords. Each clothing characteristic was combined into a single string, then vectorized using Scikit-Learn's TF-IDF Vectorizer. L2 regularization was applied during this process to prevent overfitting.

3.6.3 Cosine Similarity Calculation and Clothing Feature Association

Cosine similarity is a similarity measure calculated using the cosine of the angle between two vectors in inner product space. If $X = (x_1, \ldots, x_n)$ and $Y = (y_1, \ldots, y_n)$ are two document vectors, cosine similarity can be calculated as follows:

$$Cosine(X, Y) = \frac{\sum x_i y_i}{\sqrt{\sum x_i^2}\sqrt{\sum y_i^2}}$$

We use the cosine_similarity function from Scikit-Learn to calculate the similarity between vectorized clothing features, and items with high similarity were recommended in order. After this process, clothing features that fit the user's personal color, face shape, and body type were associated. For example, a round neck was added to the properties of the matching image for the square face.

Table 11: Results of 'Khaki Casual Stripe Sport T-shirt Slim Graphic Oversized Round Round-neck' Search

Product ID	Category	Description	Similarity
1038491	Shirts	Khaki casual stripe sport t-shirt slim graphic oversized round	1
37620	Shirts	Khaki casual white stripe t-shirt slim graphic oversized round	0.8614
1036139	Shirts	Khaki casual stripe sport t-shirt loose graphic round round-neck	0.8309
190270	Shirts	Stripe sport t-shirt slim graphic oversized round	0.8276
1213067	Shirts	Khaki casual stripe sport t-shirt double-sleeved slim graphic oversized	0.8208
735440	Shirts	Black casual white stripe sport t-shirt slim graphic oversized round round-neck	0.8035
735430	Shirts	Black casual white stripe sport t-shirt slim graphic oversized round round-neck	0.8035
735410	Shirts	Black casual white stripe sport t-shirt slim graphic oversized round round-neck	0.8035
37630	Shirts	Khaki casual white stripe t-shirt slim graphic oversized round round-neck	0.7719
397965	Shirts	Khaki casual black stripe t-shirt double-sleeved slim retro graphic oversized round...	0.7459

4 Results

4.1 Personal Color Measurement Model

Personal color measurement was classified into four seasons (spring, summer, autumn, and winter) using K-means clustering based on the user's skin color. Personal color was classified by using HSV (Value, saturation) and Lab (b channel) values of skin color according to each season, and as shown in the figure, characteristic color values were derived from each cluster.

Figure 2 is a graph that visualizes the main color characteristics of each season (cluster). This graph represents the average of the values of Value, Saturation, and b channel (yellow-blue axis).

The spring type is characterized by warm shades with high brightness and saturation values. The summer type has a relatively low saturation value, so colors corresponding to cool tones mainly appeared. The autumn type is characterized by warm shades due to a high b channel value, and the winter type has a relatively low value and moderate saturation value, reflecting a cold color of low brightness but high saturation.

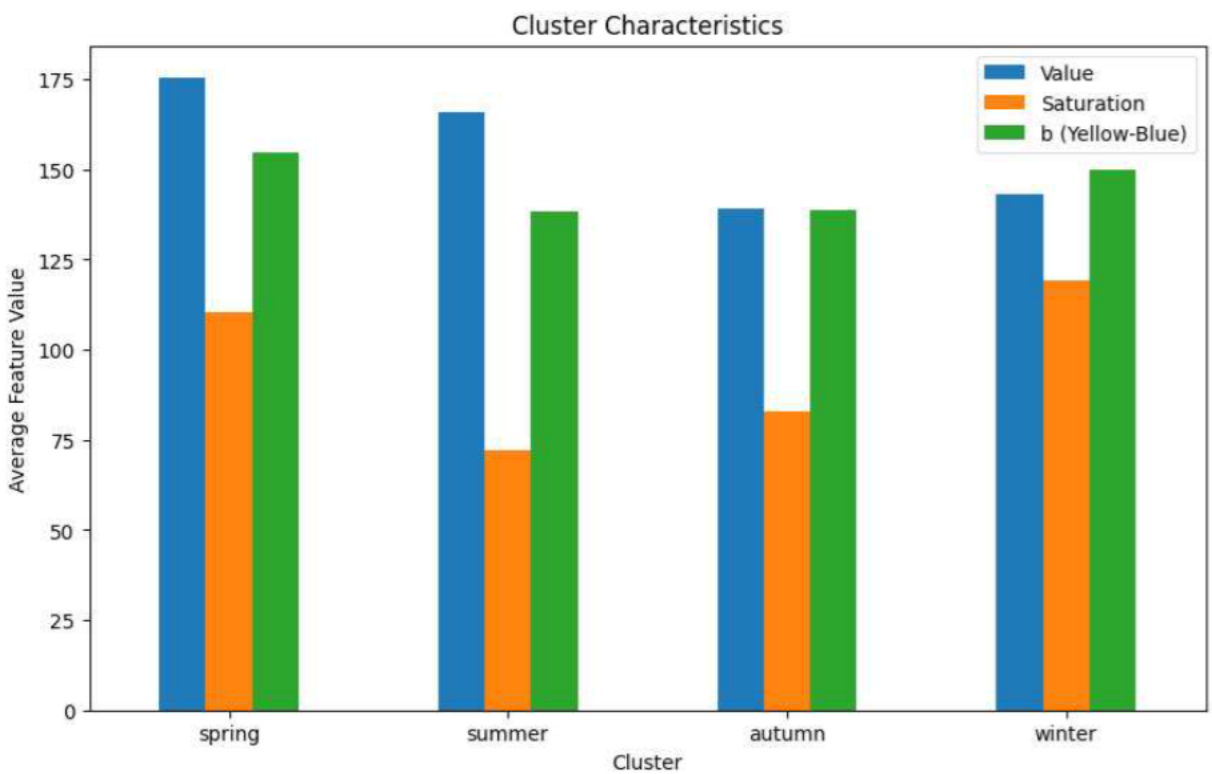

Figure 2: Personal Color Clustering

4.2 Face Shape Measurement

4.2.1 Fine-tuning with freezing backbone layers

The weights of the pre-trained EfficientNetB4 were kept frozen, and only the remaining parts were trained. The model was trained for 25 epochs using the Adam optimizer with a learning rate of 0.001. In the final epoch, the training accuracy reached 0.9851 and the test accuracy was 0.6961.

4.2.2 Fine-tuning without freezing layers

After training with freezing EfficientNetB4, the model was then trained with unfreezing EfficientNetB4. To prevent excessive modification of the previously learned weights, the learning rate was set to 0.0001. In the final epoch, the training accuracy reached 0.9993 and the test accuracy was 0.7513. EarlyStopping was applied with restore_best_weights=True, so the saved model achieved a training accuracy of 0.9994 and test accuracy of 0.7583.

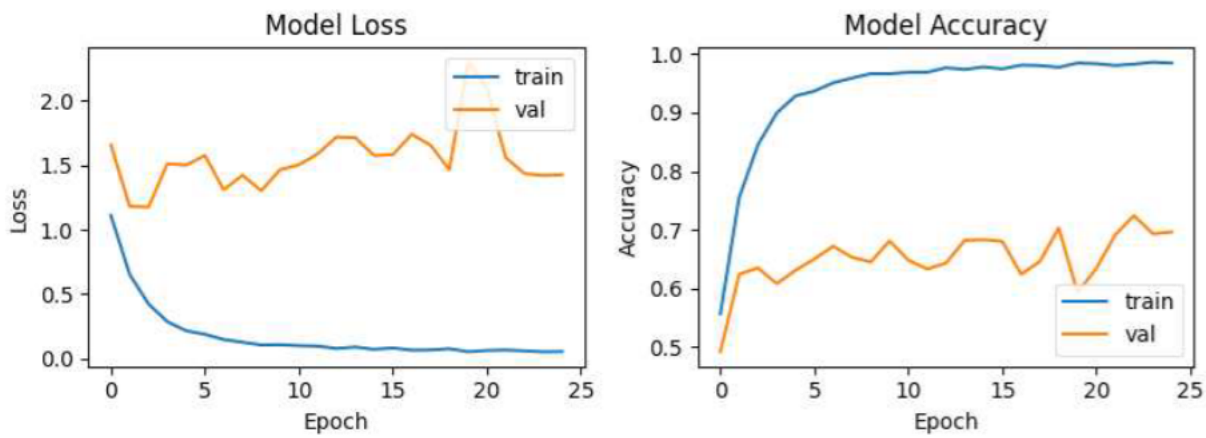

Figure 3: Fine-tuning with freezing backbone layers

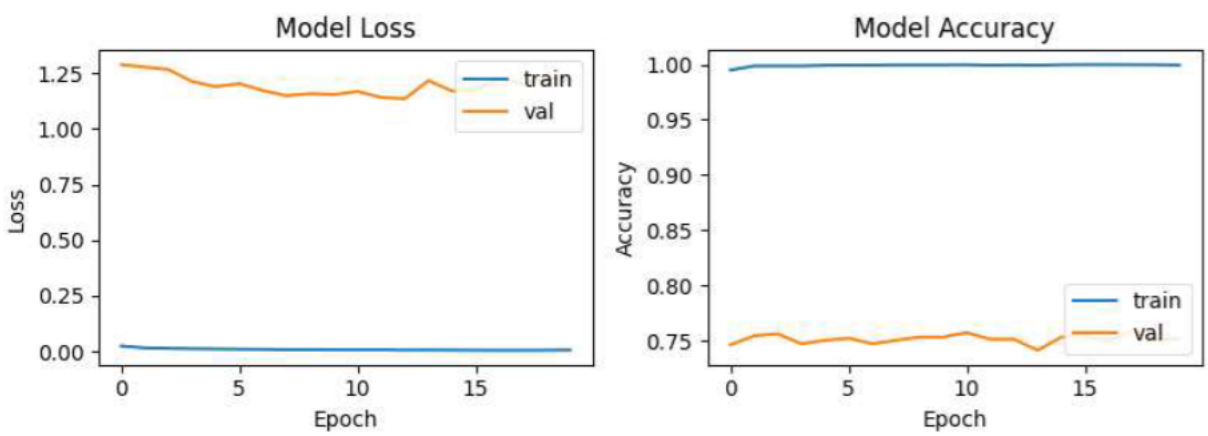

Figure 4: Fine-tuning with freezing backbone layers

4.3 Body Shape Measurement

According to the measured shoulder width, chest width, waist width, and hip width, they were classified as Table 12 and Table 13.

