FLEX: AI-Powered Diet and Workout Plan Application

Jihoon Lee, Jaewook Shin, Mano Hong, Manu Gomez

Sungkyunkwan University

October 3, 2024

Abstract

This document introduces the "FLEX" application, which stands for Food, Lifestyle, Exercise, and eXperience. The goal of this project is to provide a diet and workout plan for people who want to eat what they like and also want to become healthier. Solutions will be derived using NLP (Natural Language Processing) and MLP (Multi-Layer Perceptron) models. These models will help create personalized diet and workout plans based on each person's food preferences, workout preferences, and body metrics, such as weight, height, skeletal muscle mass, and obesity level.

Keywords: NLP, Reinforcement Learning, Feedback, Personalized

1 Introduction

Today, many people try to maintain a healthy lifestyle while enjoying the foods they love. The problem is that it is challenging to balance personal preferences with nutritional and fitness goals. This project introduces an application designed to solve this problem by providing personalized diet and workout plans based on each person's preferences and body metrics.

This application considers a person's food and workout preferences with body metrics, such as height, weight, skeletal muscle mass, and obesity level. AI models analyze these factors to create personalized solutions for health and wellness.

With a focus on personalization, users can enjoy their preferred foods without compromising their preferred activities. This will also make the journey enjoyable and sustainable. By providing individuals with personalized plans, we ultimately aim to create an environment that values well-being alongside diverse tastes and lifestyle choices.

2 Motivation

2.1 Motivation

As time goes by, the obesity rate is only increasing and the size of the health-related market continues to grow accordingly.

The figure below shows that the obesity rate is gradually increasing around the world.

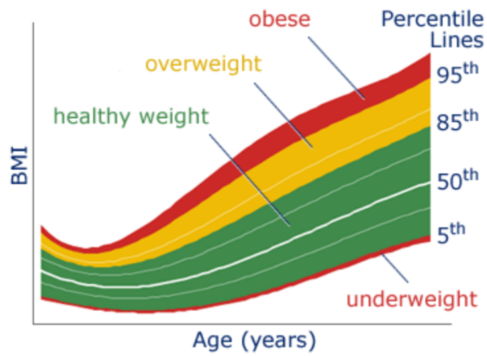

Figure 1: Obesity rate.

The figure below shows that the health and wellness market is gradually growing.

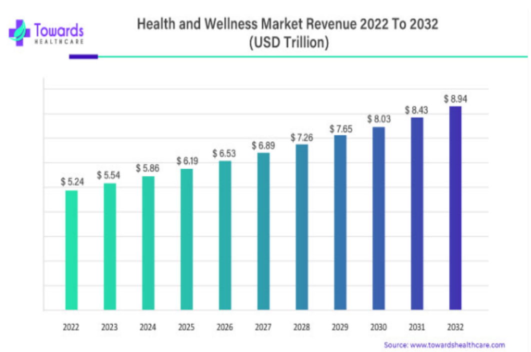

Figure 2: Health and Wellness Market Revenue.

But current health management programs often fail to take into account individual differences in fitness levels and dietary preferences, resulting in reduced motivation and less effective outcomes. The rigidity of conventional diets, such as meal plans centered around foods like chicken breast, contributes to a lack of enjoyment in maintaining a healthy lifestyle.

3 Related Work and Objective

3.1 Related Work

3.1.1 Delighting Palates with AI

The cited paper presents a method combining reinforcement learning and collaborative filtering to create personalized meal plans that meet users' nutritional needs and preferences, aiming to enhance long-term adherence. However, it does not address exercise planning, which limits its effectiveness for users seeking both dietary and physical activity recommendations to meet health goals.

3.1.2 AI Meal Planner App

This work generates meal plans suggestion based on a user's age,height, gender. But it does not consider about exercise and users taste.

