

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

Кваліфікаційна наукова
праця на правах рукопису

ЛАДИЖЕЦЬ ВІКТОР ІВАНОВИЧ

УДК 004.4; 004.6; 004.82

ДИСЕРТАЦІЯ
ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПЛАНУВАННЯ ЗБАЛАНСОВАНОГО
ХАРЧУВАННЯ ЛЮДИНИ

126 – Інформаційні системи та технології

12 – Інформаційні технології

Подається на здобуття наукового ступеня **доктора філософії**

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

В.І.Ладижець

Науковий керівник **Теренчук Світлана Анатоліївна**, кандидат фізико-математичних наук, професорка кафедри інформаційних технологій проектування та прикладної математики Київського національного університету будівництва і архітектури

Київ – 2025

АНОТАЦІЯ

Ладигець В.І. Інтелектуальна система планування збалансованого харчування людини. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 126 «Інформаційні системи та технології». – Київський національний університет будівництва і архітектури. – Київ, 2025.

Дисертаційну роботу присвячено вирішенню завдання надання людям надійної, науково обґрунтованої, ефективної і загальнодоступної системи персоналізованого планування збалансованого харчування.

Наукова новизна одержаних результатів. Наукова новизна дисертаційної роботи полягає у тому, що:

вперше розроблено:

- метод рекомендації рецептів, що дозволяє враховувати персональні вподобання користувача та його обмеження щодо інгредієнтів та технік приготування страв;

- інтелектуальну систему планування збалансованого харчування людини;

удосконалено:

- модель збалансованого харчування людини в напрямку врахування обмежень щодо інгредієнтів та технік приготування страв;

набули подальшого розвитку:

- технологія створення гібридних систем планування харчування в напрямку покращення адаптивності та ефективності роботи з великою кількістю даних в режимі реального часу за рахунок обґрунтованого вибору і окремого тренування моделей машинного навчання, які використовуються для формування рецептів страв;

- концепція створення гібридних систем планування харчування, що на відміну від існуючих, фокусуються на рекомендації плану харчування, а не пропозиції рецептів однієї страви.

Основний зміст дисертаційної роботи.

Дисертаційна робота присвячена задоволенню потреби багатьох людей в надійних системах, що забезпечують персоналізоване збалансоване планування харчування з урахуванням таких індивідуальних особливостей користувачів, як дієтичні обмеження, алергії, цілей щодо калорійності та балансу поживних речовин.

Запропоновано концептуальну модель інтелектуальної системи планування харчування людини і метод рекомендації рецептів страв. Запропоновано і протестовано системи рекомендації, що призначені для вирішення задачі формування меню блюд різних категорій прийому їжі в системі планування персоналізованого меню. Для систем рекомендацій обґрунтовано вибір таких моделей машинного навчання, як модель сингулярного розкладання матриці і модель широкого та глибокого навчання. Результати навчання кожної з цих систем рекомендацій страв кожного прийому їжі показали, що використання окремих моделей для різних категорій прийомів їжі підвищує точність рекомендацій.

За результатами дослідження розроблено:

- метод рекомендації рецептів, що дозволяє враховувати персональні вподобання користувача та його обмеження щодо інгредієнтів та технік приготування страв;
- інтелектуальну систему планування збалансованого харчування людини.

У першому розділі «Аналіз сучасних систем рекомендацій і планування харчування людини» висвітлено основні нюанси поняття «рекомендаційна система»; проведено аналіз сучасних систем рекомендацій і планування харчування людини; проаналізовано сучасні підходи до побудови систем, які здатні полегшити вибір меню і планування харчування людини; показано, що сучасні системи планування харчування переважно орієнтовані на універсальні рішення і часто не враховують індивідуальні особливості користувачів, що призводить до недостатньої ефективності таких систем; з'ясовано, що ефективне планування збалансованого харчування людини стає дедалі актуальнішою проблемою багатьох людей в умовах сучасного способу життя; обґрунтовано використання Precision@K для оцінки ефективності систем рекомендацій; за результатами аналізу проблем і недоліків існуючих систем

рекомендацій і планування харчування визначено напрями їх покращення; визначено план розробки інтелектуальної системи планування збалансованого харчування людини на основі сучасних методів машинного навчання.

У другому розділі «Моделі та методи побудови інтелектуальної системи планування харчування» показано узагальнену модель інтелектуальної системи персоналізованого планування збалансованого раціону харчування, при розробці якої основна увага приділялась послідовному формуванню і адаптації індивідуального плану харчування користувача; детально описано і проаналізовано процес планування збалансованого харчування; формалізовано модель збалансованого харчування людини, яка враховує індивідуальні дієтичні та харчові обмеження користувачів; модель побудована як правило-базована логіка і описує допустиму множину рецептів, що формують меню на день; запропоновано критерії визначення відповідності рецепту дієтичним обмеженням користувача; запропоновано і детально описано кожен етап методу рекомендації рецептів, що базується на гібридному підході, інтегрує персональні вподобання і дієтичні обмеження користувачів та враховує особливості приготування страв різних категорій проїму їжі; досліджено існуючі моделі штучного інтелекту та методи машинного навчання, що можуть бути використані в системі для рекомендації рецептів.

У третьому розділі «Розробка концептуальної моделі та архітектури системи планування збалансованого харчування людини» запропоновано концептуальну модель загальнодоступної інтелектуальної системи планування збалансованого харчування людини, що використовує окремо навчені моделі машинного навчання, спеціалізовані для сніданків, обідів, перекусів і вечерь; описано архітектуру системи; розглянуто життєвий цикл моделей систем рекомендацій; обґрунтовано вибір інструментів для розробки системи і проведено аналіз необхідних для реалізації запропонованої системи компонент; описано схему бази даних системи, модуль формування плану харчування, що перетворює вихід методу рекомендацій рецептів на цілісний меню план для заданого періоду часу, механізми зворотного зв'язку від користувача і вирішення проблеми холодного старту.

У четвертому розділі «Експериментальне дослідження і впровадження моделей машинного навчання в систему планування харчування» описано дані для експериментального дослідження, методологію і середовище експериментів; проаналізовано результати експериментального дослідження перевірки гіпотези щодо доцільності використання окремо навчених моделей машинного навчання для рекомендації різних категорій прийомів їжі; обґрунтовано доцільність використання моделі сингулярного розкладання матриці для генерації меню сніданків і моделі широкого та глибокого навчання для генерації меню перекусів, обідів і ВЕЧЕРЬ; представлені опис роботи розробленої системи планування харчування та надані інструкції щодо взаємодії користувача з системою.

Ключові слова: багатовимірна модель даних, гібридна модель, інформаційна система, інформаційна підтримка, інформаційна технологія, машинне навчання, мікросервісна архітектура, нейронна мережа, обробка даних, персоналізоване планування, програмне забезпечення, система підтримки рішень, система рекомендації, штучний інтелект.

ABSTRACT

Ladyzhets V.I. Intelligent system for planning a balanced human diet. – Qualifying scientific work presented as a manuscript.

Dissertation for obtaining the scientific degree of Doctor of Philosophy in specialty 126 "Information systems and technologies". – Kyiv National University of Construction and Architecture. – Kyiv, 2025.

The dissertation is focused on solving the problem of providing people with a reliable, scientifically sound, effective, and universally accessible system for personalized balanced nutrition planning.

Scientific novelty of the obtained results. The scientific novelty of the dissertation consists in the fact that:

developed for the first time:

- The method for recommending recipes that takes into account the user's personal preferences and restrictions on ingredients and cooking techniques;
- The intelligent system for planning a balanced diet for humans; improved;

improved:

- The model of a balanced diet for humans that takes into account restrictions on ingredients and cooking techniques;

further advanced:

- The technology for creating hybrid diet planning systems aimed at improving adaptability and efficiency when working with large amounts of data in real time through the informed selection and separate training of machine learning models used to generate recipes;
- The concept for creating hybrid diet planning systems which, unlike existing ones, focus on meal plan recommendations rather than single dish recipes;

Main content of the dissertation.

The dissertation is dedicated to meeting the needs of many people for reliable systems that provide personalized balanced meal planning, taking into account such individual characteristics of users as dietary restrictions, allergies, calorie and nutrient balance goals.

The conceptual model of an intelligent human nutrition planning system and a method for recommending recipes are proposed. Recommendation systems designed to solve the problem of forming menus of different food categories in a personalized menu planning system are proposed and tested. For recommendation systems, the choice of machine learning models such as the singular matrix decomposition model and the broad and deep learning model is justified. The results of training each of these meal recommendation systems for each meal type showed that using separate models for different meal categories improves the accuracy of recommendations.

Based on the research results, the following were developed:

- The recipe recommendation method that allows for the user's personal preferences and restrictions on ingredients and cooking techniques to be taken into account;
- The intelligent system for planning a balanced diet for humans.

The first chapter “Analysis of modern diet nutrition recommendation and planning systems” highlights the main nuances of the concept of a “recommendation system”; analyzes modern human diet recommendation and planning systems; analyzes modern approaches to building systems that can facilitate menu selection and human diet planning; it is shown that modern diet planning systems are mainly focused on universal solutions and often do not take into account the individual characteristics of users, which leads to the insufficient effectiveness of such systems; it is found that effective planning of a balanced human diet is becoming an increasingly pressing issue for many people in the context of modern lifestyles; the use of Precision@K for evaluating the effectiveness of recommendation systems has been justified; based on the analysis of problems and shortcomings of existing recommendation and meal planning systems, directions for their improvement have been identified; a plan for developing an intelligent system for planning a balanced human diet based on modern machine learning methods has been defined.

The second chapter «Models and methods for developing an intelligent meal planning system» presents a generalized model of an intelligent system for personalized balanced diet planning, which was developed with a focus on the sequential formation and adaptation of an individual user diet plan. The process of balanced diet planning is described and analyzed in detail, and a formalized model of a balanced human diet is presented, which

takes into account the individual dietary and nutritional restrictions of users; the model is built as rule-based logic and describes the permissible set of recipes that form the daily menu; criteria for determining the compliance of a recipe with the user's dietary restrictions are proposed; each stage of the recipe recommendation method is proposed and described in detail, based on a hybrid approach, integrating personal preferences and dietary restrictions of users and taking into account the peculiarities of cooking dishes of different food categories; existing artificial intelligence models and machine learning methods that can be used in the system for recipe recommendation have been investigated.

The third chapter “Development of Conceptual Model and Architecture of a System for Planning a Balanced Human Diet,” proposes a conceptual model of a publicly available intelligent system for planning a balanced human diet that uses separately trained machine learning models specialized for breakfast, lunch, snacks, and dinner; describes the system architecture; considers the life cycle of recommendation system models; justifies the choice of tools for developing the system and analyzes the proposals necessary for implementation; the database scheme, the module for generating a meal plan that transforms the output of the recipe recommendation method into a complete menu plan for a given period, user feedback mechanisms and solutions to the problem of cold start are described.

The fourth chapter “Experimental research and implementation of machine learning models in the meal planning system,” describes the data for the experimental research, the methodology, and the experimental environment; analyzes the results of the experimental research to test the hypothesis regarding the feasibility of using separately trained machine learning models to recommend different categories of meals; justifies the feasibility of using a singular matrix decomposition model to generate breakfast menus and a broad and deep learning model to generate snack, lunch, and dinner menus; describes the operation of the developed meal planning system and provides instructions for user interaction with the system.

Keywords: artificial intelligence, data processing, decision support system, hybrid model, information support, information system, information technology, machine learning, microservice architecture, multidimensional data model, neural network, personalized planning, recommendation system, software.