Table 12: Body Type Classification Criteria for Males

Criteria	Body Type
Shoulders > Chest and Chest > Hips and Waist > Chest	Inverted Triangle
Hips > Shoulders and Hips > Chest	Pear Shape
Shoulders > Chest and Shoulders > Waist	Inverted Triangle
Chest > Shoulders and Chest > Waist	Apple Shape
Others	Rectangle

4.4 Recommendation System

Based on the RandomForest learning results, the training RMSE for male data was 0.304 and the test RMSE was 0.759, while for female data, the training RMSE was 0.312 and the test RMSE was 0.773. This indicates that the recommendation model can accurately reflect users' physical characteristics and clothing features to recommend fashion items. Additionally, the content-based collaborative filtering recommendation system was effective in recommending similar clothing based on the textual characteristics of garments, enabling users to explore various clothing styles that suit them.

Table 13: Body Type Classification Criteria for Females

Criteria	Body Type
Shoulders > Hips and Chest > Hips	Inverted Triangle
Hips > Shoulders and Hips > Chest	Pear Shape
Shoulders > Chest and Shoulders > Waist	Inverted Triangle
Chest > Shoulders and Chest > Waist	Apple Shape
Others	Rectangle

5 Conclusion

We constructed a personalized fashion recommendation system based on users' physical characteristics including personal color, face shape, and body type, confirming the potential for personalized services. In particular, the RandomForest-based recommendation system achieved high prediction accuracy, and the recommendation quality improvement feature that reflects user feedback played a crucial role in continuously enhancing the system's performance. These results demonstrated the importance of user-customized services and the potential for how personalized recommendation systems can be applied in the fashion industry.

In the future, by securing diverse data and advancing recommendation systems that reflect real-time feedback, we will be able to provide better user experiences. Furthermore, if we build a fashion recommendation system that responds to users in a more personalized and real-time manner through advanced deep learning-based recommendation algorithms and fashion trend analysis, it is expected that the fashion industry will become more digitized and meet the demand for user-customized services.

Additionally, by establishing a connected system that allows direct purchase of recommended fashion items through integration with shopping malls, we can enhance user convenience and provide an integrated user experience where fashion recommendations go beyond simple style suggestions to actual clothing purchases.

Acknowledgments

This graduation project was conducted under the supervision of Professor Junsu Cho in School of Information and Computer Engineering at Pusan National University. The authors would like to express their deep gratitude for his guidance and support.

This work was conducted as part of the 2024 Early-Year Capstone Design Program of the School of Information and Computer Engineering. It was carried out as an industry-academic collaboration project, with partial guidance and advice provided by Kyung-Min Roh in NAVER Webtoon. This work has been registered with the Korea Copyright Commission (Registration No. C-2024-042740), with Research and Business Development Foundation of Pusan National University listed as the copyright holder. The author participated in the work as the creators.

Dataset [10], [11], [5] is Project Outcome by the National Information Society Agency (NIA)

References

[1] Yun-Seok Jung. A study on the quantitative diagnosis model of personal color. *Journal of Convergence for Information Technology*, (11):277 – 287, 2021.

[2] Jong-Suk An. A study on effective image making depending on hair style and neckline. *Jounal of The Korean Society of cosmetology*, 15(1):342–351, 2009.

[3] Soo ae Kwon. *Fashion and Life*. Gyohakyungusa, September 2016.

[4] MezzoMedia. 2023 consumption trend series - 03 personalized services. `https://www.mezzomedia.co.kr/data/insight_m_file/insight_m_file_1605.pdf`, 2023. Accessed: May 19, 2024.

[5] AIHub. Korean Face Image Dataset. `https://www.aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=83`, 2020. Accessed: July 26, 2025.

[6] So-young Lee. Personal color tone type and categorization of harmonious colors according to skin color. Master's thesis, Graduate School of Cultural and Information Policy, Hongik University, Seoul, South Korea, 2019. in Korean.

[7] Niten Lama. Face shape dataset. `https://www.kaggle.com/datasets/niten19/face-shape-dataset`, 2020.

[8] Size Korea. Human body information - body shape classification by age and gender. https://sizekorea.kr/human-info/body-shape-class/age-gender-body, 2024. Accessed: July 26, 2025.

[9] Cameron Patrick Trotter, Filipa Peleja, Alberto de Santos, and Dario Dotti. Human Body Shape Classification Dataset. "https://data.ncl.ac.uk/articles/dataset/Human_Body_Shape_Classification_Dataset/19307300", 5 2023.

[10] AIHub. fashion preferences and recommendation data by year. https://aihub.or.kr/aihubdata/data/view.do?dataSetSn=71446, 2022. Accessed: July 26, 2025.

[11] AIHub. K-Fashion Image. https://www.aihub.or.kr/aihubdata/data/view.do?dataSetSn=51, 2020. Accessed: July 26, 2025.

[12] Baran Bingl. Face shape detection(85% acc on test set). https://www.kaggle.com/code/baranbingl/face-shape-detection-85-acc-on-test-set, 2023. Accessed: July 26, 2025.

A Appendix - Towards Better Fashion Recommendation Systems

After the capstone project is finished, we evaluate the recommendation system more precisely.

We considered a score of 3 or higher to be preferred (or recommended) and a score of less than 3 to be disliked (or not recommended) based on the score the user gave to the clothes and the score predicted by the recommendation system.

Additionally, we reduce the face shape classification model size using EfficientNetB2.

We upload the code for model test and improved face shape classification model on the Github: https://github.com/minchoCoin/capstone-team-24-page/tree/main/appendix

Additionally, we upload the improved face shape classification code and model weights on the Kaggle: https://www.kaggle.com/code/minchocoin/tensorflow-efficientnetb2-test-accuracy-83percent

A.1 Precision and Recall

We measured the precision and recall using classification_report function in sklearn.metrics.

Table 14: Precision and recall

	Real positive	Real negative
Predicted positive	TP	FP
Predicted negative	FN	TN

$$\text{precision} = \frac{TP}{TP + FP}$$

$$\text{recall} = \frac{TP}{TP + FN}$$

$$\text{accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

A.1.1 Man

Table 15 and Table 16 shows the precision and recall of the recommendation system on the man dataset. 0 means not recommended or disliked, and 1 means recommended or preferred. These result show that the recommendation system has a high precision, which means that many of the recommended ones are preferred by the user.

Table 15: Precision and recall of the recommendation system on the man training dataset. 0 means not recommended or disliked, and 1 means recommended or preferred

	precision	**recall**	**f1-score**	**support**
0	0.65	1.00	0.79	6883
1	1.00	0.42	0.59	6224
accuracy			0.72	13107
macro avg	0.83	0.71	0.69	13107
weighted avg	0.82	0.72	0.69	13107

Table 16: Precision and recall of the recommendation system on the man test dataset. 0 means not recommended or disliked, and 1 means recommended or preferred

	precision	**recall**	**f1-score**	**support**
0	0.59	0.96	0.73	939
1	0.86	0.28	0.42	862
accuracy			0.63	1801
macro avg	0.72	0.62	0.58	1801
weighted avg	0.72	0.63	0.58	1801

A.1.2 Woman

Table 17 and Table 18 shows the precision and recall of the recommendation system on the woman dataset. 0 means not recommended or disliked, and 1 means recommended or preferred. These result show that the recommendation system has a high precision, which means that many of the recommended ones are preferred by the user.

	precision	recall	f1-score	support
0	0.58	1.00	0.73	7516
1	1.00	0.46	0.63	10027
accuracy			0.69	17543
macro avg	0.79	0.73	0.68	17543
weighted avg	0.82	0.69	0.67	17543

Table 17: Precision and recall of the recommendation system on the woman training dataset. 0 means not recommended or disliked, and 1 means recommended or preferred

Table 18: Precision and recall of the recommendation system on the woman test dataset. 0 means not recommended or disliked, and 1 means recommended or preferred

	precision	recall	f1-score	support
0	0.49	0.93	0.65	1033
1	0.86	0.30	0.44	1405
accuracy			0.57	2438
macro avg	0.67	0.61	0.54	2438
weighted avg	0.70	0.57	0.53	2438

A.2 Precision at k and Recall at k

We measured the precision at k and recall at k using users with two or more ratings. Precision at Top k measures the proportion of relevant items in the top K results returned by a system. Recall at Top k measures the proportion of all relevant items that are included in the top K results.

$$\text{Precision@k} = \frac{\text{Number of items relevant to the user in top K}}{K} \tag{1}$$

$$\text{Recall@k} = \frac{\text{Number of items relevant to the user in top K}}{\text{Total number of items relevant to the user}} \tag{2}$$

A.2.1 Man

Figure 5 and Figure 6 shows the precision at Top k and recall at Top k. Figure 7 and 8 shows the precision and recall curve.