3.1.3 Fitness Trackers with Diet Suggestions

Several fitness applications, such as Fitbit and MyFitnessPal, integrate exercise tracking and basic diet suggestions. However, they typically rely on pre-defined databases of meals and workouts, lacking the level of personalization needed to truly cater to individual user preferences. While effective at helping users track activities and caloric intake, they do not generate personalized recommendations based on taste, dietary restrictions, or the user's specific fitness goals. The focus on generic data limits the flexibility needed for sustained user engagement.

3.1.4 Hybrid Approach to Fitness and Diet

A notable example of the hybrid approach can be found in applications such as Noom and Freeletics, which attempt to combine personalized fitness routines with meal planning. Noom, for instance, employs a combination of cognitive behavioral therapy (CBT) principles and hybrid recommendation techniques, using both collaborative filtering and content-based filtering to tailor meal and workout suggestions to the user's goals. Freeletics, on the other hand, offers custom workout plans alongside basic nutrition guidance, attempting to synchronize physical activity with dietary habits.

However, these systems often rely on initial user inputs (e.g., age, weight, and fitness level) without dynamically adjusting based on real-time feedback or evolving preferences. Although they represent a step forward in merging diet and exercise management, their lack of continuous learning mechanisms limits the depth of personalization and adaptability over time. and also they do not deal with various Korean foods.

3.2 Objective

The objective of FLEX is to offer a personalized approach to exercise and diet management. By tailoring both meal plans and workouts to each individual's unique preferences and needs, our application seeks to empower users to maintain a sustainable and enjoyable health management routine. FLEX aims to break the stigma of restrictive dieting and foster a more holistic and flexible approach to wellness.

4 Problem Statement and Proposed Solution

4.1 Problem Statement

Existing health management programs fail to account for individuals' fitness levels and dietary preferences. To solve this problem, we focus on individual characteristics.

4.2 Proposed Solution

4.2.1 Feature Selection

Out of thousands of available foods, the project will select specific items to be utilized. Since our target audience is Korean, we will focus on choosing foods that are commonly consumed by Koreans.

Furthermore, we will set Sungkyunkwan University students as the initial customer base and prioritize selecting foods that are easily accessible around the campus area.

4.2.2 Database development

The database will be constructed by evaluating preference scores for the selected features from all participants

This project is likely to face the cold start problem. To mitigate this, we will leverage the maximum available human and financial resources to build the initial database as large as possible. Additionally, the feedback system mentioned in section 4.2.10 will be utilized to gradually expand the database over time.

Participant	Tteokbokki	Pizza	Spaghetti	Sushi	Bibimbap	Fried Chicken	Kimchi Stew
Mano	10	6	6	7	2	3	8
JaeWook	4	10	10	9	4	7	5
JiHoon	2	10	10	1	3	2	8
Manu	7	6	4	5	3	3	8

Table 1: Preference Scores for Different Foods

4.2.3 New user

It is not feasible to ask a new user to rate their preferences for all available food items. Therefore, we aim to apply a fundamental recommendation algorithm approach. Upon joining, new users will be asked to rate their preferences for a carefully curated set of 10 food items. These items, selected by researchers for their diverse characteristics, can be rated on a scale from 1 to 10. This process helps capture the user's general taste profile, allowing the system to generate more accurate recommendations.

Participant	Tteokbokki	Pizza	Spaghetti	Sushi	Bibimbap	Fried Chicken	Kimchi Stew
NewUser	9	5	6				

Table 2: Result of new user survey

4.2.4 Find Similar user

Based on the user's selected preferences for the 10 features, the system will identify the most similar existing user from the database. To determine this, we utilize Euclidean distance as the similarity measure.

$$d(p,q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{1}$$

By comparing the user's preferences with those of others in the database, we can effectively match the new user with similar users, allowing for more personalized recommendations without requiring exhaustive input.

Participant	Tteokbokki	Pizza	Spaghetti	Sushi	Bibimbap	Fried Chicken	Kimchi Stew
Mano	10	6	6	7	2	3	8
NewUser	9	5	6	7	2	3	8

Table 3: The new user's preference score, generated by replicating the data of the most similar existing user.