СПИСОК ОПУБЛІКОВАНИХ НАУКОВИХ ПРАЦЬ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

*Статті у наукових періодичних виданнях інших держав та у виданнях
України, які включено до міжнародних наукометричних баз*

1. **Ладижець В. І.**, Теренчук С. А. Моделі, методи і засоби планування раціону харчування людини. *Управління розвитком складних систем*. Київ, 2023. № 53. С. 39 – 44 DOI: <https://doi.org/10.32347/2412-9933.2023.53.39-44> [Фахове видання]
2. **Ладижець, В.**, & Теренчук, С. Проектування методу рекомендації для інформаційної системи планування харчування. *Шляхи підвищення ефективності будівництва*, №53 (2), С. 377–388. DOI: [https://doi.org/10.32347/2707-501x.2024.53\(2\).377-388](https://doi.org/10.32347/2707-501x.2024.53(2).377-388) [Фахове видання]
3. **Ладижець, В. І.**, Теренчук, С. А., Азнаурян, І. О., & Махиня, А. А. Вдосконалення інтелектуальних систем підтримки рішень щодо планування збалансованого раціону харчування. *Eastern-European Journal of Enterprise Technologies*, 2025, 1(3 (133), 37–47. DOI: <https://doi.org/10.15587/1729-4061.2025.322316> [Категорія «А», Scopus, Q3]

Наукові праці, що представлені як тези доповіді у міжнародних науково-технічних конференціях

4. **V. Ladyzhets, B. Yeremenko & S. Terenchuk**, "Candidate Generation for Meal Recommendation System" *2024 IEEE 4th International Conference on Smart Information Systems and Technologies (SIST)*, May, Astana, Kazakhstan, 2024, pp. 560-564 DOI: <https://doi.org/10.1109/SIST61555.2024.10629517> [Scopus, ISBN:979-835033504-0]

Тези доповідей наукових та науково-практичних конференцій

5. **Ладижець В.І.**, Теренчук, С.А. Інтелектуальна система підтримки прийняття рішень для збалансованого харчування людини: матрична факторизація. *Development of Education, Science and Business: Results 2023: Proceedings of the International Scientific and Practical Internet Conference*, December 21-22, 2023. FOP

Marenichenko V.V., Dnipro, Ukraine, pp. 108-113 <http://www.wayscience.com/wp-content/uploads/2023/12/Conference-Proceedings-December-21-22-2023.pdf>

6. **Ладижець, В.,** Теренчук, С. Перспективи використання глибинного навчання при вирішенні питань збалансованого харчування. *Перспективні напрямки розвитку науки, освіти, технологій та суспільства: теорія і практика: збірник тез доповідей міжнародної науково-практичної конференції*, Полтава, 19 квітня 2023 р. ЦФЕНД, 2023. 55 с <http://www.economics.in.ua/2023/04/19-2023.html>

ЗМІСТ

ВСТУП	13
РОЗДІЛ 1. АНАЛІЗ СУЧАСНИХ СИСТЕМ РЕКОМЕНДАЦІЙ І ПЛАНУВАННЯ	
ХАРЧУВАННЯ ЛЮДИНИ.....	21
1.1 Трактовка поняття «рекомендаційна система».....	21
1.2 Сучасні інтелектуальні системи рекомендацій і планування харчування людини	26
1.3 План розробки інтелектуальної системи планування збалансованого харчування людини	33
Висновки до розділу 1	35
РОЗДІЛ 2. МОДЕЛІ ТА МЕТОДИ ПОБУДОВИ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПЛАНУВАННЯ ХАРЧУВАННЯ	36
2.1 Моделювання інтелектуальної системи персоналізованого планування збалансованого раціону харчування	36
2.2 Формальна модель збалансованого харчування	40
2.3 Персоналізація в системах планування харчування	46
2.3.1 Класичні методи.....	47
2.3.2 Методи на основі дерев прийняття рішень	52
2.3.3 Методи на основі нейронних мереж	56
2.4 Метод рекомендацій рецептів.....	65
Висновки до розділу 2	75
РОЗДІЛ 3. РОЗРОБКА КОНЦЕПТУАЛЬНОЇ МОДЕЛІ ТА АРХІТЕКТУРИ СИСТЕМИ ПЛАНУВАННЯ ЗБАЛАНСОВАНОГО ХАРЧУВАННЯ ЛЮДИНИ	76
3.1 Концептуальна модель системи	76
3.2 Архітектура інтелектуальної системи планування збалансованого харчування	79
3.3 Вибір інструментів для розробки системи.....	83
3.4 Життєвий цикл моделі машинного навчання для методу рекомендацій.....	85

3.5	Схема бази даних	90
3.6	Модуль формування плану харчування.....	93
3.7	Реалізація механізмів зворотного зв'язку користувачів	95
3.8	Вирішення проблеми холодного старту.....	99
	Висновки до розділу 3	104
РОЗДІЛ 4. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ І ВПРОВАДЖЕННЯ МОДЕЛЕЙ МАШИНОГО НАВЧАННЯ В СИСТЕМУ ПЛАНУВАННЯ ХАРЧУВАННЯ.....		105
4.1.	Опис даних для експериментального дослідження	105
4.2.	Методологія та середовище експериментів	108
4.3.	Параметри моделей	110
4.4	Експериментальне дослідження моделей для кожного прийому їжі	112
4.4	Обґрунтування вибору моделей систем рекомендації рецептів	118
4.5	Результати та аналіз експериментального дослідження моделей-кандидатів	120
4.6	Взаємодія користувача з системою.....	121
	Висновки до розділу 4	126
ВИСНОВКИ.....		127
СПИСОК ЛІТЕРАТУРИ.....		128
ДОДАТКИ.....		140

ВСТУП

Актуальність теми. Результати чисельних досліджень проблем, які пов'язані з режимом харчування сучасних людей свідчать про те, що незбалансоване харчування та недотримання медичних дієтичних рекомендацій суттєво впливають на якість і тривалість життя сучасної людини.

Дослідження, які були проведені за даними, зібраними в США і Австралії, показують як надмірна вага впливає на ризик серцево-судинних захворювань, а автори цих досліджень припускають, що спостережуване зростання відсотка молоді з надмірною вагою є соціальною проблемою, яка може стати набагато глибшою в найближчі роки. Статистичні дані про зростаючий відсоток людей з надмірною вагою в ЄС, Азії і в усьому світі також свідчать про нагальну необхідність впровадження ефективних рішень щодо збалансованого харчування.

Це дослідження є своєчасним і важливим, оскільки воно узгоджується з глобальними ініціативами в галузі охорони здоров'я, що спрямовані на використання технологій для покращення стану здоров'я. В той же час, наявні інструменти і додатки для планування харчування, досі не надають персоналізованих і адаптивних рекомендацій, що відповідають індивідуальним потребам і вподобанням багатьох людей. Саме тому лишається актуальною тема розробки інтелектуальних систем рекомендації, які навчається на даних, що характеризують користувача, щоб представити набір страв, які можуть бути йому корисними і цікавими для планування харчування.

Актуальність дослідження полягає у вирішенні критичної прогалини: потреби в персоналізованій системі, яка не лише пропонуватиме страви на основі вподобань користувача, а й буде враховувати індивідуальні обмеження щодо стану здоров'я. До того хочеться відзначити, що особисто автора до проведення наукових досліджень в напрямку створення інтелектуальної системи планування збалансованого харчування людини спонукала зростаюча тенденція та яскраво виражене тяжіння молоді до питань піклування про своє здоров'я та самостійного пошуку рішення щодо врегулювання особистого режиму харчування.

Робота зосереджена на вирішенні проблеми розробки надійних, доступних і ефективних персоналізованих систем рекомендації для планування збалансованого харчування людини, що здатні враховувати його вподобання і дієтичні обмеження користувача. При цьому встановлено доцільність використання в системах рекомендацій рецептів страв таких моделей машинного навчання, як модель сингулярного розкладання матриці для генерації меню сніданків та модель широкого та глибокого навчання для генерації меню перекусів, обідів і вечерь. Такий результат пояснюється тим, що різні моделі оптимально працюють з різними типами даних.

Передбачається, що за наявності достатніх достовірних даних про вподобання, алергії, дієтичні обмеження і потреби користувача, запропонована система планування збалансованого харчування може значно покращити передбачуваність його попиту. Але урахування подібних обмежень наразі потребує тестування систем рекомендацій на ретельно згрупованих за відповідними ознаками.

Зв'язок роботи з науковими програмами, планами, темами. Наукова робота відповідає завданням Стратегії реформування системи шкільного харчування на період до 2027 року, схваленої розпорядженням Кабінету Міністрів України від 27 жовтня 2023 р. № 990-р, в частині запровадження доступних та ефективних інтелектуальних систем, які дозволяють формувати плани збалансованого, якісного, безпечного та повноцінного харчування з урахуванням норм, індивідуальних особливостей, стану здоров'я та харчових уподобань і відповідає напрямку, визначеному у Законі України «Про схвалення Концепції розвитку штучного інтелекту в Україні» (2020 р.),

Дисертація відповідає паспорту спеціальності 126 – Інформаційні системи та технології

Дисертаційне дослідження виконане на кафедрі інформаційних технологій проектування та прикладної математики факультету автоматизації і інформаційних технологій Київського національного університету будівництва і архітектури

Результати дослідження впроваджено в навчальний процес в межах дисциплін на спеціальності 122 та 126 факультету автоматизації і інформаційних технологій

Київського національного університету будівництва і архітектури, зокрема, в навчальні програми дисциплін:

– «Теорія алгоритмів» для магістрів спеціальності «Інформаційні системи та технології. Штучний інтелект. Когнітивні технології» кафедри управління проєктів і кафедри інформаційних технологій проєктування та прикладної математики;

– «Інформаційні технології представлення обробки та розпізнавання зображень» для магістрів спеціальності «Інформаційні системи та технології» кафедри інформаційних технологій проєктування та прикладної математики;

– «Прикладна теорія графів» для магістрів спеціальності «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» кафедри автоматизації технологічних процесів.

Дисертація містить наукові положення, нові науково обґрунтовані теоретичні результати проведених досліджень, які мають істотне значення для галузі знань 12 – Інформаційні технології.

Мета дослідження – розробка інтелектуальної системи планування збалансованого раціону харчування людини, що орієнтована на передбачення персонального попиту користувача.

Для досягнення поставленої мети було виділено такі завдання:

1. Провести системний аналіз існуючих систем рекомендацій і планування харчування людини та визначити напрямки їх покращень.

2. Сформувати модель збалансованого харчування людини з урахуванням дієтичних обмежень.

3. Розробити метод рекомендації рецептів, що може бути використаний інтелектуальною системою планування збалансованого харчування людини.

4. Запропонувати концептуальну модель інтелектуальної системи планування збалансованого харчування людини, що поєднує високу персоналізацію та адаптивність до особистих вподобань і обмежень користувача.

5. Обґрунтувати і експериментально підтвердити вибір моделей машинного навчання для методу рекомендації рецептів страв різних категорій прийомів їжі.

6. Розробити інтелектуальну систему планування збалансованого харчування що використовує запропонований метод рекомендації рецептів і враховує обмеження користувача згідно з вдосконаленою моделлю збалансованого харчування людини.

Об'єкт дослідження – інтелектуальні системи рекомендацій і планування.

Предмет дослідження – інтелектуальні системи планування, що використовують моделі машинного навчання.

Наукова новизна одержаних результатів. Наукова новизна дисертаційної роботи полягає у тому, що:

вперше розроблено:

– метод рекомендації рецептів, що дозволяє враховувати персональні вподобання користувача та його обмеження щодо інгредієнтів та технік приготування страв;

– інтелектуальну систему планування збалансованого харчування людини;

удосконалено:

– модель збалансованого харчування людини в напрямку врахування обмежень щодо інгредієнтів та технік приготування страв;

набули подальшого розвитку:

– технологія створення гібридних систем планування харчування в напрямку покращення адаптивності та ефективності роботи з великою кількістю даних в режимі реального часу за рахунок обґрунтованого вибору і окремого тренування моделей машинного навчання, які використовуються для формування рецептів страв;

– концепція створення гібридних систем планування харчування, що на відміну від існуючих, фокусуються на рекомендації плану харчування, а не пропозиції рецептів однієї страви.

Практичне значення отриманих результатів.

Результати цієї роботи можуть бути використані для розробки надійних, науково обґрунтованих, високоефективних і загальнодоступних персоналізованих сервісів планування харчування в мобільних додатках і онлайн-платформах, пропонуючи реальне вирішення нагальних проблем громадського здоров'я, що пов'язані з управлінням раціоном та харчуванням.

Основний зміст дисертаційної роботи.

Дисертаційна робота присвячена задоволенню потреби багатьох людей в надійних системах, що забезпечують персоналізоване збалансоване планування харчування з урахуванням таких індивідуальних особливостей користувачів, як дієтичні обмеження, алергії, цілей щодо калорійності та балансу поживних речовин.

Запропоновано концептуальну модель інтелектуальної системи планування харчування людини і метод рекомендації рецептів страв. Запропоновано і протестовано системи рекомендації, що призначені для вирішення задачі формування меню блюд різних категорій прийому їжі в системі планування персоналізованого меню. Для рекомендаційних систем обґрунтовано вибір таких моделей машинного навчання, як модель сингулярного розкладання матриці і модель широкого та глибокого навчання. Результати навчання кожної з цих систем показали, що використання окремих моделей машинного навчання для різних категорій прийомів їжі підвищує точність рекомендацій.