A.2.2 Woman

Figure 9 and Figure 10 shows the precision at Top k and recall at Top k. Figure 11 and 12 shows the precision and recall curve on woman dataset.

A.3 Precision at k and Recall at k Evaluation Based on Personal Characteristic-Relevant Items

In this section, we measure precision at k and recall at k by comparing the clothing items actually preferred by users with the same personal characteristics (personal color, face shape, body type) and those predicted by the model to be

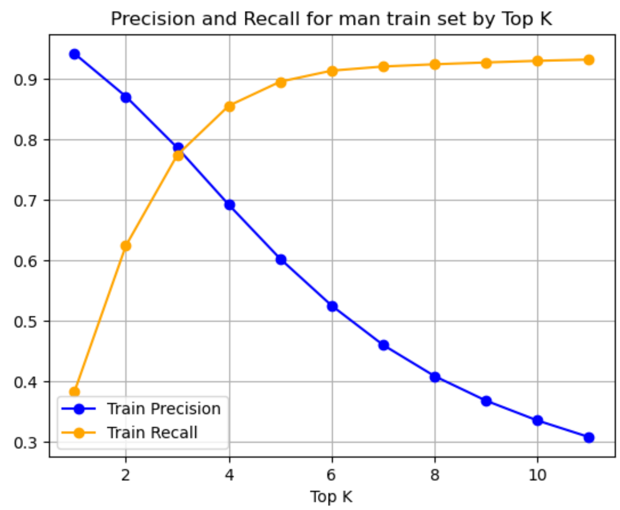

Figure 5: Precision at Top k and recall at Top k of the recommendation system on the man training dataset.

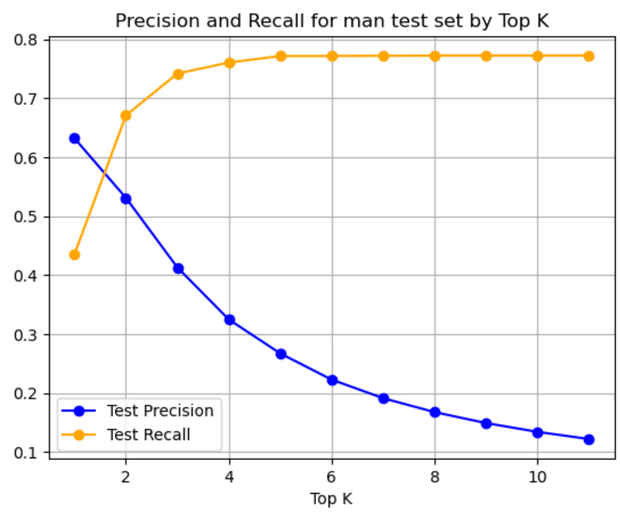

Figure 6: Precision at Top k and recall at Top k of the recommendation system on the man test dataset.

suitable for the corresponding personal characteristics.

$$\text{Precision@k} = \frac{\text{Number of items relevant to the personal characteristic in top K}}{K} \tag{3}$$

$$\text{Recall@k} = \frac{\text{Number of items relevant to the personal characteristic in top K}}{\text{Total number of items relevant to the personal characteristic}} \tag{4}$$

A.3.1 Man

Figure 13 and Figure 14 show the precision at Top k and recall at Top k. Figure 15, Figure 16 and Figure 17 show the precision and recall curves.

A.3.2 Woman

Figure 18 and Figure 19 show the precision at Top k and recall at Top k. Figure 20, Figure 21 and Figure 22 show the precision and recall curves.

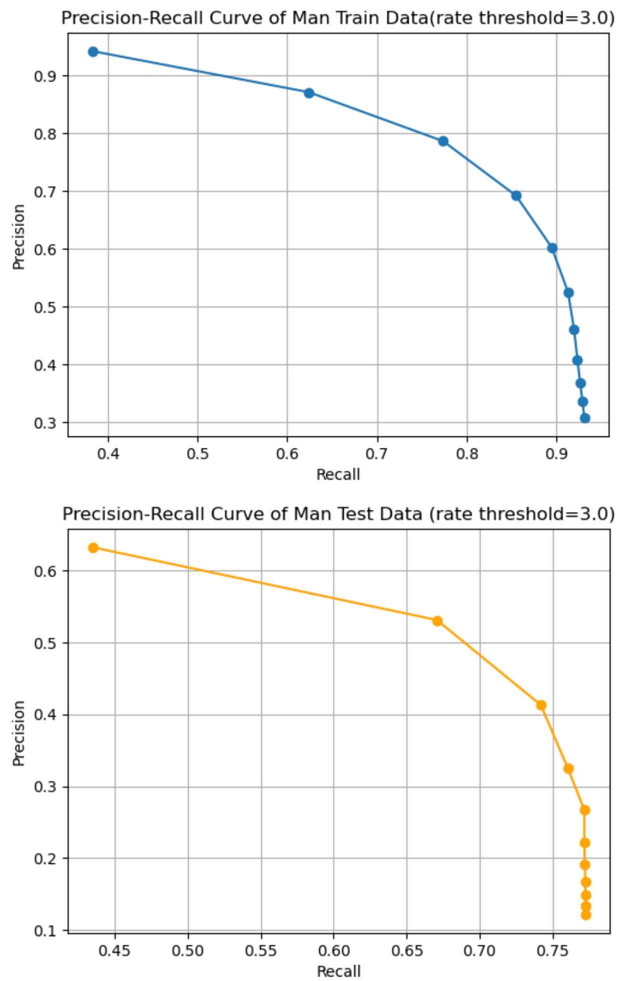

Figure 7: Precision-recall curve of man training set and test set

A.4 Limitation of Measuring Precision with Test Dataset

As shown in Figure 24and 26, in the test dataset, many of the users rated only 1-2 clothes. This makes the precision and recall inaccurate. Therefore, we measure the precision at 10 and recall at 10 computed based on items relevant to the personal characteristic

A.5 Model Comparison

In this section, we compare the recommendation system based on xgboost, lightgbm, gradientboosting, randomforest, and decision tree model.

We created three groups of machine learning models with slightly different parameters:

A.5.1 Group 1

```
models = {
    "XGBoost": xgb.XGBRegressor(
        n_estimators=500, learning_rate=0.05, max_depth=6,
        subsample=0.8, colsample_bytree=0.8, gamma=0,
        reg_alpha=0.1, reg_lambda=1, random_state=0
    ),
    "LightGBM": lgb.LGBMRegressor(
        n_estimators=500, learning_rate=0.05, num_leaves=31,
        max_depth=6, subsample=0.8, colsample_bytree=0.8,
```

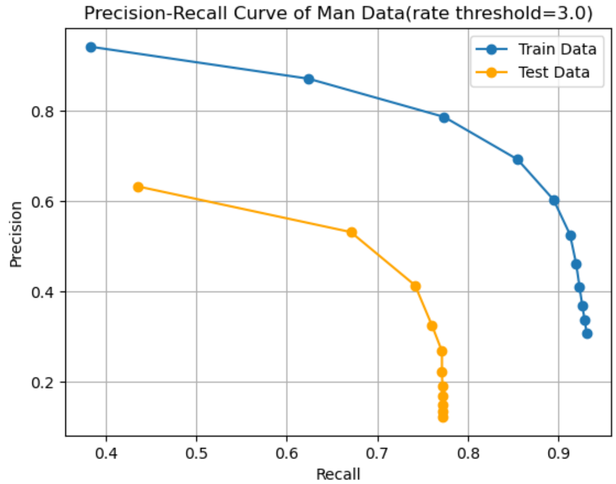

Figure 8: Comparison of precision-recall curve of man training set and test set

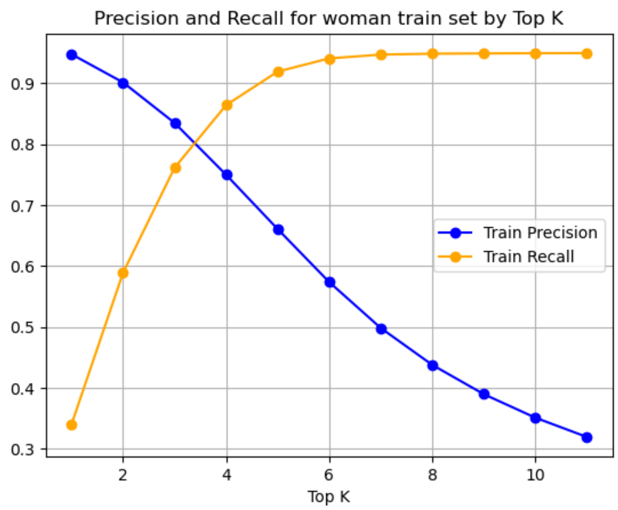

Figure 9: Precision at Top k and recall at Top k of the recommendation system on the woman training dataset.

```
    reg_alpha=0.1, reg_lambda=1, random_state=0
),
"GradientBoosting": GradientBoostingRegressor(
    n_estimators=300, learning_rate=0.05, max_depth=5,
    min_samples_split=5, min_samples_leaf=2, subsample=0.8,
    random_state=0
),
"RandomForest": RandomForestRegressor(
    n_estimators=300, max_depth=10, min_samples_split=5,
    min_samples_leaf=2, random_state=0
),
"CatBoost": CatBoostRegressor(
    iterations=500, learning_rate=0.05, depth=6,
    l2_leaf_reg=1, verbose=0,subsample=0.8,random_state=0
),
"DecisionTree": DecisionTreeRegressor(
    max_depth=10, min_samples_split=5, min_samples_leaf=2,
```