4.2.5 NLP Model 1

Based on the new user's written responses, the system will analyze their preferred or disliked foods and tastes. This analysis will be used to assign bonus points to the data of the similar user extracted in 4.2.4, further refining the matching process and improving the accuracy of the recommendations by incorporating more nuanced user preferences.

For this task, we will utilize a BERT-based NLP model. The BERT model will be used to assess the sentiment of the user's comments about specific food items, determining whether the user expresses a positive or negative sentiment towards each item. By leveraging BERT's advanced language understanding capabilities, we aim to generate more accurate evaluations of user preferences based on their written feedback, which will be factored into the recommendation process.

4.2.6 Call Food API

The system will call a food API to retrieve nutritional information such as calories, carbohydrates, fats, and other relevant details for each food item. This data will then be stored in the database for each feature (food item) to enhance the accuracy and relevance of the recommendations. The stored nutritional information will be used in conjunction with user preferences and exercise data to generate personalized meal suggestions.

	Tteokbokki	Pizza	Spaghetti	Sushi	Bibimbap	Fried Chicken	Kimchi Stew
preference score	9	5	6	7	2	3	8
calories (kcal)	500	650	400	300	450	700	250
carbohydrates (g)	80	75	60	50	55	45	20
fats (g)	10	20	12	8	7	35	5
protein (g)	7	15	10	12	8	40	6

Table 4: The new user's preference score, generated by replicating the data of the most similar existing user.

4.2.7 MLP Model

Using data about the user's body metrics, an MLP model will calculate the user's daily recommended intake based on their physical needs. This intake is then divided across breakfast, lunch, and dinner. Since there are variations in the portion sizes consumed during each meal, the division will be done after reviewing relevant studies on meal distribution patterns. The corresponding amounts are subtracted from the sequences generated in 4.2.6 to match the user's nutritional requirements.

By subtracting these amounts, we will also have data on how much the user's consumption exceeds or falls short of the recommended intake, allowing us to capture this critical information. This makes the resulting data highly influential for further adjustments and improvements in the recommendation system, as it helps refine the understanding of the user's nutritional habits.

4.2.8 NLP Model 2

The system will analyze the user's workout routine and call an exercise API to calculate the calories burned for each activity. This calorie data will be subtracted once again from the sequences generated in 4.2.7.

In addition, users will need to specify whether they plan to exercise today, and if so, what type of exercise they will perform and for how long. The system will employ an NLP model to extract keywords related to the exercise type and duration. Afterward, the exercise API will be used to calculate the calories burned for the specified activities. These calorie values will then be subtracted from the sequences, further refining the data to better align with the user's actual caloric expenditure.

4.2.9 Output Extraction

This process involves using preprocessed data to recommend the optimal food choices. Two different methods have been developed for this purpose, and their performances will be compared to identify which approach yields better results.

First method Based on the nutritional information extracted in 4.2.8, the system will perform a comparison against predefined outlier thresholds corresponding to the user's goal (e.g., weight maintenance, weight gain, or weight loss). The system will then identify the five most similar food candidates. From these candidates, the food item with the highest preference score will be selected for recommendation.

- For weight maintenance, the outlier thresholds for all nutritional factors (calories, carbohydrates, protein, and fat) are set to zero.
- For weight gain, the outliers are positive values, represented as $+\alpha$.
- For weight loss, the outliers are negative values, represented as $-\alpha$.

Second method Through reinforcement learning, the user continuously evaluates the system's performance, allowing the model to identify the best-performing outcomes. User feedback is incorporated into the learning process, gradually improving the system's performance and ultimately leading to better recommendation results.

4.2.10 FeedBack

By utilizing this feedback mechanism, the system's performance will improve over time as it continuously learns from user interactions. As more data is gathered and updated, the system will refine its ability to provide accurate recommendations, while the database becomes increasingly reliable. This ongoing process of adjusting and enhancing user profiles ensures that the system grows more effective and personalized with each use.