За результатами дослідження розроблено:

- метод рекомендації рецептів, що дозволяє враховувати персональні вподобання користувача та його обмеження щодо інгредієнтів та технік приготування страв;

- інтелектуальну систему планування збалансованого харчування людини.

У першому розділі «Аналіз сучасних систем рекомендацій і планування харчування людини» висвітлено основні нюанси поняття «рекомендаційна система»; проведено аналіз сучасних систем рекомендацій і планування харчування людини; проаналізовано сучасні підходи до побудови систем, які здатні полегшити вибір меню і планування харчування людини; показано, що сучасні системи планування харчування переважно орієнтовані на універсальні рішення і часто не враховують індивідуальні особливості користувачів, що призводить до недостатньої ефективності таких систем; з'ясовано, що ефективне планування збалансованого харчування людини стає дедалі актуальнішою проблемою багатьох людей в умовах сучасного способу життя; обґрунтовано використання Precision@K для оцінки ефективності систем рекомендацій; за результатами аналізу проблем і недоліків існуючих систем

рекомендацій і планування харчування визначено напрями їх покращення; визначено план розробки інтелектуальної системи планування збалансованого харчування людини на основі сучасних методів машинного навчання.

У другому розділі «Моделі та методи побудови інтелектуальної системи планування харчування» показано узагальнену модель інтелектуальної системи персоналізованого планування збалансованого раціону харчування, при розробці якої основна увага приділялась послідовному формуванню і адаптації індивідуального плану харчування користувача; детально описано і проаналізовано процес планування збалансованого харчування; формалізовано модель збалансованого харчування людини, яка враховує індивідуальні дієтичні та харчові обмеження користувачів; модель побудована як правило-базована логіка і описує допустиму множину рецептів, що формують меню на день; запропоновано критерії визначення відповідності рецепту дієтичним обмеженням користувача; запропоновано і детально описано кожен етап методу рекомендації рецептів, що базується на гібридному підході, інтегрує персональні вподобання і дієтичні обмеження користувачів та враховує особливості приготування страв різних категорій проїму їжі; досліджено існуючі моделі штучного інтелекту та методи машинного навчання, що можуть бути використані в системі для рекомендації рецептів.

У третьому розділі «Розробка концептуальної моделі та архітектури системи планування збалансованого харчування людини» запропоновано концептуальну модель загальнодоступної інтелектуальної системи планування збалансованого харчування людини, що використовує окремо навчені моделі машинного навчання, спеціалізовані для сніданків, обідів, перекусів і вечерь; описано архітектуру системи; розглянуто життєвий цикл моделей систем рекомендацій; обґрунтовано вибір інструментів для розробки системи і проведено аналіз необхідних для реалізації запропонованої системи компонент; описано схему бази даних системи, модуль формування плану харчування, що перетворює вихід методу рекомендацій рецептів на цілісний меню план для заданого періоду часу, механізми зворотного зв'язку від користувача і вирішення проблеми холодного старту.

У четвертому розділі «Експериментальне дослідження і впровадження моделей машинного навчання в систему планування харчування» описано дані для експериментального дослідження, методологію і середовище експериментів; проаналізовано результати експериментального дослідження перевірки гіпотези щодо доцільності використання окремо навчених моделей машинного навчання для рекомендації різних категорій прийомів їжі; обґрунтовано доцільність використання моделі сингулярного розкладання матриці для генерації меню сніданків і моделі широкого та глибокого навчання для генерації меню перекусів, обідів і ВЕЧЕРЬ; представлені опис роботи розробленої системи планування харчування та надані інструкції щодо взаємодії користувача з системою.

Ключові слова: багатовимірна модель даних, гібридна модель, інформаційна система, інформаційна підтримка, інформаційна технологія, машинне навчання, мікросервісна архітектура, нейронна мережа, обробка даних, персоналізоване планування, програмне забезпечення, система підтримки рішень, система рекомендації, штучний інтелект.

Особистий внесок здобувача. Дисертація є самостійною науковою працею, в якій висвітлені власні ідеї та розробки автора, що дозволили досягти поставленої мети. З наукових праць, опублікованих у співавторстві, у дисертації використано лише ті ідеї та положення, які є результатом особистої роботи здобувача.

Апробація результатів дисертації. Основні положення та результати дисертаційної роботи висвітлені та обговорені на наступних конференціях:

1. Міжнародна науково-практична конференція "Перспективні напрямки розвитку науки, освіти, технологій та суспільства: теорія і практика", м. Полтава. 19 квітня 2023.

2. Development of Education, Science and Business: Results 2023: Proceedings of the International Scientific and Practical Internet Conference, December 21-22, 2023. FOP Marenichenko V.V., Dnipro, Ukraine

3. 2024 IEEE 4th International Conference on Smart Information Systems and Technologies (SIST), Astana, Kazakhstan, 15-17 May 2024. 2024

В повному обсязі дисертація доповідалась на науковому семінарі кафедри інформаційних технологій, кафедри інформаційних технологій проектування та прикладної математики Київського національного університету будівництва і архітектури (м. Київ, 2025 рік).

Структура та обсяг дисертації. Дисертаційна робота складається з вступу, чотирьох розділів, загальних висновків, списку використаних джерел і додатків. Загальний обсяг роботи становить 141 сторінку, у тому числі, основна частина складає 115 сторінок. Основна частина, крім тексту, включає 7 таблиць і 22 рисунки.

РОЗДІЛ 1. АНАЛІЗ СУЧАСНИХ СИСТЕМ РЕКОМЕНДАЦІЙ І ПЛАНУВАННЯ ХАРЧУВАННЯ ЛЮДИНИ

У розділі висвітлено основні нюанси поняття «рекомендаційна система»; проведено аналіз сучасних систем рекомендацій і планування харчування людини; проаналізовано сучасні підходи до побудови систем, які здатні полегшити вибір меню і планування харчування людини; показано, що сучасні системи планування харчування переважно орієнтовані на універсальні рішення і часто не враховують індивідуальні особливості користувачів, що призводить до недостатньої ефективності таких систем; з'ясовано, що ефективне планування збалансованого харчування людини стає дедалі актуальнішою проблемою багатьох людей в умовах сучасного способу життя; обґрунтовано використання Precision@K для оцінки ефективності систем рекомендацій; за результатами аналізу проблем і недоліків існуючих систем рекомендацій і планування харчування визначено напрями їх покращення; визначено план розробки інтелектуальної системи планування збалансованого харчування людини на основі сучасних методів машинного навчання.

1.1 Трактовка поняття «рекомендаційна система»

Одним з основних концептів домену «рекомендаційне планування» є поняття «рекомендаційна система» або «система рекомендацій». Цей термін з'явився в середині 1990-х років для визначення системи, що доповнює природні соціальні процеси рекомендацій, основна роль яких полягає в агрегуванні думок користувачів або підборі відповідних рекомендацій для користувачів [1]. Проте в міру розвитку галузі «персоналізована рекомендація» це визначення вийшло за межі суто збору в агрегації вхідної інформації для соціального обміну рекомендаціями користувачів.

У [2] для визначення системи рекомендацій запропоновано ширше поняття: «будь-яка система, яка генерує індивідуальні рекомендації як результат або має ефект

персоналізованого направлення користувача до цікавих або корисних об'єктів у великому просторі можливих варіантів».

Це визначення розширює сферу застосування до будь-якого персоналізованого інструменту фільтрування, незалежно від техніки, що використовується для генерації рекомендацій, передбачаючи, що система може вважатися системою рекомендацій, якщо вона направляє кожного користувача до корисних об'єктів персоналізованим чином за допомогою [2]:

- спільного фільтрування;
- методів на основі вмісту;
- алгоритмів на основі знань;
- чи інших засобів.

При цьому акцент у [2] робиться на персоналізації та допомозі користувачам орієнтуватися у великому просторі варіантів, а не на необхідності явного соціального втручання.

У той же період почалась формалізація поняття «система рекомендацій». В [3] описано проблему рекомендацій у математичних термінах як пошук для кожного користувача елемента чи набору елементів, який максимізує функцію корисності цього користувача серед усіх можливих елементів.

Формально згідно з [3]: якщо « C – набір користувачів, S – набір усіх елементів, а $u(c,s)$ – функція корисності, що вимірює корисність елемента s для користувача c », то «мета системи рекомендувати елемент $s' \in S$, який максимізує $u(c,s')$ для кожного користувача c ».

Це формальне тлумачення охоплює будь-яке застосування, що обчислює корисність або переваги конкретного користувача, проте не визначає, як саме обчислюється корисність [4]:

- чи за допомогою явних оцінок;
- чи за допомогою неявної поведінки;
- чи за допомогою будь-яких алгоритмічних засобів.

Також в [3] вказано, що таке широке визначення охоплює навіть деякі системи, які традиційно розглядаються як пошук інформації або запити до баз даних, якщо

вони персоналізують результати для користувача, і визнаються крайні випадки (такі як групові рекомендації чи динамічно генеровані елементи), які дещо суперечать поняттю фіксованого набору елементів для кожного користувача, що відображає все більш інклюзивне розуміння того, чим можуть бути системи рекомендацій [4].

Приблизно в цей час рекомендаційні системи стали розглядати як «інструменти фільтрації, що персоналізовано направляють користувача до цікавих або корисних об'єктів у великому просторі можливих варіантів» [5] і основна термінологія змістилась в бік позиціонування терміна «рекомендаційна система», як спеціалізованої форми системи фільтрування інформації, що відрізняється персоналізованим результатом. При цьому більшість дослідників підкреслюють орієнтованість систем рекомендацій на користувача.

У [6] системи рекомендацій описують як інструменти, що «допомагають користувачам, визначаючи цікаві продукти та послуги в ситуаціях, коли кількість і складність пропозицій перевищують можливості користувача їх оглянути та прийняти рішення» [4].

Це визначення підкреслює, що рекомендаційні системи виконують функцію підтримки прийняття рішень. А саме: фільтруючи і пріоритезуючи варіанти, ці системи дозволяють користувачам сприйняти велику кількість альтернатив, спрощуючи прийняття рішень шляхом зменшення когнітивного навантаження.

Відповідно до цього:

- у [7] системи рекомендацій визначається як «програмні засоби та методи, що надають пропозиції щодо елементів, які можуть бути корисними для користувача»;

- у [8], системи рекомендацій визначається як «програмні засоби та методи, що надають пропозиції щодо елементів, які можуть бути цікавими для користувача. Надані пропозиції спрямовані на підтримку процесів прийняття рішень».

У цих визначеннях сфера застосування не залежить від домену а спільним є персоналізація та можливість дії (підтримка рішень користувача).

При цьому в [4] системи рекомендацій визначаються як «інструменти для взаємодії з великими та складними інформаційними просторами», які «забезпечують

персоналізований перегляд таких просторів, надаючи пріоритет предметам, які можуть зацікавити користувача».

Таке розуміння перетворює систему рекомендацій на інтелектуального посередника між користувачем і тим, що він бачить в інформаційному просторі, адаптуючи видиме відповідно до зазначених користувачем переваг. Таке визначення, на відміну від стандартного пошукового двигуна, узгоджує поняття «рекомендаційні системи» з продуктами пошуку чи фільтрування інформації, але з чітким ухилом на персоналізацію відповіді системи для різних користувачів при однакових запитах [4].

Інші автори визначають системи рекомендацій фокусуючись на фільтруванні інформації, що пов'язує системи рекомендацій з класом систем, які зменшують інформаційне навантаження шляхом фільтрування вмісту за релевантністю [9].

Таким чином «рекомендаційна система» на теперішній час є терміном, що узагальнює нюанси різних понять, визначаючи обов'язковий елемент адаптованості системи до різних користувачів [4] і таких ключових відмінностей, як «рекомендація» проти «фільтрування» чи «допомога у прийнятті рішення» залежно від акцентів, при чому [7]:

- «система фільтрування» спрямована на зменшення потоку інформації до релевантного підмножини;
- «система персоналізації» спрямована на адаптацію до унікальних уподобань користувача;
- «консультаційна система» підкреслює функцію автоматичного генерування пропозицій.

Відмінність персоналізованих систем рекомендації від систем прогнозування полягає у тому, що перші повертають рекомендації в порядку спадання ймовірності вподобання користувачем рекомендованого елемента [10].