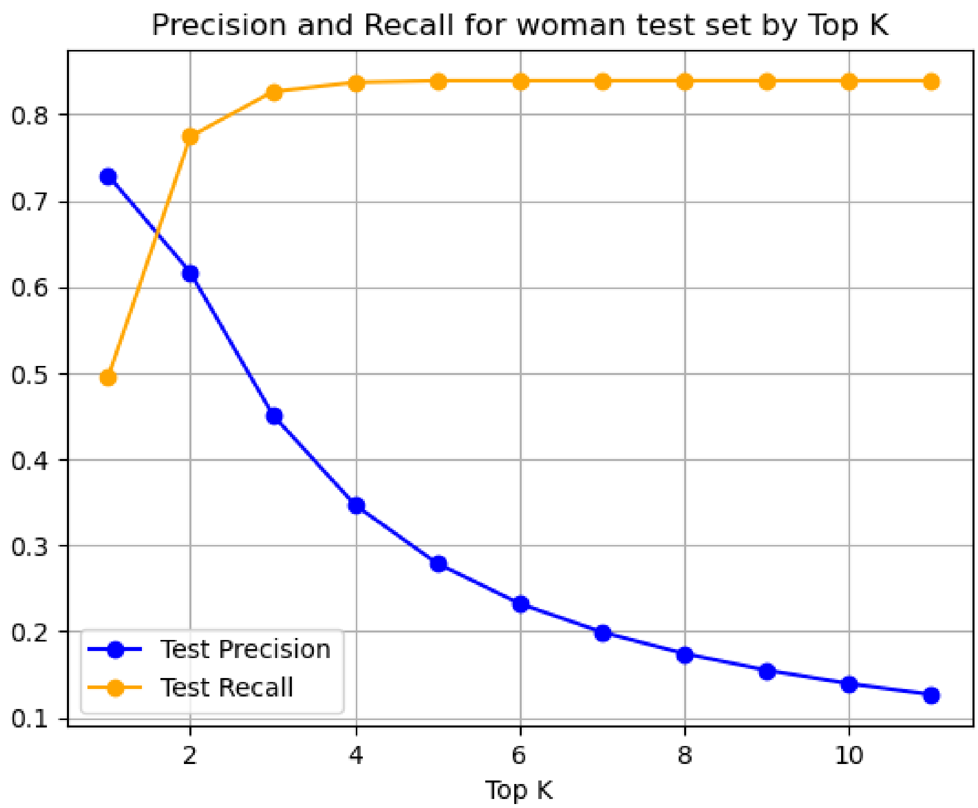

Figure 10: Precision at Top k and recall at Top k of the recommendation system on the woman test dataset.

```
        random_state=0
    )
}
```

A.5.2 Group 2: More Estimators

```
models = {
    "XGBoost": xgb.XGBRegressor(
        n_estimators=1000, learning_rate=0.05, max_depth=6,
        subsample=0.8, colsample_bytree=0.8, gamma=0,
        reg_alpha=0.1, reg_lambda=1, random_state=0
    ),
    "LightGBM": lgb.LGBMRegressor(
        n_estimators=1000, learning_rate=0.05, num_leaves=31,
        max_depth=6, subsample=0.8, colsample_bytree=0.8,
        reg_alpha=0.1, reg_lambda=1, random_state=0
    ),
    "GradientBoosting": GradientBoostingRegressor(
        n_estimators=1000, learning_rate=0.05, max_depth=5,
        min_samples_split=5, min_samples_leaf=2, subsample=0.8,
        random_state=0
    ),
    "RandomForest": RandomForestRegressor(
        n_estimators=1000, max_depth=10, min_samples_split=5,
        min_samples_leaf=2, random_state=0
    ),
    "CatBoost": CatBoostRegressor(
        iterations=1000, learning_rate=0.05, depth=6,
```

```
        l2_leaf_reg=1, subsample=0.8, verbose=0,random_state=0
    ),
    "DecisionTree": DecisionTreeRegressor(
        max_depth=10, min_samples_split=5, min_samples_leaf=2,
        random_state=0
    )
}
```

A.5.3 Group 3: No Regularization

```
models = {
    "XGBoost": xgb.XGBRegressor(
         random_state=0
    ),
    "LightGBM": lgb.LGBMRegressor(
        random_state=0
    ),
    "GradientBoosting": GradientBoostingRegressor(
        random_state=0
    ),
    "RandomForest": RandomForestRegressor(
         random_state=0
    ),
    "CatBoost": CatBoostRegressor(
        random_state=0
    ),
    "DecisionTree": DecisionTreeRegressor(
        random_state=0
    )
}
```

A.5.4 Man Results

Table 19, Table 20, and Table 21 shows the model comparison results of group1, group2 and group3, respectively with man dataset. Results show that Catboost model with group 1 parameter has highest precision at 10 and recall at 10.

Table 19: Comparison of Group 1 model with man dataset. CatBoost show relatively high personal precision and recall at 10

Model	Train RMSE	Test RMSE	Test Acc	Test Precision	Test Recall	Precision@10	Recall@10
XGBoost	**0.5816**	0.7459	0.63	0.85	0.29	0.6424	0.4697
LightGBM	0.6572	0.7423	**0.64**	0.87	0.28	0.6424	0.4694
GradientBoost	0.6748	0.7373	**0.64**	0.87	0.28	0.6485	0.4727
RandomForest	0.6799	0.7481	0.62	0.87	0.24	0.6242	0.4651
CatBoost	0.6727	**0.7314**	0.63	**0.89**	0.27	**0.6576**	**0.4800**
DecisionTree	0.7086	0.8034	0.63	0.81	**0.30**	0.6061	0.4485

Table 20: Comparison of Group 2 model with man dataset. CatBoost shows relatively high personal precision and recall at 10

Model	Train RMSE	Test RMSE	Test Acc	Test Precision	Test Recall	Precision@10	Recall@10
XGBoost	**0.5003**	0.7623	**0.64**	0.83	0.30	0.6455	0.4701
LightGBM	0.6070	0.7510	**0.64**	0.85	0.29	0.6455	0.4700
GradientBoost	0.5869	0.7502	**0.64**	0.83	**0.31**	0.6424	0.4697
RandomForest	0.6796	0.7483	0.62	**0.87**	0.24	0.6242	0.4632
CatBoost	0.6179	**0.7348**	0.63	0.85	0.28	**0.6485**	**0.4734**
DecisionTree	0.7086	0.8034	0.63	0.81	0.30	0.6061	0.4485

Table 21: Comparison of Group 3 model with man dataset. CatBoost show relatively high personal precision and recall at 10, while RandomForest and DecisionTree suffer from overfitting

Model	Train RMSE	Test RMSE	Test Acc	Test Precision	Test Recall	Precision@10	Recall@10
XGBoost	0.5647	0.7671	0.64	0.85	0.31	0.6394	0.4654
LightGBM	0.6930	0.7389	0.64	**0.90**	0.27	0.6424	0.4657
GradientBoost	0.7469	0.7416	0.62	0.89	0.24	0.6424	0.4715
RandomForest	0.3043	0.7593	0.63	0.86	0.28	0.6212	0.4631
CatBoost	0.6095	**0.7348**	0.64	0.87	0.29	**0.6485**	**0.4724**
DecisionTree	**0.0933**	1.0773	**0.68**	0.67	**0.66**	0.5606	0.4356

A.5.5 Woman Results

Table 22, Table 23, and Table 24 shows the model comparison results of group1, group2 and group3, respectively with woman dataset. Results show that Catboost model with group 1 parameter has highest precision at 10, and Catboost model with group 2 parameter has highest recall at 10.

Table 22: Comparison of Group 1 model with woman dataset. CatBoost shows relatively high personal precision and recall at 10

Model	Train RMSE	Test RMSE	Test Accuracy	Test Precision	Test Recall	Precision@10	Recall@10
XGBoost	**0.635**	0.773	**0.56**	0.84	**0.29**	0.607	0.629
LightGBM	0.703	0.771	0.55	0.84	0.27	0.602	0.625
GradientBoost	0.719	0.769	0.55	0.86	0.26	0.605	0.622
RandomForest	0.722	0.773	0.53	**0.87**	0.22	0.609	0.624
CatBoost	0.719	**0.767**	0.55	0.86	0.26	**0.614**	**0.630**

Table 23: Comparison of Group 2 model with woman dataset. Catboost shows relatively high personal precision and recall at 10

Model	Train RMSE	Test RMSE	Test Accuracy	Test Precision	Test Recall	Precision@10	Recall@10
XGBoost	**0.559**	0.785	**0.57**	0.83	**0.31**	**0.614**	0.631
LightGBM	0.659	0.775	0.56	0.84	0.29	0.605	0.625
GradientBoost	0.641	0.779	0.56	0.85	0.30	0.612	0.630
RandomForest	0.721	0.773	0.53	**0.87**	0.22	**0.614**	0.627
CatBoost	0.672	**0.771**	0.56	0.85	0.28	0.612	**0.632**
DecisionTree	0.752	0.823	0.53	0.81	0.24	0.563	0.601

A.6 Lightweight Face Shape Classification

We reduce the face shape classification model size using EfficientNetB2. We change the input image size to 260×260, and add dropout(0.3) for preventing overfitting. We initially trained the model with the EfficientNetB2 backbone frozen for 25 epochs, learning rate 1e-3 (with cosine decay) and AdamW (weight_decay=5e-5), and then fine-tuned the entire network by unfreezing all layers and using a low learning rate (3e-5 with cosine decay) and AdamW (weight_decay=5e-6) for 30 epochs. Earlystopping parameters are same as previous model. As shown in Table 25, We achieve similar accuracy while almost halving the model size.