The user provides a new preference score for the selected food item, and this updated score is immediately recorded in the system's database.

In addition to this, the system adjusts the weights of all food items based on the user's consumption behavior. For food items that the user did not consume, a small positive weight is added, indicating a slight increase in preference uncertainty for those items. Conversely, for foods that the user did consume, a small negative weight is subtracted, reflecting that the user has actively chosen and rated these foods.

When a new user begins using the system, their data is initially replaced with the most similar user's information, which is determined through a survey. This new user data is then added to the database as a fresh entry, allowing the system to start making personalized recommendations.

For existing users, when they interact with the system, only the preference score for the specific food items they select is updated in the database. This ensures that their preferences are continuously refined without altering the overall structure of their previously stored data. This approach helps the system maintain a personalized profile for each user, adjusting only when new preferences are expressed or updated.

5 Technical Background

5.1 BERT (Bidirectional Encoder Representations from Transformers)

BERT is a transformer-based model developed by Google that has revolutionized natural language processing (NLP). Unlike traditional models, BERT uses a bidirectional approach, meaning it reads text both from left to right and right to left, allowing it to better understand the context of a word within a sentence. This deep contextual understanding enables BERT to excel in a wide range of NLP tasks such as sentiment analysis, question answering, and keyword extraction. For keyword extraction, BERT can be fine-tuned to detect important phrases by analyzing the relationship between tokens and their contextual importance.

5.2 MLP (Multi-Layer Perceptron)

MLP is a class of feedforward artificial neural networks (ANN) that consist of at least three layers: an input layer, one or more hidden layers, and an output layer. It is fully connected, meaning each node in one layer is connected to every node in the next layer. MLPs are capable of learning complex patterns in data through backpropagation and are widely used for classification and regression tasks. In this project, the MLP model is employed to predict the user's daily nutritional needs based on their physical metrics.

5.3 Reinforcement Learning

Reinforcement Learning (RL) is an area of machine learning where an agent learns to make decisions by interacting with an environment. The agent takes actions and receives feedback in the form of rewards or penalties. Over time, the agent optimizes its actions to maximize cumulative rewards. In this project, RL is used to continuously improve the recommendation system by incorporating user feedback. As users interact with the system and provide feedback on the recommendations, the RL algorithm adjusts its policies to enhance the accuracy and personalization of future suggestions.

5.4 Recommendation Algorithm

A recommendation algorithm is designed to predict and suggest items that a user may be interested in, based on various data points such as their past behavior, preferences, or similarity to other users. There are several approaches to building recommendation systems, including collaborative filtering, content-based filtering, and hybrid methods. In this project, we employ a hybrid recommendation algorithm that combines user preference data with nutritional information. The system compares the new user's preferences to those of existing users using techniques such as Euclidean distance, and recommends the most suitable food items. This algorithm is continuously refined as more data is collected, ensuring the system adapts to the user's evolving preferences.

5.5 Cross-Platform Development with React Native

React Native is an open-source framework developed by Facebook, designed for the creation of cross-platform mobile applications utilizing a single codebase. By leveraging React Native, the FLEX application is deployed on both Android and iOS devices, ensuring broad accessibility for users. This framework offers a component-based architecture, enabling efficient development while maintaining high performance and responsiveness. React Native's ability to provide near-native experiences across platforms significantly reduces development overhead, ensuring consistent functionality and user experience across various mobile operating systems.

5.6 Backend Architecture with Django and PostgreSQL

The backend infrastructure of the FLEX application is implemented using Django, a high-level Python web framework renowned for its scalability, security, and built-in features that facilitate rapid development. Django's Object-Relational Mapping (ORM) is utilized to interface with a PostgreSQL database, allowing for efficient management of structured data such as user profiles, personalized meal plans, and workout routines. PostgreSQL, an advanced open-source relational database system, offers robust performance and data integrity, making it an ideal choice for handling large volumes of user-generated data. This backend architecture ensures that user data is securely managed while allowing for seamless scaling of the application as the user base grows. Furthermore, Django's modular design, coupled with PostgreSQL's powerful querying capabilities, enables flexible data retrieval and manipulation, ensuring that personalized recommendations are delivered in real-time.