В цій роботі поняття «рекомендаційна система» розуміємо як програмний інструмент фільтрування та ранжування, який навчається на даних, що характеризують користувача, щоб представити набір страв, які можуть бути йому корисними і цікавими для планування харчування на певних період [9].

Для оцінки ефективності систем рекомендацій зазвичай використовують $Precision@K$ чи $Recall@K$ [10].

$Recall@K$ показує, яку частину з усього можливого релевантного набору система змогла знайти серед цих рекомендацій і використовується в системах, які генерують індивідуальні рекомендації.

$Precision@K$ показує який відсоток із перших K рекомендацій є релевантним, тобто відповідає інтересам користувача і таким чином надає оцінку ефективності роботи систем рекомендацій за часткою релевантних елементів у перших K рекомендаціях, представлених користувачеві системою [10, 11]:

$$P = NR/K, \quad (1.1)$$

де: NR – кількість релевантних рекомендацій; K – кількість найкращих рекомендацій.

Релевантними вважаються елементи, оцінки яких в тестовій підмножині перевищують заданий поріг, а $Precision@K$ фокусується на тому, скільки з найкращих рекомендацій K є дійсно релевантними, таким чином відображаючи здатність системи надавати точні та корисні поради.

Таким чином, $Precision@K$ пропонує більш пряме вимірювання практичної корисності персоналізованої системи рекомендацій порівняно з MAE і RMSE. Зосереджуючись на частці правильно знайдених релевантних елементів у K топ-рекомендаціях, $Precision@K$ тісніше узгоджується з реальними процесами прийняття рішень, тим самим забезпечуючи більш чіткий показник ефективності системи за відповідністю потребам і очікуванням користувачів.

Оскільки користувачі, як правило, звертають увагу тільки на кілька найкращих пропозицій, важливіше забезпечити високу якість цих пропозицій. Таким чином $Precision@K$ імітує реальну поведінку користувача, що зазвичай розглядає лише найперші рекомендації [11, 12].

Такий підхід сприяє покращенню досвіду користувача та позитивно впливає на бізнес-показники, адже перші рекомендації є ключовими для залучення та конверсії

клієнтів. Орім того, у реальних умовах дані про релевантність часто є неповними, що робить $Precision@K$ більш практичною метрикою. Саме тому в цій роботі основним критерієм оцінки ефективності систем рекомендації меню вибрано $Precision@K$.

1.2 Сучасні інтелектуальні системи рекомендацій і планування харчування людини

Серед сучасних досліджень існує кілька підходів до побудови систем, які здатні полегшити вибір меню і планування харчування.

Залежно від підходів системи планування збалансованого харчування поділяють на такі, що використовують:

- великі мовні моделі (LLM);
- підходи на основі експертних знань;
- гібридні моделі та колаборативну фільтрацію;
- моделі та методи машинного навчання (ML);
- глибинне навчання (DL),
- моделі глибинного навчання Wide&Deep (W&D).

В роботі [13] представлено систему CORA, що використовує великі мовні моделі (LLM) для побудови діалогової системи з пам'яттю для рекомендації страв. CORA формує список інгредієнтів для страв, які можуть найбільше сподобатись користувачеві, за частотою споживання певних страв в останні кілька днів з урахуванням найкращої і найгіршої оцінки. Вибір страви з бази даних системи здійснюється на основі відповідей користувача на певні питання системи, що забезпечує зворотний зв'язок. Також в [13] наведено результати дослідження доцільності використання LLM для рекомендації страв і показано, що вони можуть забезпечити прийнятний рівень задоволеності користувачів рекомендаціями та зручністю користування.

Однак в [13] залишилися невирішеними питання, що пов'язані з:

- обмеженим використанням особистих даних користувача внаслідок його небажання розкривати детальну інформацію про свій режим харчування;

– індивідуальними відмінностями в стилях спілкування користувачів.

Ці обмеження і відмінності призводять до формування неповних профілів і недостатньо якісних рекомендацій.

Варіантом вирішення цієї проблеми, може бути опосередкований збір та аналіз даних про користувача, без запитування цих даних в діалоговому форматі.

В [14] представлено і досліджено саме такий варіант рішення.

В цій роботі використано:

- варіаційний автокодер для трансформації профілю користувача;
- рекурентну нейронну мережу для генерації рекомендацій;
- LLM ChatGPT для покращення згенерованих рекомендацій, розширення можливих страв і пропозиції альтернатив.

Результати дослідження [14] показали, що LLM може покращувати точність систем рекомендації, проте в ході дослідження також відзначається, що LLM схильні до помилок і неточностей щодо калорійності їжі, порцій та наявності алергенів у страві. Ці помилки і неточності спричинені тим, що на теперішній час не створені LLM, які могли б суттєво знизити рівень невірних відповідей при генерації і валідувати свої результати. Альтернативним варіантом подолання зазначених помилок і неточностей LLM є застосування адаптивних систем, що побудовано на базі експертних знань.

Підхід на базі експертних знань і правил для різних груп користувачів використано в [15]. В цій роботі представлено систему PROTEIN AI Advisor, яка формує план харчування враховуючи як дієтичні обмеження і вподобання користувача, так і визначені експертами правила.

В цій системі лишаються невирішеними питання, що пов'язані з:

- браком зворотного зв'язку від користувача;
- неможливістю адаптувати рекомендації до динаміки його вподобань;
- слабкою масштабованістю системи.

Проблема адаптації систем рекомендацій на базі експертних знань до динаміки попиту користувача вирішується шляхом використання моделей штучного інтелекту, що здатні враховувати зворотній зв'язок.

Такий підхід використано в [16] для розробки системи SHARE, яка поєднує:

- колаборативну фільтрацію для виявлення вподобань користувачів шляхом аналізу схожості між їхніми вподобаннями;
- фільтрацію на основі вмісту для рекомендацій користувачу рецептів, схожих на ті, які він вподобав раніше.

Таке поєднання дозволяє SHARE рекомендувати меню страв і готовий план харчування на тиждень, що відповідають персональним смакам користувачів, на основі колективних вподобань.

Дослідження проведені в [16] показали, що SHARE здатна адаптуватися до змін у вимогах користувача, використовуючи тільки історію вибору страв, яка зібрана опосередковано від самого користувача. Проте залишаються проблеми масштабованості і ефективності, що пов'язані з необхідністю маніпулювати великими наборами даних, яку в SHARE ускладнює застосування методу К-найближчих сусідів (KNN).

Використання KNN призводить до повільного формування рекомендацій і унеможливорює врахування [17]:

- таких прихованих факторів, як тонкі смакові вподобання чи дієтичні потреби, а саме ці фактори лежать в основі вподобань користувачів.
- такої контекстної і супутньої інформації, як демографічні дані користувачів і атрибути страв.

Також у дослідженні не враховано обмеження за кулінарними техніками, та досягається тільки обмежений рівень персоналізації.

Вирішенню проблеми роботи з великими наборами даних присвячено дослідження [18, 19]. В цих роботах для систем рекомендації протестовано модифіковану версію моделі KKN і модель Gated Recurrent Unit (GRU). Результати тестувань показали, що обидві моделі потребують подальших досліджень щодо покращення персоналізованості рекомендацій, ефективності і адаптивності.

В роботі [20] описано систему ChronicDietAI для планування харчування, яка враховує медичні обмеження, алергії та особливості харчування людей з хронічними захворюваннями. Система працює на основі експертних правил для забезпечення

обмежень, та використовує фільтрацію на основі вмісту та дерева прийняття рішень для персоналізації. Водночас система має обмеження щодо адаптивності персоналізації й відсутності можливостей оновлення рекомендацій у реальному часі, що знижує гнучкість використання. Також система орієнтована тільки на користувачів з хронічними хворобами, що зменшує її практичне застосування для ширшої категорії користувачів.

В роботі [21] запропоновано ElderNutriPlanner, що застосовує експертний підхід з використанням профілів користувачів і баз даних продуктів для створення збалансованих раціонів для літніх людей. Система використовує фільтрацію на основі вмісту для забезпечення виконання обмежень, проте має слабку персоналізацію та адаптивність, що не дозволяє враховувати вподобання користувача. Також система орієнтована тільки на людей похилого віку.

В [22] показано, що моделі глибинного навчання актуальні для задачі побудови систем рекомендацій завдяки своїй потужності в обробці великих обсягів різномірної інформації. Глибинні неймережі можуть навчатися високорівневим абстракціям, що забезпечує точні та персоналізовані рекомендації. Ця область активно розвивається, внаслідок чого з'являються нові архітектури і методи, що покращують продуктивність систем рекомендації.

В роботі [23], що спрямована на покращення структури систем рекомендації щодо харчування:

- запропоновано архітектуру системи рекомендації для генерації меню;
- протестовано модель Wide&Deep.

Wide&Deep одночасно враховує і загальні закономірності і складні взаємозв'язки в даних. У порівнянні з традиційними глибокими неймережами і автокодерами, ця модель більш ефективна при роботі з розрідженими даними і потребує менше обчислювальних ресурсів, що спрощує її розробку та впровадження.

Результати дослідження [23] показали доцільність використання моделі глибинного навчання Wide&Deep для систем рекомендації харчування, але суттєве покращення ефективності таких систем досі залишається відкритою проблемою. До того ж підходи глибокого навчання потребують великих обсягів маркованих

навчальних даних і значних обчислювальних ресурсів, що створює проблеми з масштабованістю систем планування [24].

Дослідження, що представлено в роботі [25] підкреслює важливість урахування різних атрибутів користувачів для поліпшення рекомендацій рецептів при побудові систем планування харчування, що базуються на експертних знаннях. Проте системи на основі знань можуть ненавмисно посилити існуючі нездорові уподобання користувача, що призведе до ефекту «ехо-камери», і негативно вплине на досягнення довгострокових цілей харчування.

Система рекомендацій щодо повторення або новизни страв, яка описана в [26] надмірно спрощує наміри користувачів, і це може призвести до ігнорування важливих нюансів. Використання таких методів ML, як навчання з підкріпленням для інтерактивного формування плану харчування супроводжується проблемами з високими вимогами до даних для навчання, інтерпретацією, холодним запуском і обчислювальними витратами.

Результати аналізу сучасних систем рекомендацій і планування харчування людини і рекомендації рецептів надано в табл. 1.1, яка показує, що незважаючи на інноваційність, існуючі системи ще не в змозі забезпечити високо персоналізоване і адаптивне планування збалансованого харчування.

Основною об'єктивною причиною такої обмеженості є покладання традиційних підходів до планування режиму харчування на статичні дані, які не враховують динамічність життя, вподобань і стану здоров'я користувачів. Також ці системи поки не в змозі на прийнятному рівні врахувати складні харчові обмеження та потреби користувачів.

Однією з основних суб'єктивних причин складності створення і впровадження в звичне життя людей універсальної системи рекомендацій є невдоволення системою при відхиленні персональних вимог від нормального розподілу. Таке відхилення за тим чи іншим критерієм вибору меню пояснюються навчанням моделей на загальних наборах даних. Проте вподобання кожної людини формуються під впливом широкого спектра факторів особистих цілей (табл. 1.2.).

Таблиця 1.1

Проблеми і недоліки існуючих систем рекомендацій і планування харчування

Система	Проблеми та недоліки	Потребують покращення в напрямку
На основі LLM	Діалоговий формат спілкування; потребує великого залучення користувача.	Використання неявних сигналів від користувача для формування плану харчування.
На основі експертних знань	Обмежена кількість рецептів, визначених експертами; відсутність зворотного зв'язку.	Використання моделей ML в системах рекомендацій рецептів, що будуть використовуватись для побудови плану харчування.
На основі контетної та колаборативної фільтрації	Обмежена адаптивність до вподобань користувача; погана масштабованість; відсутність урахування цілей щодо здоров'я користувача.	Використання моделей ML, що краще адаптуються до вподобань користувача при великих наборах даних; формування плану харчування з урахуванням цілей щодо здоров'я користувача.
На основі глибинного навчання (Wide&Deep)	Обмежена пояснюваність і слабе урахування складних дієтичних обмежень (алергенів, нутрієнтів)	Підвищення пояснюваності та включення знань про поживну цінність і алергени
На основі ML	Високі вимоги до даних, інтрерпретація результатів, високі обчислювальні витрати	Оптимізація моделей для зменшення вимог до обсягів даних і обчислювальних ресурсів

Сучасні системи планування харчування переважно орієнтовані на універсальні рішення і часто не враховують індивідуальні особливості користувачів. Це призводить до недостатньої ефективності таких систем і знижує мотивацію користувачів до їх використання. Через це узагальнення усі системи недостатньо добре персоналізуються, щоб враховувати унікальні вимоги, що пов'язані з вподобаннями користувача [27].