A.6.1 Increasing Accuracy by Data Augmentation and Label Smoothing

Inspired by [12], we totally renew the face shape classification code (ver4, ver5, ver6) with more data augmentation(horizontal flip, rotation, changing brightness, width shifting, height shifting, zoom, normalization to (-1,1)), label smoothing, learning rate 5e-3 and batch size 32 for 30 epochs. As shown in the Table 26, The results show that data augmentation and label smoothing helps increasing accuracy.

You can view the Ver5 code on the Kaggle(`https://www.kaggle.com/code/minchocoin/tensorflow-efficientnetb2-test-accuracy-83percent`). Due to the random state, results on Colab(Table 26) and Kaggle are slightly different (test accuracy of Kaggle are 83.30%).

Table 24: Comparison of Group 3 model with woman dataset. CatBoost shows relatively high personal precision and recall at 10, while RandomForest and DecisionTree suffer from overfitting

Model	Train RMSE	Test RMSE	Test Accuracy	Test Precision	Test Recall	Precision@10	Recall@10
XGBoost	0.621	0.796	0.57	0.84	0.32	0.600	0.623
LightGBM	0.734	**0.769**	0.55	0.87	0.26	0.607	0.624
GradientBoost	0.778	0.770	0.53	**0.88**	0.21	0.600	0.621
RandomForest	0.314	0.773	0.57	0.86	0.30	0.607	0.622
CatBoost	0.659	0.774	0.56	0.84	0.29	**0.612**	**0.629**
DecisionTree	**0.087**	1.082	**0.64**	0.70	**0.66**	0.544	0.596

Table 25: Comparison of original and lightweight model. Results show that EfficientNetB2 model achieve similar accuracy while has small size

Model version	Test loss	Test accuracy	Model size
Ver2(EfficientNetB4 backbone)	1.2050	75.83%	17.68M
Ver3(EfficientNetB2 backbone)	1.5023	75.13%	7.78M

A.6.2 Quantization

To run the face shape classification model on the CPU-only AWS EC2, we quantize the ver5 model to int8 precision using tensorflow lite. you can view the quantization code on `https://github.com/minchoCoin/capstone-team-24-page/blob/main/appendix/faceshape_model_quantization.ipynb`, and Table 27 shows the quantization results. results show that model size was reduced by one-tenth, inference time at Colab CPU was reduced by one-half, and accuracy was reduced by about 10% point

A.7 Conclusion

We calculate precision, recall, precision at k, recall at k. Also, we draw the precision-recall curve. Above results show that the recommendation system can recommend the clothes based on the personal characteristic (personal color, faceshape, bodyshape). Additionally, we compare the recommendation system based on various machine learning model. Results show that XGBoost, GradientBoosting, and Catboost shows relatively high precision at 10 and recall at 10 on personal characteristics.

Also, we lightweight the face shape classification model using EfficientNetB2. Results show that lightweight model achieve similar accuracy while reducing the model size. Additionally, we renew the face shape classification model (ver4, ver5, ver6) model. Results show that data augmentation helps increasing accuracy.

Table 26: Comparison of models. The results show that data augmentation and label smoothing helps increasing accuracy

Model version	Test loss	Test accuracy	Model size
Ver2(EfficientNetB4 backbone)	1.2050	75.83%	17.68M
Ver3(EfficientNetB2 backbone)	1.5023	75.13%	7.78M
Ver4(EfficientNetB4 backbone with [12])	0.7366	83.40%	17.68M
Ver5(EfficientNetB2 backbone with [12])	0.7444	83.60%	7.78M
Ver6(EfficientNetB0 backbone with [12])	0.8224	81.10%	4.06M

Table 27: Comparison between base model(ver5) and integer quantization model. results show that model size was reduced by one-tenth, inference time at Colab CPU was reduced by one-half, and accuracy was reduced by about 10% point

	Model file size	Inference time per image	Accuracy
Base model	89.39MB	160.31ms	83.3%
Int8 model	8.71MB	73.78ms	75.0%