5.7 Cloud Deployment using AWS EC2

To ensure reliability and scalability, the backend is hosted on Amazon Web Services (AWS) Elastic Compute Cloud (EC2) instances. EC2 offers scalable compute capacity in the cloud, enabling FLEX to dynamically adjust resources based on traffic and user demand. By leveraging EC2's auto-scaling features, the application can maintain high availability and performance, even under peak loads. Additionally, AWS provides a secure and compliant environment, safeguarding user data while enabling the global deployment of the application. and storage needs.

6 Planning in Detail

6.1 Role Distribution

The project is largely divided into three parts: Front-end, Back-end, and AI. AI includes data preprocessing and modeling. The roles are divided based on each member's interests and capabilities. Additionally, data collection will be done by all members.

Name	Role
Jihoon Lee	Front-end, UI/UX Design, Back-end
Jaewook Shin	Back-end
Mano Hong	Data Science, AI, Front-end
Manu Gomez	Data Science, AI

Table 5: Role Distribution Table.

6.2 Development Schedule

The table below is a rough schedule of our project. This can be changed based on circumstances.

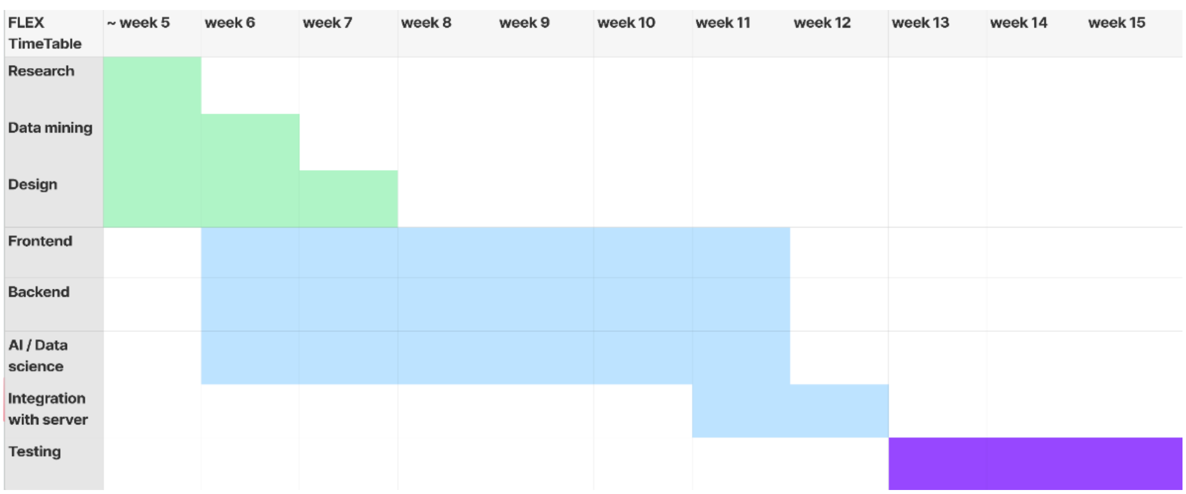

Figure 3: Project Schedule.

References

1. Delighting Palates with AI : https://www.mdpi.com/2072-6643/16/3/346

2. AI Meal Planner App : https://github.com/myselfshravan/AI-Meal-Planner

3. Myfitnesspal : https://www.myfitnesspal.com/ko

4. fitbit : https://support.google.com/product-documentation/answer/13823068?hl=kosjid=8476029724393428024-AP

5. noom : https://www.ko-noom.com/

6. freeletics : https://www.freeletics.com/en/