В роботі [28] пропонується система на основі колаборативної фільтрації з

викорстанням генетичних алгоритмів для навчання системи рекомендації, проте в ній не враховуються обмеження щодо поживних речовин користувача, а також не розкрито оцінку якості рекомендацій.

В основу інтелектуальної системи представленої в [29] покладені правила вибору страв для користувача з урахуванням його ситуативного контексту, поєднуючи фізіологічні сенсори, параметри середовища та профільні дані для формування моментальних, персоналізованих рекомендацій. Проте правила сформовані тільки для користувачів з Кореї і не можуть бути поширені на інших користувачів.

Таблиця 1.2

Здатність існуючих систем рекомендації рецептів ураховувати фактори впливу на вибір користувача

№	Система	Харчова цінність	Смакові вподобання	Час приготування	Візуальна складова	Демографічні особливості	Культурні тренди	Обладнання для приготування їжі	Навички приготування їжі	Колаборативна складова	Побажання щодо зміни раціону і/чи режиму харчування
1	FGCN [26]	+	+	-	-	+	-	-	-	-	-
2	Naik-FRS [28]	-	+	-	-	+	-	-	-	-	-
3	u-BabSang [29]	+	+	-	-	+	-	-	-	-	-
4	PiNet [27]	+	+	-	+	-	-	-	-	-	-
5	SHARE [16]	+	+	-	-	-	-	-	-	+	+
6	ChronicDietAI [20]	+	-	-	-	+	-	-	-	-	+
7	ElderNutriPlanner [21]	+	-	-	-	+	-	-	-	-	-

Наразі великої популярності набули методи колаборативної фільтрації, фільтрації на основі вмісту та гібридні моделі, що поєднують ці методи. Гібридні моделі показали багатообіцяючі результати в балансуванні між відповідністю пропозицій рецептів вподобанням користувача і різноманітністю рекомендацій [30].

В дослідженні [31] підкреслено необхідність формування персоналізованих рекомендацій, оскільки користувачі в першу чергу керуються власними вподобаннями. При цьому в роботі [31] показано, що багато рецептів, які отримали високі оцінки на онлайн-платформах, містять нездорові інгредієнти, такі як велика кількість цукру і жирів, та визначено необхідність розробки систем, які можуть рекомендувати більш здорові альтернативи, адаптуючи плани харчування до потреб користувачів.

За даними Всесвітньої організації охорони здоров'я (ВООЗ), близько 39% дорослих у світі мають надмірну вагу, а 13% страждають на ожиріння, що підкреслює нагальність вирішення проблем, пов'язаних із харчуванням [32, 33]. Саме тому розробка доступних і ефективних інтелектуальних систем, які дозволяють формувати плани збалансованого, якісного, безпечного та повноцінного харчування з урахуванням норм, індивідуальних особливостей, стану здоров'я та харчових уподобань лишається актуальним завданням сучасних прикладних інформаційних систем і технологій.

1.3 План розробки інтелектуальної системи планування збалансованого харчування людини

Для розробки інтелектуальної системи планування збалансованого харчування необхідно вирішити низку завдань, для цього запропоновано наступний план розробки:

Етап 1. Формалізація моделі збалансованого харчування людини з урахуванням обмежень користувача. На цьому етапі необхідно розробити математичну модель, що враховує індивідуальні потреби користувачів, включаючи вікові, фізіологічні, медичні особливості, а також харчові вподобання і обмеження. Формалізація здійснюється шляхом визначення набору критеріїв та параметрів збалансованості харчування.

Етап 2. Розробка концептуальної моделі інтелектуальної системи планування збалансованого харчування людини. Концептуальна модель включає визначення

основних компонентів та їх взаємозв'язків: інтерфейсу користувача, модуля введення даних, механізму рекомендацій, бази даних рецептів і нутрієнтів, підсистеми планування.

Етап 3. Розробка методу рекомендації рецептів для використання інтелектуальною системою. На цьому етапі передбачено створення методу рекомендації рецептів з використанням моделей та методів машинного навчання, з метою підбору оптимальних варіантів рецептів страв, що відповідають встановленим критеріям збалансованості та користувацьким вподобанням.

Етап 4. Обґрунтування вибору технологічного стеку для розробки системи. На основі аналізу сучасних технологій, які забезпечують масштабованість, ефективність та гнучкість, необхідно обґрунтований вибір стеку технологій, включаючи мови програмування, бази даних, фреймворки.

Етап 5. Розробка архітектури системи. Архітектура системи включає детальну розробку структурних компонентів, визначення їхніх функцій, схеми взаємодії, а також механізми інтеграції компонентів, зокрема серверної та клієнтської частин системи, модулів обробки та аналізу даних, хмарних сервісів для розгортання та підтримки роботи системи.

Етап 6. Обґрунтування та експериментальне підтвердження вибору моделей машинного навчання для використання методом рекомендації рецептів. На основі попереднього аналізу сучасних методів машинного навчання обрати оптимальні моделі, що забезпечують найкращі результати у контексті точності рекомендацій та швидкості роботи системи. Вибір моделей необхідно експериментально підтвердити за допомогою відповідних метрик якості.

Етап 7. Впровадження та підтримка системи. Етап впровадження передбачає створення інструкції користувача щодо взаємодії з системою, описується процес реєстрації, введення даних, отримання рекомендацій, та інтерпретації результатів. Також на цьому етапі планується організація підтримки користувачів, збір зворотного зв'язку та регулярне оновлення системи на основі отриманих даних і новітніх наукових рекомендацій у сфері харчування.

Висновки до розділу 1

1. Аналіз існуючих систем рекомендацій і планування харчування людини показав, що незважаючи на інноваційність, існуючі системи ще не в змозі забезпечити високо персоналізоване і адаптивне планування збалансованого харчування у зв'язку з:

- використанням статичних даних, які не враховують динамічність життя, вподобань і стану здоров'я користувачів.
- обмеженим використанням особистих даних користувача внаслідок його небажання розкривати детальну інформацію про свій режим харчування;
- індивідуальними відмінностями в харчових обмеженнях, потреби і стилях спілкування користувачів.

2. На основі проведеного аналізу визначено, що існуючі системи рекомендацій і планування харчування потребують покращення в напрямку:

- використання моделей машинного навчання, що краще адаптуються до вподобань користувача при великих наборах даних;
- формування плану харчування з урахуванням цілей користувача, технік приготування їжі та обмежень на нездорові інгредієнти.

3. Визначено план розробки інтелектуальної системи планування збалансованого харчування людини на основі сучасних методів машинного навчання.

РОЗДІЛ 2. МОДЕЛІ ТА МЕТОДИ ПОБУДОВИ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПЛАНУВАННЯ ХАРЧУВАННЯ

У розділі показано узагальнену модель інтелектуальної системи персоналізованого планування збалансованого раціону харчування, при розробці якої основна увага приділялась послідовному формуванню і адаптації індивідуального плану харчування користувача; детально описано і проаналізовано процес планування збалансованого харчування; формалізовано модель збалансованого харчування людини, яка враховує індивідуальні дієтичні та харчові обмеження користувачів; модель побудована як правило-базована логіка і описує допустиму множину рецептів, що формують меню на день; запропоновано критерії визначення відповідності рецепту дієтичним обмеженням користувача; запропоновано і детально описано кожен етап методу рекомендації рецептів, що базується на гібридному підході, інтегрує персональні вподобання і дієтичні обмеження користувачів та враховує особливості приготування страв різних категорій прийому їжі; досліджено існуючі моделі штучного інтелекту та методи машинного навчання, що можуть бути використані в системі для рекомендації рецептів.

2.1 Моделювання інтелектуальної системи персоналізованого планування збалансованого раціону харчування

Узагальнену модель інтелектуальної системи персоналізованого планування збалансованого раціону харчування, що розробляється, показано на рис. 2.1 [10].

При моделюванні цієї системи передбачалось, що для формування конкретних пропозицій меню система використовує кілька автономних, окремо навчених моделей рекомендацій, кожна з яких спеціалізується на одній з категорій прийому їжі. Такий підхід мотивується необхідністю оптимізації рекомендацій за такими різними факторами, як смаковими характеристиками, техніки приготування і прагненням збільшити точність передбачень щодо вподобань користувача.

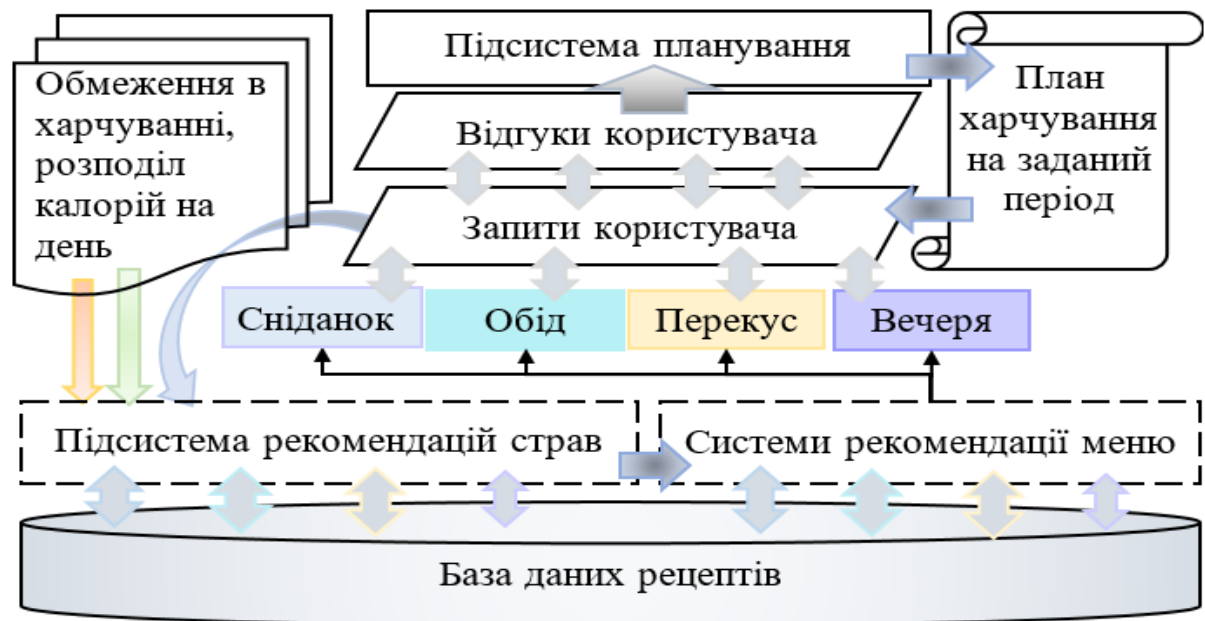


Рисунок 2.1. Узагальнена модель інтелектуальної системи персоналізованого планування збалансованого раціону харчування [10].

При цьому однією з вимог до систем рекомендацій є здатність виявляти загальні закономірності і складні взаємозв'язки в даних, оскільки це надає можливість урахувати приховані фактори, що лежать в основі вподобань користувачів і таким чином [10]:

- запобігти невдоволенню системою при відхиленні персональних вподобань користувача від нормального розподілу вподобань інших користувачів;
- адаптувати рекомендації до динаміки вподобань користувача мінімізуючи його спілкування з системою.

На рис. 2.2. показано схему процесу планування збалансованого харчування.

Процес формування плану харчування в цій системі включає декілька взаємопов'язаних етапів, що забезпечують комплексний та персоналізований підхід до створення оптимального раціону користувача.

Перший етап – збір профілю користувача. На цьому етапі система отримує детальну інформацію щодо фізіологічних характеристик, таких як вік, стать, зріст, вага, рівень фізичної активності, а також інформацію про обмеження в харчуванні, наявність алергій, хронічних захворювань, особливостей метаболізму, харчових вподобань та обмежень.



Рисунок 2.2. Схема процесу формування плану збалансованого харчування

На основі цих даних підсистема рекомендацій страв визначає приблизний розподіл калорій на сніданок, обід, перекус і вечерю.

В цій роботі за замовчуванням пропонується такий розподіл [10]:

- сніданок: 25 % добової норми калорій;
- обід: 35 % добової норми калорій;
- перекус: 10 % від добової норми калорій;
- вечеря: 30 % добової норми калорій.