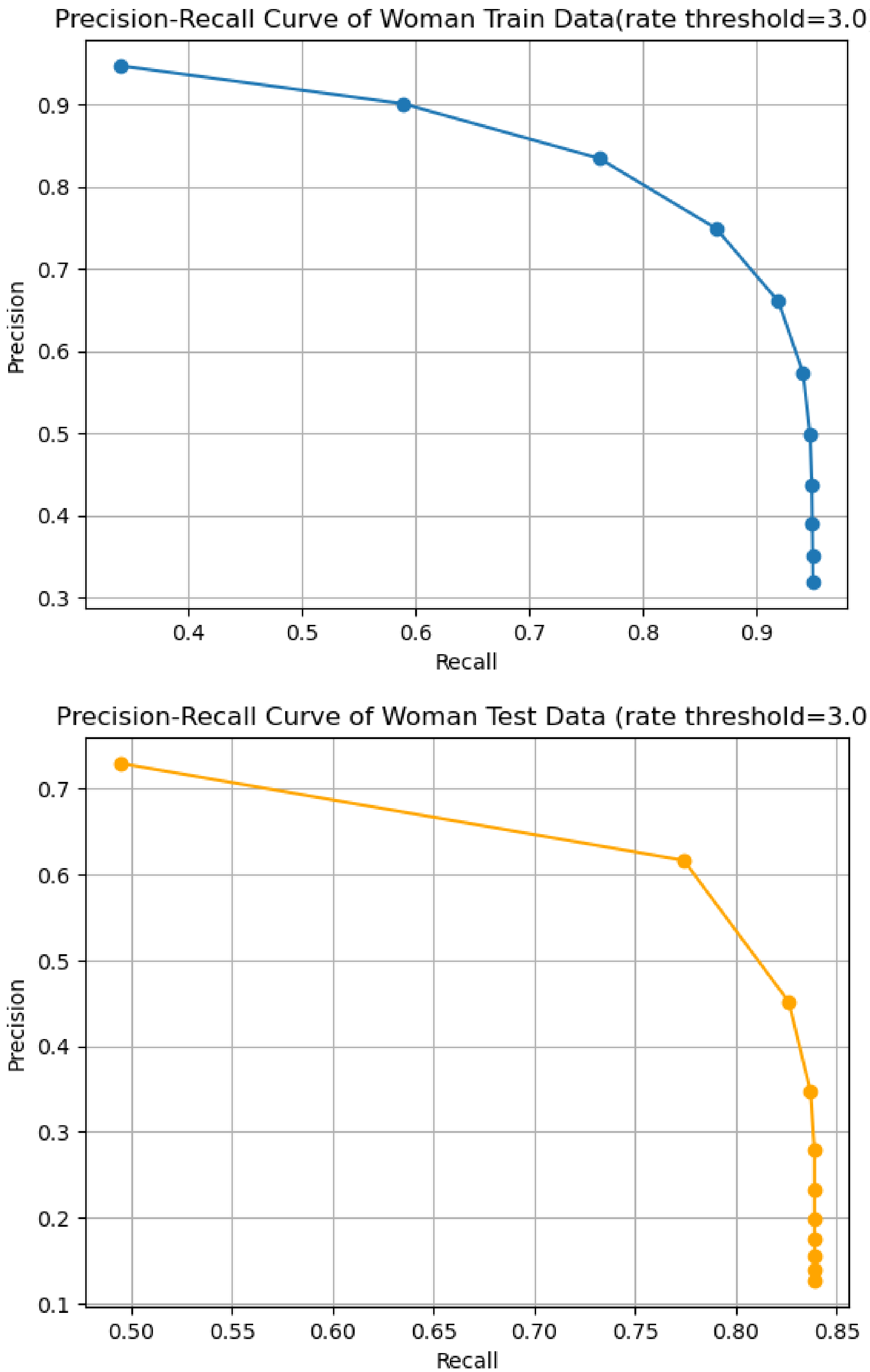

Figure 11: Precision-recall curve of woman training set and test set

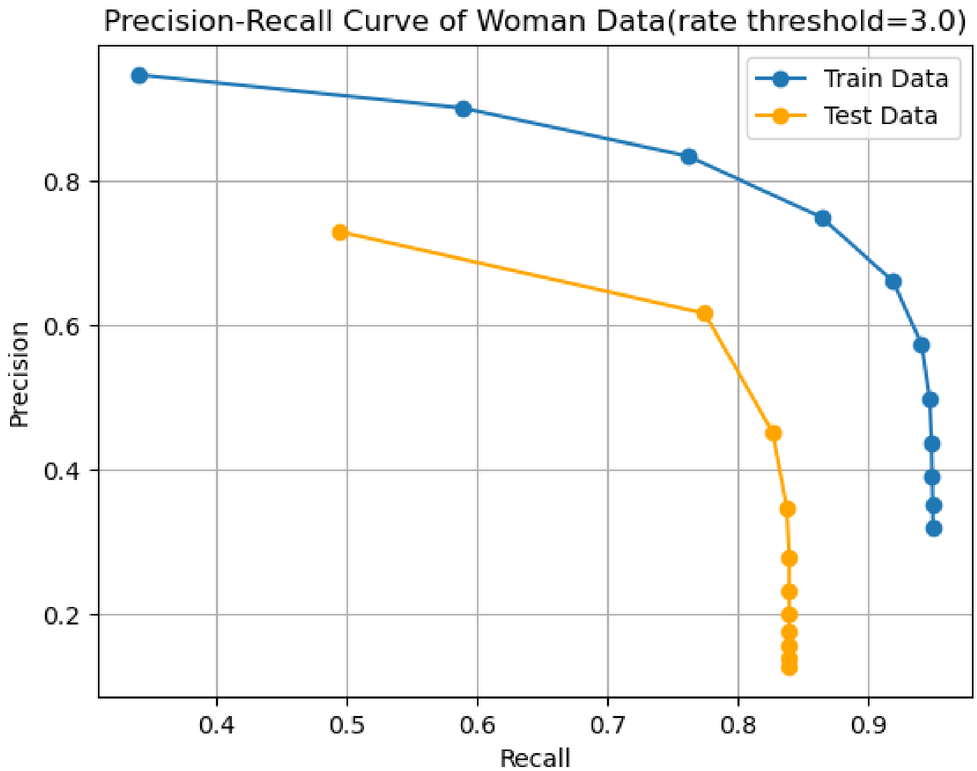

Figure 12: Comparison of precision-recall curve of woman training set and test set

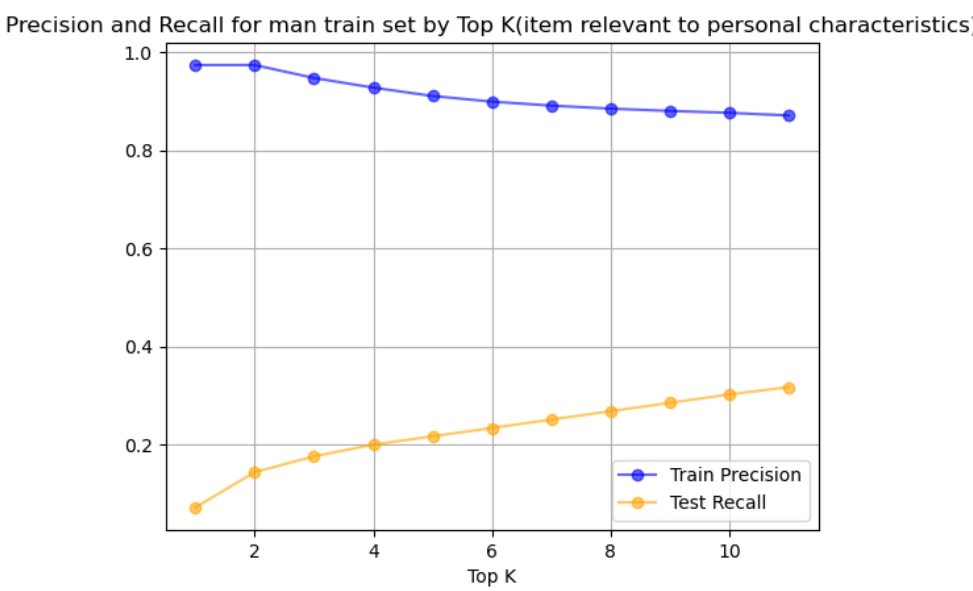

Figure 13: Precision at Top k and recall at Top k computed based on items relevant to personal characteristics for the men's training dataset.

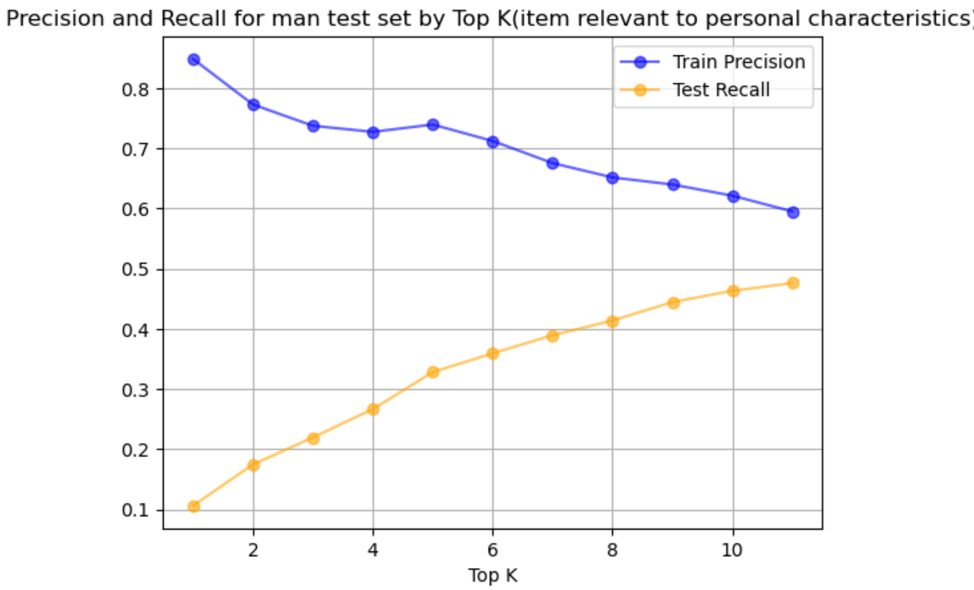

Figure 14: Precision at Top k and recall at Top k computed based on items relevant to personal characteristics for the men's test dataset.

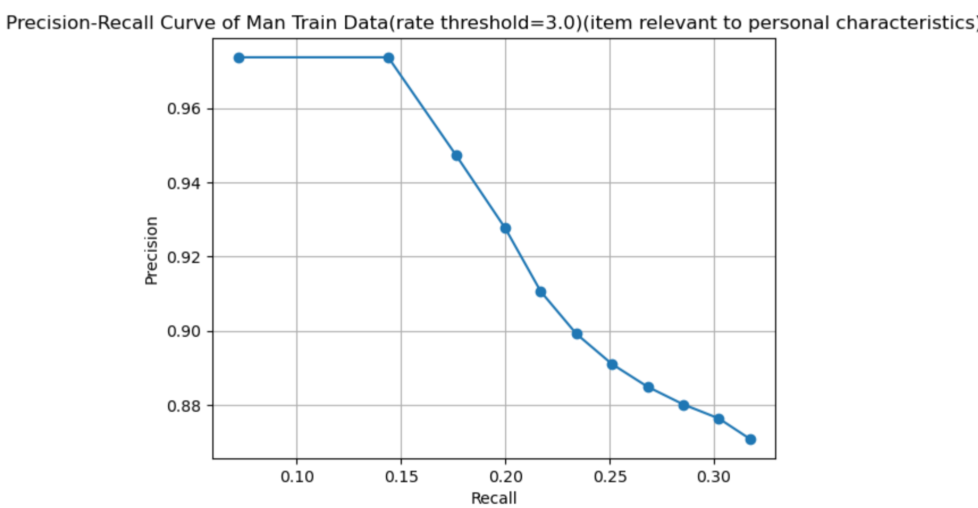

Figure 15: Precision-recall curve computed based on items relevant to personal characteristics for the men's training dataset.

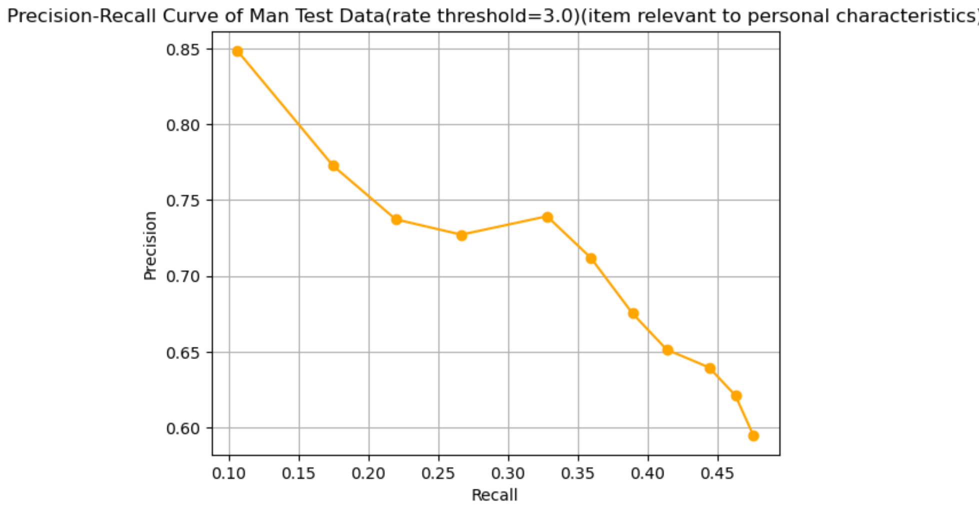

Figure 16: Precision-recall curve computed based on items relevant to personal characteristics for the men's test dataset.

Figure 17: Comparison of precision-recall curves computed based on items relevant to personal characteristics for the men's test dataset.

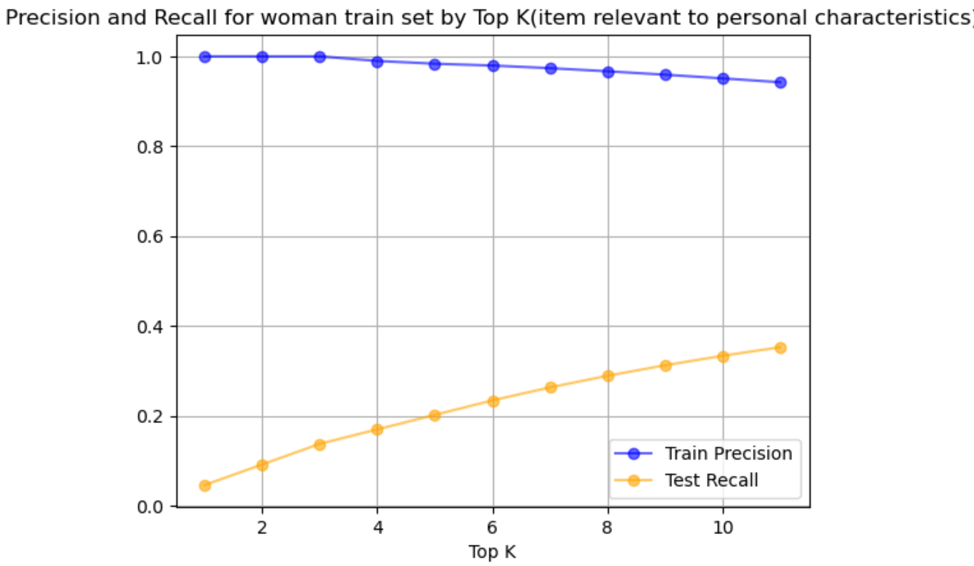

Figure 18: Precision at Top k and recall at Top k computed based on items relevant to personal characteristics for the women's training dataset.

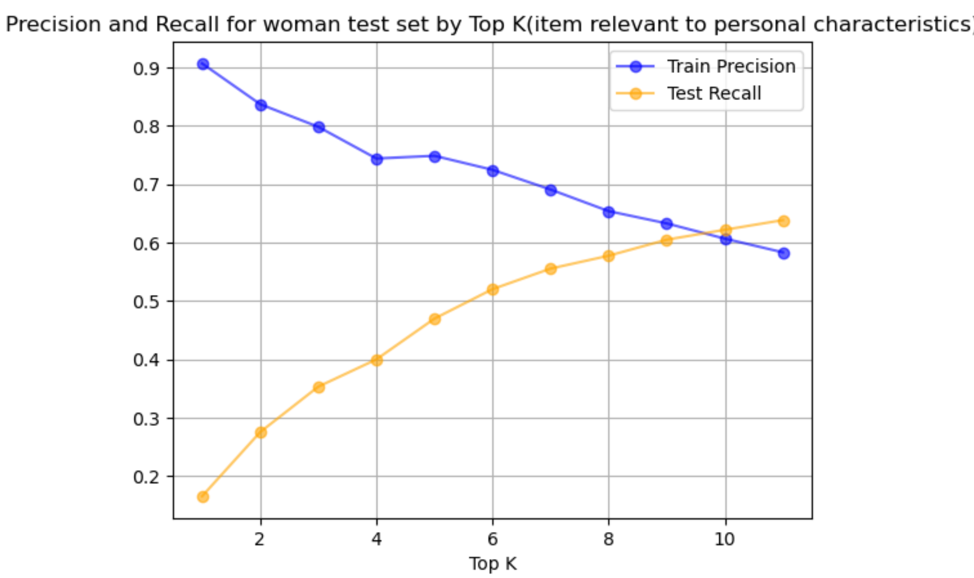

Figure 19: Precision at Top k and recall at Top k computed based on items relevant to personal characteristics for the women's test dataset.

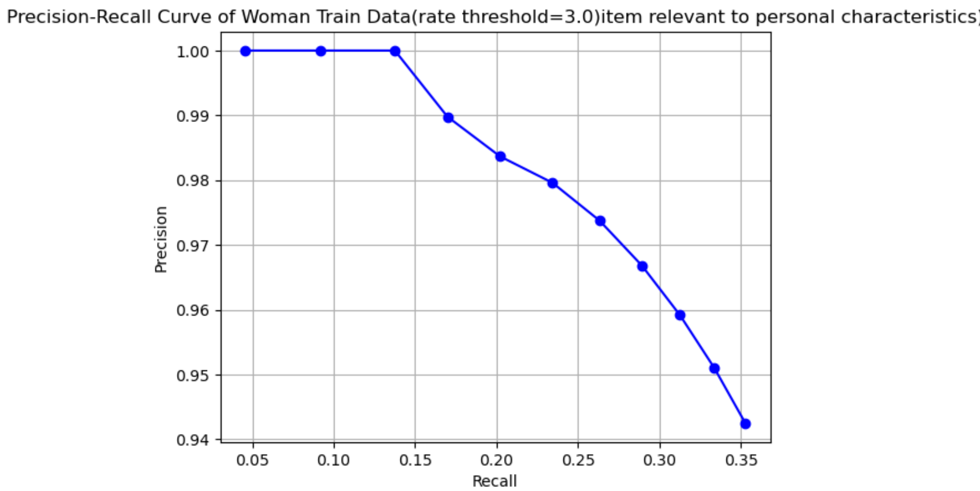

Figure 20: Precision-recall curve computed based on items relevant to personal characteristics for the women's training dataset.

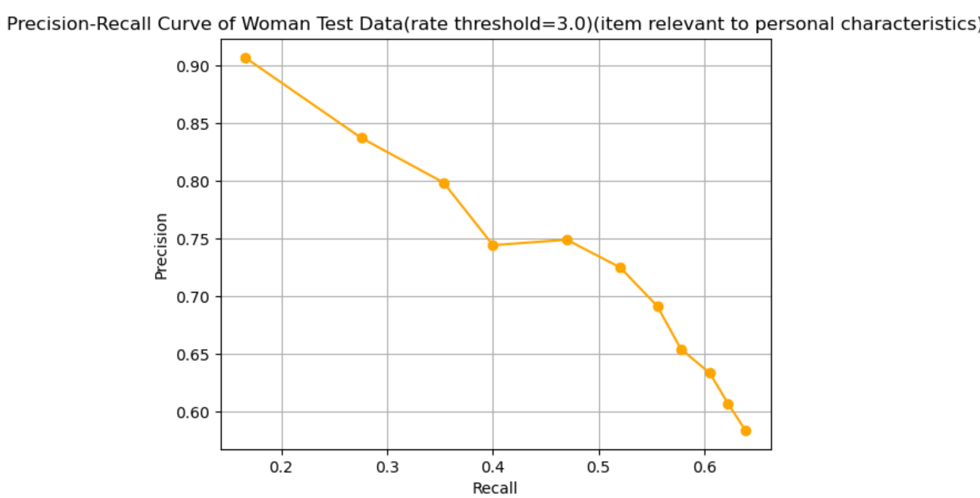

Figure 21: Precision-recall curve computed based on items relevant to personal characteristics for the women's test dataset.

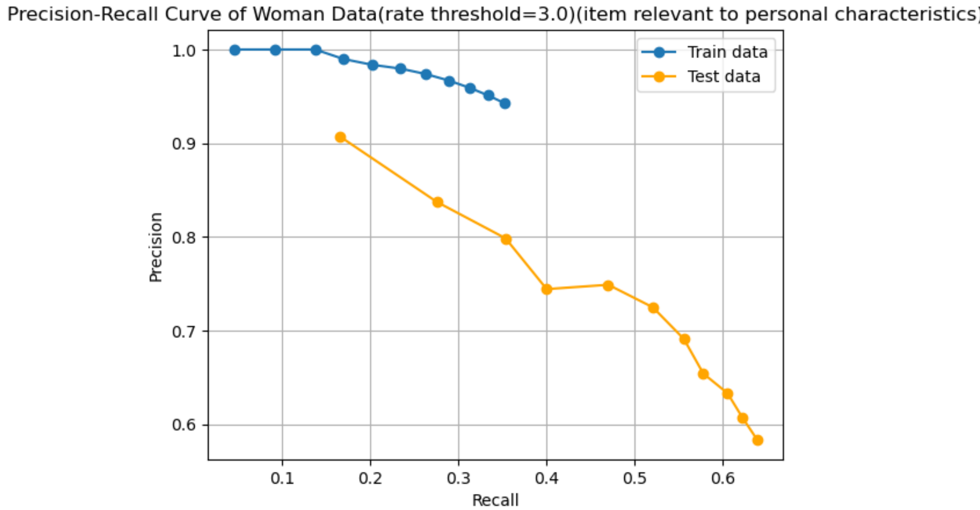

Figure 22: Comparison of precision-recall curves computed based on items relevant to personal characteristics for the women's test dataset.