Згідно з цим розподілом система рекомендацій меню формує добірку страв різних категорій прийому їжі.

Другий етап – формування правил збалансованого харчування. В цьому процесі застосовуються стандартизовані рекомендації щодо поживних речовин, вітамінів та мікроелементів, які адаптуються з урахуванням індивідуальних особливостей користувача. Цей етап передбачає побудову індивідуальної моделі харчування, яка відповідає медичним, дієтологічним та особистим вимогам.

Третій етап – конфігурація прийомів їжі на визначений період. Система визначає оптимальну кількість та розподіл прийомів їжі протягом дня, враховуючи добову норму калорій та баланс нутрієнтів. Конфігурація також передбачає врахування режиму дня користувача, включаючи робочий графік та фізичні навантаження.

Далі підсистема планування, використовуючи отримані від користувача відгуки, формує план харчування на заданий період. На цьому етапі система аналізує відгуки користувача про раніше запропоновані страви та їх рецепти. Врахування цих даних дає змогу зменшити рівень невизначеності й підвищити персоналізацію рекомендацій, оскільки план харчування коригується зі змінами уподобань і можливих додаткових обмежень користувача.

Результатом роботи системи є інтегрований план харчування з рецептами страв, що враховують калорійність і співвідношенням білків, жирів та вуглеводів. Після отримання оцінки і відгуків користувача система адаптує майбутні пропозиції, з урахуванням динаміки його вподобань чи виявлення додаткових обмежень. Таким чином, уся архітектура системи спрямована на гнучке формування персоналізованого раціону користувача.

При розробці цієї системи основна увага приділялась послідовному формуванню і адаптації індивідуального плану харчування користувача на основі:

- аналізу його особистих даних;
- специфічних обмежень і цілей щодо маси тіла;
- розділення рекомендаційного процесу на декілька спеціалізованих компонент для кожного виду прийому їжі.

Тому система, що розробляється, починає свою роботу з отримання від користувача початкових даних щодо:

- цілі (схуднення, підтримка або набір ваги);
- харчових обмежень (вегетаріанські, безглютеніві, безлактозні, т. і.);
- добової норми калорій.

Також при розробці цієї системи передбачалось використання окремо навчені системи рекомендації меню для кожного прийому їжі: сніданку, обіду, перекусу та вечері.

Четвертий етап – рекомендаційний модуль. На цьому етапі інтелектуальна система використовує алгоритми ML для генерації персоналізованих рекомендацій щодо конкретних страв і продуктів, які найкраще відповідають індивідуальним потребам користувача та його попереднім реакціям на запропоновані плани харчування.

П'ятий етап – формування конкретного плану харчування. В результаті роботи рекомендаційного модуля створюється зрозумілий, зручний і деталізований план харчування, що включає розклад прийомів їжі, рецепти страв, список необхідних продуктів і рекомендації щодо їх приготування.

Наступний, шостий етап – збір зворотного зв'язку користувача. Це критичний етап, на якому користувач надає інформацію щодо відповідності, задоволення і ефективності реалізованого плану харчування.

Сьомий етап – адаптація плану харчування на основі отриманого зворотного зв'язку. Враховуючи коментарі користувача, система вдосконалює план харчування, вносячи необхідні корективи для кращого досягнення поставлених цілей.

Фінальний етап – представлення результатів. Система формує детальні звіти, графіки та іншу наочну інформацію, яка допомагає користувачеві чітко оцінити власні досягнення, відстежувати прогрес і налаштовуватись на подальше використання системи для досягнення сталих результатів у підтримці збалансованого харчування.

Таким чином, процес формування плану харчування є циклічним, забезпечує постійне удосконалення якості рекомендацій і враховує всі індивідуальні особливості та потреби користувачів.

2.2 Формальна модель збалансованого харчування

Згідно з рекомендаціями ВООЗ [34]: збалансоване (здорове) харчування – це раціон, який забезпечує організм усіма необхідними поживними речовинами у правильних пропорціях та кількостях, відповідно до енергетичних потреб людини.

Збалансоване харчування передбачає раціон, який забезпечує організм усіма необхідними поживними речовинами у правильних пропорціях та кількостях

відповідно до енергетичних потреб людини [34]. Цей раціон є ключовим чинником підтримки здоров'я та профілактики хвороб і має відповідати індивідуальним особливостям – віку, статі, рівню фізичної активності, культурним звичаям тощо.

Планування раціону передбачає не лише складання списку прийомів їжі чи підрахунок калорій. Це багатовимірний процес, який одночасно враховує фізіологічні потреби, індивідуальні уподобання, культурний контекст і нові медичні дослідження [35]. Таким чином, збалансоване харчування є ключовим чинником підтримки здоров'я та профілактики хвороб.

Одним з головних правил формування плану збалансованого харчування є енергетичний баланс: споживання калорій має відповідати витратам енергії, щоб підтримувати нормальну масу тіла, оскільки надлишок енергії в раціоні веде до набору ваги та підвищує ризик ожиріння, а нестача – до виснаження; обидва стани шкідливі для здоров'я [36].

Для індивідуального розрахунку добової калорійної потреби використовуються формули, що оцінюють базальний метаболізм (BMR) за антропометричними даними, з подальшим коригуванням на активність [37]:

$$BMR = \begin{cases} 10 \times W + 6.25 \times H - 5 \times Y + 5 & (\text{для чоловіків}); \\ 10 \times W + 6.25 \times H - 5 \times Y - 161 & (\text{для жінок}), \end{cases} \quad (2.1)$$

де: BMR – основний обмін речовин; W – вага користувача в кг; H – зріст користувача в см; Y – вік користувача.

Отримане значення BMR відображає приблизну кількість калорій, що витрачає організм у стані спокою.

Для визначення загальної добової витрати калорій (TDEE) використовують формули (2.2), (2.3) [38]:

$$TDEE = BMR \cdot \lambda; \quad (2.2)$$

$$T_{kcal} = \begin{cases} TDEE - \delta \cdot TDEE & (\text{для схуднення}); \\ TDEE & (\text{для підтримки ваги}); \\ TDEE + \delta \cdot TDEE & (\text{для набору ваги}), \end{cases} \quad (2.3)$$

де: $TDEE$ – денна норма калорій для підтримки ваги; T_{kcal} – цільова норма калорій; BMR – основний обмін речовин; W – вага користувача в кг; H – зріст користувача в см; Y – вік користувача; δ – коефіцієнт швидкості зміни ваги (за замовчуванням 0.15); λ – коефіцієнт активності користувача (табл. 2.1).

Таблиця 2.1

Значення коефіцієнтів активності

Рівень активності	Опис	λ
Низька активність	Сидяча робота, майже повна відсутність вправ	1.2
Легка активність	Легкі вправи 1-3 рази на тиждень	1.375
Помірна активність	Помірні вправи 4-5 разів на тиждень	1.55
Висока активність	Високонавантажені тренування 6-7 разів на тиждень або фізична робота	1.725
Надвисока активність	Професійний спортсмен/важка фізична робота	1.9

ВООЗ також встановила орієнтовні популяційні нормативи споживання макронутрієнтів у відсотках від загальної калорійності раціону (рис. 2.3) [35].



Рисунок 2.3. Баланс макронутрієнтів згідно рекомендацій ВООЗ [39]

Згідно з [39], у збалансованому харчуванні частка енергії повинна надходити приблизно на 45–65% з вуглеводів, 10–35% з білків і 20–35% з жирів.

Такий розподіл забезпечує покриття потреб організму як в енергії, так і в необхідних поживних речовинах

Ці межі дозволяють гнучкість залежно від індивідуальних потреб. Проте вихід за ці межі може негативно впливати на обмін речовин і здоров'я загалом. Також варто враховувати, що 1 грам різних макронутрієнтів дає неоднакову кількість енергії. Приблизно 4 ккал міститься в 1 г білка чи вуглеводів, тоді як 1 г жиру забезпечує ~9 ккал. Отже, щоб дотримуватися рекомендованого співвідношення, необхідно контролювати не лише масу споживаних білків, жирів і вуглеводів, але й їх вклад у загальну калорійність.

Норма споживання [39]:

$$P_g = W \cdot \alpha, \quad (2.4)$$

$$F_g = W \cdot \beta, \quad (2.5)$$

$$C_g = \frac{T_{kcal} - P_g \cdot 4 - F_g \cdot 9}{4}, \quad (2.6)$$

де P_g , F_g , C_g – норма макронутрієнтів в грамах для білків, жирів та вуглеводів відповідно; W – вага користувача в кілограмах; $\alpha \in [1.6; 2.2]$ – коефіцієнт вмісту білків (за замовчуванням 1.8); $\beta \in [0.6; 1]$ – коефіцієнт вмісту жирів, (за замовчуванням 0.8); T_{kcal} – денна норма калорій для підтримки ваги.

У [38] підкреслюється, що це орієнтовні значення, які можуть коливатися залежно від індивідуального метаболізму, стану здоров'я і мети людини. Якщо метою є зміна ваги, калорійність раціону коригують: для схуднення створюють помірний дефіцит калорій, а для набору ваги – надлишок, порівняно з розрахованою нормою.

Окрім того, за оцінками [40], близько 8% дітей і 10% дорослих у світі мають харчові алергії та непереносимість певних продуктів. Хронічні захворювання також часто вимагають спеціальної дієти чи корекції звичного харчування для підтримки прийняттого стану здоров'я і запобігання прогресу хвороби шляхом дієтотерапії. Такі

дієти включають обмеження щодо споживання певних інгредієнтів страв, а також певних технік приготування страв, як от обмеження щодо смаженої їжі.

Для формалізації моделі збалансованого харчування в цій роботі запропоновано введення дозволеної кількості певного продукту в плані харчування, а також приготування страв певною технікою. Для заборонених продуктів дозволена межа складає 0, що дозволяє повністю виключити їх з плану харчування.

Модель збалансованого харчування побудована як правило-базована логіка й описує допустиму множину рецептів S , що формують меню на день. Для кожного з нутрієнтів (білки, жири, вуглеводи) k введено нижню L_k і верхню R_k межі споживання (2.4) – (2.6).

Таким чином, для того щоб запропонований набір рецептів $r \in S$ підтримував баланс нутрієнтів, повинна виконуватись нерівність:

$$L_k \leq \sum_{r \in S} w_r v_r^k \leq R_k, k = 1 \dots 3 \quad (2.7)$$

де $r \in S$ – рецепти запропоновані на день; $k = 1 \dots 3$ – нутрієнти; L_k та R_k – нижня та верхня межі споживання нутрієнту k ; w_r – кількість порцій рецепту r в меню; v_r^k – кількість нутрієнту k в одній порції рецепту r (в грамах).

Для врахування дієтичних обмежень користувача, алергій, та обмежень щодо здоров'я, для кожного інгредієнту i та техніки приготування страви t вводиться додаткове обмеження максимальної кількості інгредієнту i в меню харчування на день (в грамах), а також загальної маси їжі приготованої за допомогою техніки t в меню харчування на день (в грамах). Для заборонених інгредієнтів i та технік t встановлюються нульові обмеження.

Для того, що запропонований набір рецептів $r \in S$ задовольняв обмеження користувача повинні виконуватись наступні нерівності:

$$\sum_{r \in S} q_{ir} \leq \Lambda_i \quad (\forall i \in I) \quad (2.8)$$

$$\sum_{r \in S} w_{tr} \leq \Theta_t \quad (\forall t \in T) \quad (2.9)$$

де: $r \in S$ – рецепти запропоновані на день; q_{ir} – загальна кількість інгредієнта i у вибраному наборі рецепті r в грамах; I – множина інгредієнтів; Λ_i – добова доза інгредієнту (0, якщо він заборонений); w_{tr} – загальна маса їжі приготованої за допомогою техніки t у вибраному наборі рецептів r ; T – множина технік приготування страв; θ_t – добова норма їжі, що приготована з використанням техніки t (0, якщо вона заборонена).

Профіль обмежень користувача можна представити у вигляді вектора:

$$u = (L \mid R \mid \Lambda \mid \theta) \quad (2.10)$$

$$\Lambda = (i^{(\Lambda)}, d^{(\Lambda)}) \quad (2.11)$$

$$i^{(\Lambda)} = (i_1, i_2 \dots i_p) \quad (2.12)$$

$$d^{(\Lambda)} = (\lambda_{i_1}, \lambda_{i_2} \dots \lambda_{i_p}) \quad (2.13)$$

де: $L = (L_1, L_2, L_3)$ – нижні межі споживання нутрієнтів; $R = (R_1, R_2, R_3)$ – верхні межі споживання нутрієнтів; $\theta = (\theta_1, \theta_2 \dots \theta_m)$ – допустимі добові маси страв, приготованих кожною технікою; m – кількість технік приготування в системі; Λ – допустимі межі інгредієнтів; p – кількість обмежень щодо інгредієнтів; $i^{(\Lambda)}$ – упорядкований вектор індексів інгредієнтів, для яких задано обмеження; $d^{(\Lambda)}$ – упорядкований вектор числових значень меж споживання інгредієнтів, що відповідають індексам з $i^{(\Lambda)}$.

Профіль рецепту може бути представлений наступним чином:

$$v = (v \mid q \mid w); \quad (2.14)$$

$$q = (j^{(q)}, d^{(q)}); \quad (2.15)$$

$$j^{(q)} = (i_1, i_2 \dots i_s); \quad (2.16)$$

$$d^{(q)} = (\lambda_{j_1}, \lambda_{j_2} \dots \lambda_{j_s}), \quad (2.17)$$

де: $v = (v_1, v, v_3)$ – кількість кожного з нутрієнтів у одній порції рецепту в грамах; $w = (w_1, w_2 \dots w_m)$ – маса їжі, що приготована за допомогою техніки в одній

порції рецепту; m – кількість технік приготування в системі; q – допустимі межі інгредієнтів; s – кількість обмежень щодо інгредієнтів; j – упорядкований вектор індексів інгредієнтів; $d^{(q)}$ – упорядкований вектор числових значень маси інгредієнтів в одній порції рецепту, що відповідають індексам з $j^{(q)}$.

Набір технік приготування їжі, що враховані при побудові системи складається з множини: {барбекю, бланшування, варіння, вирівнювання, витримування, готування в горщику, глазурування, деактивування, дистилювання, емульгування, засолювання, заморожування, замішування, запікання, згущування, зливка, змішування, карамелізування, квашення, комбінування, коптіння, маринування, нарізання кубиками, охолодження, ошпарювання, повільне приготування, приготування пюре, піднімання наверх, приготування в мікрохвильовій печі, приготування під тиском, проціджування, подрібнювання, проціджувати, приготування на грилі, підсмажування, розтирання, розгладжування, тертя, розм'якшування, розтирання, розтоплювання, соління, смаження на пательні, смаження на решітці, тушкування, тушкування, тушкування на пару, ферментування} [10].

2.3 Персоналізація в системах планування харчування

Системи планування харчування мають скласти збалансований графік харчування (зазвичай на тиждень), підбираючи відповідні рецепти на кожен день.

Тижневий план харчування зазвичай складається з декількох прийомів їжі на день (сніданок, обід, вечеря та перекуси), обраних із бази рецептів і розподілених на 7 днів. Завдання полягає в тому, щоб ця колекція рецептів відповідала загальним цілям харчування та практичним обмеженням, забезпечуючи при цьому різноманітність і збалансованість. На практиці це означає, що система повинна враховувати загальне споживання поживних речовин (калорій, макроелементів і мікроелементів) протягом тижня і розподіляти їх відповідно на кожний прийом їжі/день.

План має відповідати добовим нормам калорійності та рекомендованому споживанню поживних речовин, уникаючи надмірно високих або низьких показників у окремі дні.

Також план має забезпечувати різноманітність раціону, уникаючи повторення страв або інгредієнтів протягом декількох днів поспіль, тим самим забезпечуючи широкий спектр поживних речовин і запобігаючи втомі від меню. Вибираючи рецепти з найвищим рейтингом для кожного прийому їжі та уникаючи повторень, система може створити збалансований за поживними речовинами та різноманітний тижневий план харчування, що враховує уподобання та вимоги до здоров'я користувача.

Хоча планування на основі правил і оптимізація харчування є надзвичайно важливими, ефективна система планування харчування має надавати пріоритет персоналізації. Кожна людина має унікальні харчові вподобання, дієтичні потреби та стан здоров'я, тому універсальний план харчування, традиційні підходи до планування харчування (статичні дієтичні плани чи загальні шаблони харчування) часто не дотримуються користувачами, бо не відповідають способу їх життя і смакам [41].

Щоб вирішити цю проблему, сучасні системи планування харчування інтегрують методи рекомендацій, подібно до того, як електронна комерція або медіа-рекомендатори підбирають пропозиції відповідно до інтересів користувачів [42, 43].

У контексті інтелектуальної системи планування збалансованого харчування компонент рекомендацій відповідає за перегляд бази даних рецептів та визначення страв, які не тільки відповідають обмеженням щодо харчування та дієти, але й є привабливими для користувача. Ця персоналізація важлива, оскільки план харчування може забезпечити збалансоване харчування лише в тому випадку, якщо користувач дійсно його дотримується, а користувачі частіше дотримуються планів, що містять продукти, які вважають підходящими [44].

Для реалізації персоналізованого вибору рецептів використовують різні моделі та методи рекомендацій. Методи рекомендацій щодо планування харчування об'єднуються в три групи, кожна з яких має свої переваги й недоліки та відіграє свою роль у комплексній системі планування харчування.

2.3.1 Класичні методи

До класичних методів відносять спільне та контент-орієнтоване фільтрування. Класичні системи рекомендацій розглядають рецепти як «елементи», а користувачів

системи – як «користувачів», адаптуючи їх з ранніх досліджень у сферах рекомендацій щодо фільмів та продуктів [45, 46].

Спільна фільтрація (CF) – на основі користувачів чи на основі елементів k -найближчих сусідів, чи на основі латентних факторів, таких як розклад на сингулярні значення – виводить переваги з моделей історичних відгуків: користувачі, яким у минулому подобалися подібні страви, як передбачається, будуть насолоджуватися подібними стравами в майбутньому [44, 45].

Фільтрування на основі вмісту (CB) представляє кожен рецепт за допомогою явних атрибутів – списку інгредієнтів, тегів кухні, векторів поживних речовин, часу приготування – і рекомендує страви, характеристики яких відповідають вказаним або виведеним уподобанням користувача.

Класичні методи рекомендацій є історичною основою персоналізованих систем вибору продуктів харчування і залишаються важливими орієнтирами перед впровадженням більш сучасних, складних методів. У контексті планування харчування ці методи використовують явні оцінки (наприклад, «п'ять зірок») або неявні сигнали (наприклад, перегляди рецептів або журнали кулінарів), щоб виявити приховані смакові уподобання, спільні для різних користувачів і страв.

У цьому розділі з акцентом на тому, як кожен метод моделює взаємодію користувача та рецепта, управляє розрідженістю відгуків про харчування та балансує обчислювальну ефективність та інтерпретованість розглядаються:

- розклад на сингулярні значення (SVD) [49];
- KNN [50];
- ненегативна матрична факторизація (NMF) [51].

Розуміння механізмів і компромісів цих методів закладає основу для порівняння класичного спільного фільтрування з сучасними гібридними підходами та підходами глибокого навчання, а також пояснює, чому ці методи досі є основою багатьох систем рекомендацій.

KNN є одним з найбільш популярних алгоритмів машинного навчання, що широко застосовується у задачах рекомендацій. В контексті рекомендації рецептів для збалансованого харчування KNN може успішно використовувати історичні дані

взаємодій користувачів з рецептами для формування точних та персоналізованих рекомендацій.

Робота алгоритму KNN для цієї задачі відбувається наступним чином:

1. Формується профіль користувача на основі його попередніх взаємодій (наприклад, оцінок, переглядів чи обраних рецептів). Профіль користувача представляється у вигляді вектора [50]:

$$u = (u_1, u_2, \dots, u_m), \quad (2.18)$$

де u_j – оцінка, яку користувач поставив рецепту j , m – кількість рецептів у системі.

2. Також формуються профілі інших користувачів [50]:

$$v_i = (v_{i1}, v_{i2}, \dots, v_{im}), \quad (2.19)$$

де v_{ij} – оцінка рецепту j користувачем i .

3. Для рекомендації рецептів поточному користувачу спочатку обчислюється схожість між його профілем та профілями інших користувачів. Часто використовується косинусна схожість [52]:

$$\text{sim}(u, v_i) = \frac{u \cdot v_i}{|u| |v_i|} = \frac{\sum_{j=1}^m u_j v_{ij}}{\sqrt{\sum_{j=1}^m u_j^2} \sqrt{\sum_{j=1}^m v_{ij}^2}}. \quad (2.20)$$

4. Вибираються користувачів, профілі яких мають найбільшу схожість з профілем активного користувача.

5. На основі рецептів, які ці найближчих користувачів позитивно оцінили, формуються рекомендації для активного користувача. Для цього обчислюється прогнозована оцінка рецепта [50]:

$$\hat{u}_j = \frac{\sum_{i \in N_K(u)} \text{sim}(u, v_i) \cdot v_{ij}}{\sum_{i \in N_K(u)} |\text{sim}(u, v_i)|}. \quad (2.21)$$

де – множина найближчих до активного користувача сусідів.

6. Користувачу рекомендуються рецепти з найбільшими прогнозованими оцінками.

Матрична факторизація моделює матрицю оцінок користувачів і елементів R як добуток двох матриць низького рангу P (користувачі $\times k$) і Q (елементи $\times k$), де k латентних факторів вивчаються таким чином, щоб скалярний добуток вектора користувача u P_u і вектора елемента i Q_i наближався до спостережуваної оцінки R_{ui} . Навчання (часто за допомогою стохастичного градієнтного спуску або чергування найменших квадратів) коригує ці вектори, щоб мінімізувати похибку прогнозування за відомими рейтингами, а регуляризація запобігає їх надмірному пристосуванню.

Приховані фактори фіксують приховані виміри, такі як жанрова спорідненість або цінова чутливість, які спільно пояснюють переваги користувачів і характеристики елементів, що дозволяє системі узагальнювати: навіть якщо користувач ніколи не оцінював певний елемент, скалярний добуток його навченого вектора з вектором елемента дає правдоподібний рейтинг рекомендації.

Сингулярна декомпозиція (SVD) є класичним методом для вирішення задачі матричної факторизації, що широко застосовується для побудови колаборативних рекомендаційних систем. Фундаментальний принцип SVD полягає в декомпозиції матриці взаємодії користувач-об'єкт на три матриці: одна представляє користувачів, інша – об'єкти, а діагональна матриця містить сингулярні значення, які вказують на силу зв'язків між користувачами та об'єктами [53].

$$A = U\Sigma V^T, \quad (2.22)$$

де A – матриця взаємодії користувач=об'єкт, U – ортогональна матриця, що репрезентує властивості користувачів, V – ортогональна матриця, що репрезентує властивості об'єктів, Σ – діагональна матриця, що вказує силу зв'язків.

Така декомпозиція дозволяє виявити приховані фактори, які відображають основні вподобання користувачів і характеристики товарів, що полегшує створення персоналізованих рекомендацій.

Ефективність SVD пов'язана з розрідженістю даних матриці взаємодій A , оскільки підхід дозволяє суттєво зменшувати розмірність матриць, при цьому зберігаючи основні характеристики та залежності між ними.

Метод невід'ємної матричної факторизації (NMF) за задумом подібний до SVD, але з додатковим обмеженням невід'ємності всіх значень в отриманих матрицях.

Формально NMF розкладає початкову не-негативну матрицю вподобань R на добуток [54]:

$$R \approx W H, \quad (2.23)$$

де W – матриця розмірності $m \times k$, H – матриця $k \times n$, і $W_{ik}, H_{kj} \geq 0$ для всіх елементів.

Ця вимога не-негативності означає, що латентні фактори інтерпретуються як адитивна комбінація базисних компонент, без від'ємних ваг. На відміну від SVD, який дозволяє позитивні та негативні значення (що можуть компенсувати одне одного), NMF будує фактори, які можна тлумачити як тематика або групи характеристик.

При аналізі рецептів NMF може виявити латентний фактор, який відповідає вегетаріанським стравам, інший – десертам, третій – білковим низьковуглеводним стравам тощо, і кожен рецепт буде мати певний не-негативний «вклад» у кожен з цих факторів.

Такий підхід часто підвищує інтерпретованість: фактори NMF більш прямо відповідають реальним ознакам, адже відсутня проблема від'ємних значень (не потрібно віднімати один фактор із іншого, щоб зрозуміти характеристику).