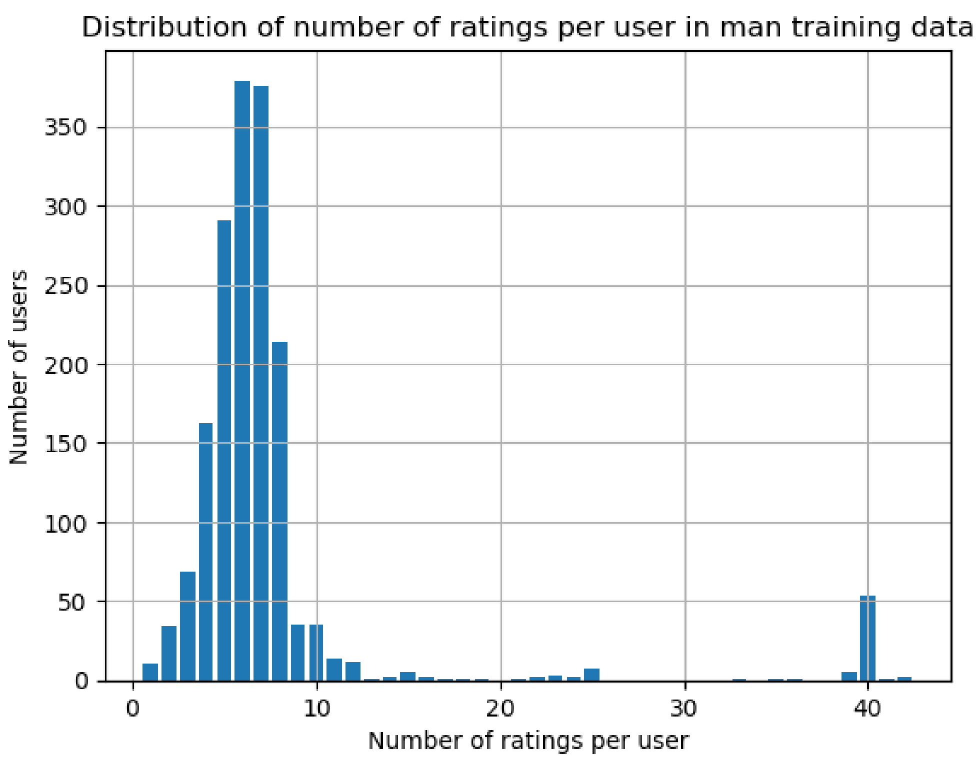

Figure 23: Distribution of number of ratings per user in man training dataset

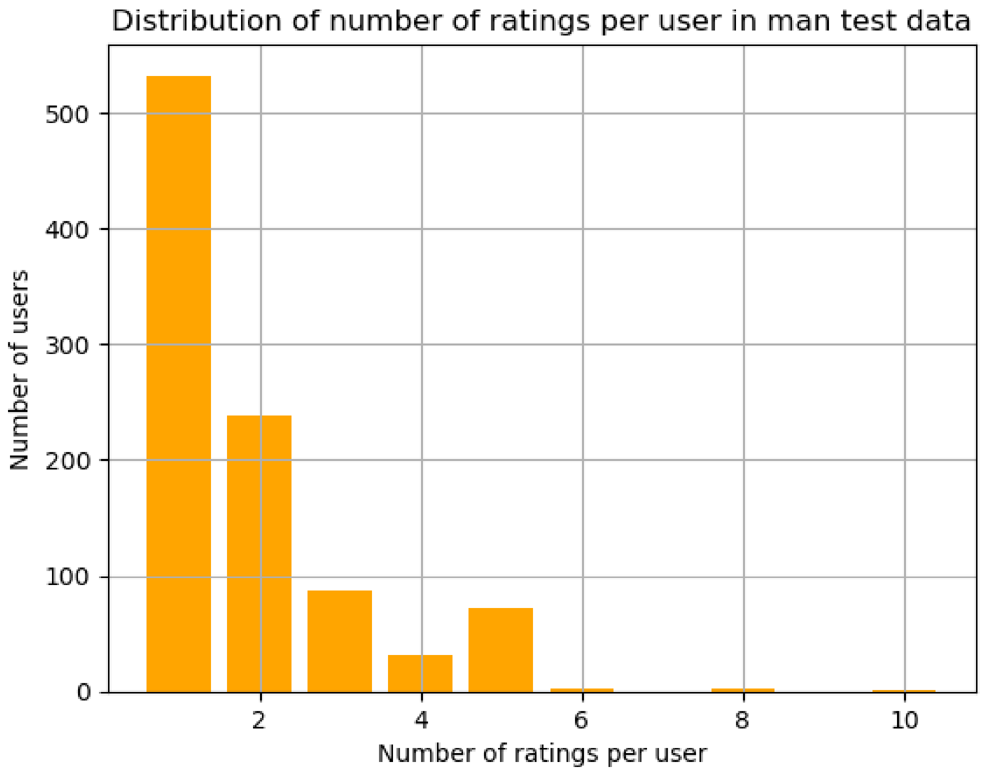

Figure 24: Distribution of number of ratings per user in man test dataset

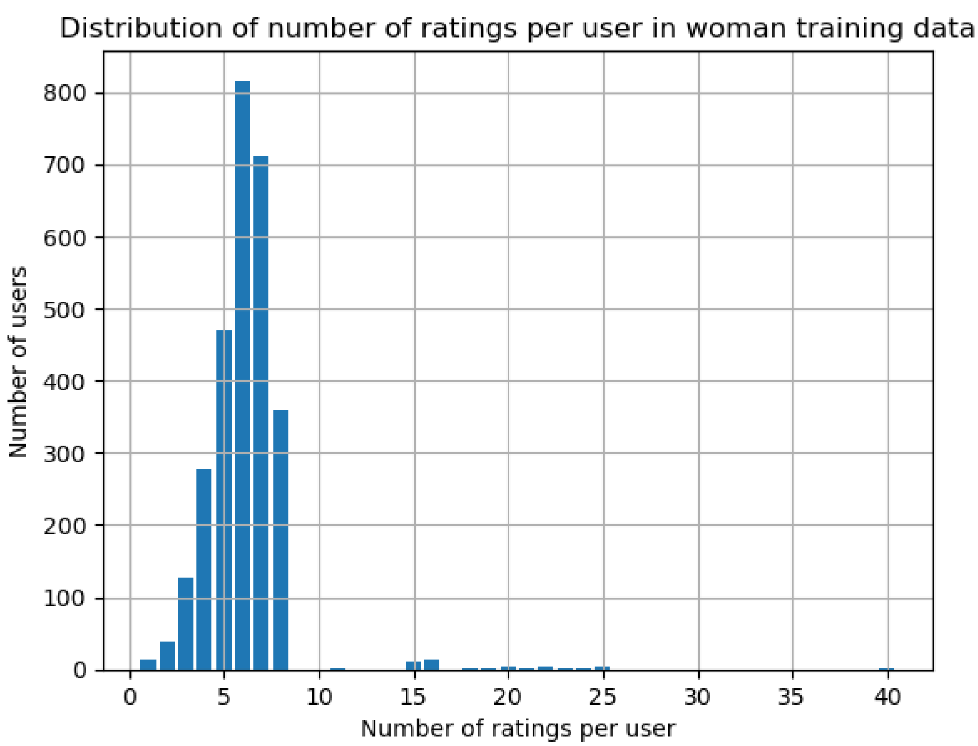

Figure 25: Distribution of number of ratings per user in woman training dataset

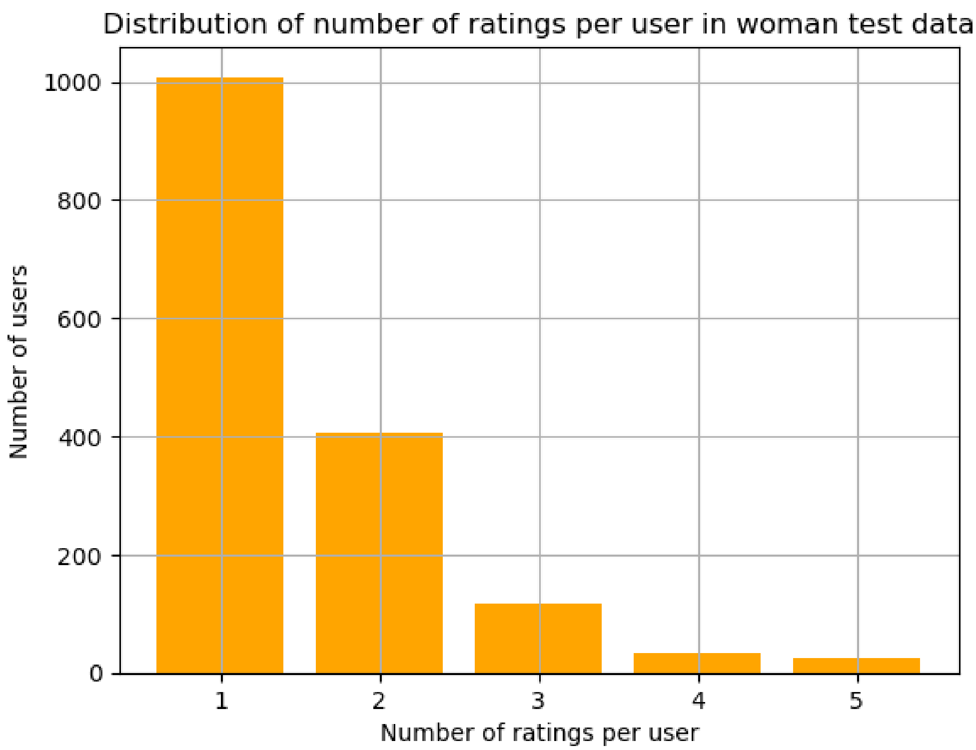

Figure 26: Distribution of number of ratings per user in woman test dataset