Для рекомендації рецептів NMF застосовується до тієї ж матриці користувач-рецепт R , що й SVD, з тією різницею, що всі рейтинги трактуються як не-негативні. Після факторизації отримуємо матрицю W , де кожен користувач і описаний вектором

інтенсивності прихильності до k латентних факторів та матрицю H , де кожен рецепт j описаний вектором належності до цих k факторів.

Добуток $W \times H$ – наближення матриці оцінок і передбачити, наскільки користувачу сподобається рецепт.

Таким чином, прогнозування рейтингу здійснюється аналогічно до SVD: через скалярний добуток відповідних векторів користувача і рецепта.

Різниця в тому, що ці значення можна інтерпретувати: якщо для користувача високі компоненти W_{ik} відповідають, скажімо, факторам «низькокалорійні страви» і «багато овочів», то система буде рекомендувати рецепти, в яких високі відповідні компоненти H_{kj} – тобто страви, що сильно належать до цих категорій. На основі цього рекомендації будуються більш прозоро: користувачу пропонуються страви з тих категорій, до яких він має найбільшу схильність (згідно з його історією оцінок).

2.3.2 Методи на основі дерев прийняття рішень

Методи на основі дерев розглядають вибір рецептів як задачу ранжування під наглядом, в якій сукупність дерев рішень прогнозує релевантність кожного кандидата. Популярні реалізації включають Gradient-Boosted Decision Trees (GBDT) [55], XGBoost [56], LightGBM [57], оскільки обробляють такі неоднорідні характеристики, як харчова цінність, теги дієти, демографічні дані без важкої попередньої обробки, вони добре підходять для змішаних даних, типових для систем харчування. Оцінки важливості ознак також дають дієтологам зрозуміле уявлення про те, чому пропонується той чи інший рецепт.

На відміну від лінійних підходів, деревоподібні методи природно відображають складні, багаторівневі взаємодії між індивідуальними особливостями людини – віком, стилем харчування, алергіями, смаковими уподобаннями розкладом дня – та характеристиками страв, такими як тип кухні, наявність певних інгредієнтів чи час приготування. Для кожного користувача формується однозначний шлях від кореня до листка дерева; цей шлях легко зчитати як правило персоналізації. Завдяки такій прозорій логіці система не лише генерує релевантні рекомендації, а й аргументує,

чому саме ці страви відповідають потребам конкретної людини, підвищуючи довіру та залученість користувача.

Методи дерев рішень представляють задачу рекомендації рецептів як кероване відображення $f_{\theta}: \mathcal{X} \rightarrow \mathcal{Y}$, де f_{θ} – параметризована рекомендаційна функція, θ – вектор її параметрів, $\mathcal{X}$ – простір ознак, $\mathcal{Y}$ – простір цільових значень.

Простір ознак $\mathcal{X} \subset R^d$ об'єднує загальні показники поживної цінності, ознаки алергії та дієтичні обмеження, індивідуальні переваги користувача та часовий контекст, тоді як ціль $\mathcal{Y}$ є дискретною міткою «прийняти/відхилити» або дійсним числовим значенням оцінки користувачем рецепту.

Дерево рекурсивно розбиває $\mathcal{X}$ на осередки-гіперпрямокутники, обираючи у кожній вершині розклад (A^*, t^*) , де A^* – індекс (назва) ознаки, що дає найкраще розщеплення, t^* – числовий поріг (або категоріальне значення) для цієї ознаки, за яким виконується розщеплення, таким чином щоб мінімізувати похибку за критерієм.

При використанні ентропії критерій має вигляд [58]:

$$H(S) = - \sum_{c=1}^k p_c \log_2 p_c, \quad (2.24)$$

де $H(S)$ – ентропія вибірки S ; k – кількість класів; p_c – пропорція класу c у S [57]:

$$p_c = \frac{1}{|S|} \sum_i 1 \{y_i = c\}, \quad (2.25)$$

де $|S|$ – розмір вибірки; i – індекс спостереження; y_i – клас спостереження i ; $1 \{ \cdot \}$ – індикаторна функція, що дорівнює 1, коли умова істинна.

Приріст інформації задається наступним чином [58]:

$$IG(S, A) = H(S) - \sum_v \frac{|S_v|}{|S|} H(S_v), \quad (2.26)$$

де $IG(S, A)$ – приріст інформації для ознаки A ; S_v – підвибірка, утворена значенням v (або половиною розщеплення); $|S_v|$ – її розмір.

Альтернативою для класифікації є мінімізація індексу Джині [59]:

$$\Delta_{\text{Gini}}(S, A) = (1 - \sum_c p_c^2) - \sum_v \frac{|S_v|}{|S|} (1 - \sum_c p_{c,v}^2), \quad (2.27)$$

де $\Delta_{\text{Gini}}(S, A)$ – зменшення індексу Джині при розщепленні; $p_{c,v}$ – частка класу c у підвибірці S_v .

Для задачі регресії – мінімізація середньоквадратичної помилки [59]:

$$\Delta_{\text{MSE}}(S, A) = \frac{1}{|S|} \sum_i (y_i - \overline{y_S})^2 - \sum_v \frac{|S_v|}{|S|} \cdot \frac{1}{|S_v|} \sum_{i \in S_v} (y_i - \overline{y_{S_v}})^2, \quad (2.28)$$

де $\Delta_{\text{MSE}}(S, A)$ – значення середньоквадратичної похибки; y_i – реальне значення; $\overline{y_S}$ – середнє значення по всій вибірці; $\overline{y_{S_v}}$ – середнє в S_v .

Навчання припиняється, коли приріст менший за ϵ або кількість спостережень у вузлі падає нижче $n_{\min}$. Для нового вектору ознак x модель видає розподіл класів чи середнє цільове значення листа. Базова модель прозора й швидка, та її порогові правила легко перенавчаються на рідкісних даних відгуків і погано відображають плавні компроміси між нутрієнтами, тому зазвичай застосовують ансамблеві варіації.

Метод випадкових лісів зменшує дисперсію шляхом бутстрап-агрегування В дерев. Кожне дерево $T^{(b)}$ навчається на бутстрап-вибірці $S^{(b)}$ і при кожному розщепленні розглядає лише $m \ll d$ випадково вибраних ознак.

Для класифікації ансамбль оптимізує критерій [60]:

$$\widehat{P}_{\text{RF}}(y = c | \mathbf{x}) = \frac{1}{B} \sum_{b=1}^B \widehat{P}^{(b)}(y = c | \mathbf{x}), \quad (2.29)$$

де $\widehat{P}_{\text{RF}}(y = c | \mathbf{x})$ – ймовірність класу c у випадковому лісі; B – кількість дерев; b – індекс дерева; $P^{(b)}$ – прогноз b -го дерева; $\mathbf{x}$ – вектор ознак.

Для задачі регресії використовується критерій [60]:

$$\widehat{y}_{\text{RF}}(x) = \frac{1}{B} \sum_{b=1}^B \widehat{y}^{(b)}(x), \quad (2.30)$$

де $\widehat{y}_{\text{RF}}(x)$ – усереднений регресійний прогноз лісу; B – кількість дерев; $\widehat{y}^{(b)}(x)$ – прогноз b -го дерева.

Якщо дисперсія одного дерева дорівнює σ^2 , а парна кореляція між деревами – ρ , то дисперсія лісу наближено дорівнює $\rho\sigma^2 + (1 - \rho)\frac{\sigma^2}{B}$. Бутстрап зменшує другу доданку, а випадковий піддобір ознак – ρ .

Прогнози (out-of-bag) для $\approx 37\%$ вибірки, що не потрапляє у бутстрапи, дають неупереджену оцінку помилки, а перемішування конкретної ознаки в цих підвибірках вимірює її важливість.

Дерева рішень з градієнтним прискоренням (GBDT) – це модель систем рекомендацій, що використовує ансамблевий метод, в якому послідовно навчаються декілька дерев рішень і кожен з наступних ансамблів дерев має виправити помилки попереднього.

Кожне дерево – це послідовність розбиття за ознаками, які розбивають дані таким чином, щоб найкраще передбачити залишки поточної ітерації. Метою навчання є передбачення чіткої оцінки рецепту користувачем.

Функція втрат визначається як [61]:

$$Loss = \frac{1}{2} \sum_{i=1}^N (y_i - F(x_i))^2, \quad (2.31)$$

де N – кількість елементів в навчальній вибірці (x_i, y_i) ; x_i – характеристики користувача; y_i – реальна оцінка i -го рецепту користувачем, що включає техніки приготування страви; $F(x_i)$ – поточний прогноз GBDT.

Прогноз GBDT на першій ітерації $F_0(x)$ визначається як середнє значення реальних оцінок i -го рецепту [61]:

$$F_0(x) = \frac{1}{2} \sum_{i=1}^N y_i. \quad (2.32)$$

Прогноз GBDT на поточній $(m-1)$ -ій ітерації будуються таким чином [61]:

$$F_m(x) = F_{m-1}(x) + \gamma h_m(x), \quad (2.33)$$

де F_m – прогноз GBDT на поточної ітерації; γ – параметр усадки, що приймає значення від 0 до 1; $h_m(x)$ – m -те дерево регресії, навчене спеціально для апроксимації залишків поточної ітерації.

Підібране дерево додається до загальної моделі, наближаючи прогнози до істинних оцінок.

Залишки поточної $(m-1)$ -ої ітерації обчислюються за формулою [61]:

$$r_i^{(m)} = y_i - F_{m-1}(x_i), \quad (2.34)$$

де $r_i^{(m)}$ – залишки поточної ітерації; F_{m-1} – прогнози GBDT на поточній ітерації; y_i – реальна оцінка користувача.

Після M ітерацій, фінальна модель GBDT для прогнозування оцінок має вигляд [61]:

$$F_M(x) = F_0(x) + \sum_{m=1}^M \gamma h_m(x). \quad (2.35)$$

Проте деревоподібні ансамблі покладаються на вручну створені ознаки і менш придатні для моделювання довгострокових взаємодій або часової динаміки.

2.3.3 Методи на основі нейронних мереж

Методи на основі нейронних використовує нейронні архітектури для навчання високовимірних представлень користувачів і рецептів, таких як: нейронна матрична факторизація (NeuMF) [62], моделі Wide&Deep [63], рекомендаційні системи на

основі трансформерів або графічних нейронних мереж, а також агенти з підкріплювальним навчанням, які розглядають складання меню як послідовну задачу прийняття рішень.

Глибокі моделі поєднують запам'ятовування відомих пар користувач-рецепт з потужною генералізацією: вони можуть обробляти необроблений текст, зображення або часові ряди поживних речовин і одночасно оптимізувати смак, поживність, різноманітність і навіть вартість продуктів. Їхні недоліки, такі як великі вимоги до даних, низька прозорість і більші обчислювальні вимоги, є типовими для глибокого навчання. Тому їх зазвичай поєднують із простішими фільтрами, які гарантують дотримання основних обмежень щодо поживної цінності або безпеки.

На практиці може бути застосована комплексна або гібридна стратегія, використання фільтрації на основі вмісту для відбору невідповідних рецептів, а потім модель спільного фільтрування для ранжування решти кандидатів за прогнозованими уподобаннями користувачів. Точний вибір моделі залежить від наявності даних (достатньої кількості оцінок користувачів для навчання моделі CF або багатих метаданих для роботи моделі на основі вмісту) і конкретних цілей системи (точність прогнозування, різноманітність рекомендацій, новизна тощо).

Нейронне колаборативне фільтрування (NCF) – це підхід глибокого навчання, який моделює взаємодію користувачів і елементів, виходячи за межі лінійних припущень традиційних методів спільного фільтрування. NCF має на меті виявити нелінійні високорівневі закономірності між користувачами та елементами шляхом поєднання вбудованих шарів з однією або кількома нейронними мережами [64]. NeuMF (нейронна матрична факторизація) може розглядатися як конкретний приклад NCF, в якому дві гілки – узагальнена матрична факторизація (GMF) та багат шаровий перцептрон (MLP) – об'єднуються для фіксації як лінійних, так і складних нелінійних взаємозв'язків в єдиній структурі [65, 66].

У сфері рекомендацій рецептів та збалансованого планування харчування ці моделі чудово справляються з виявленням тонких особливостей користувачів та рецептів, що в кінцевому підсумку дає персоналізовані пропозиції, що відповідають потребам харчування.

Колаборативне фільтрування (CF): Користувачі з подібною історією взаємодії, як правило, мають схожі вподобання щодо товарів. У випадку рекомендацій рецептів, якщо два користувачі оцінюють або клікають на рецепти з подібними поживними властивостями, CF робить висновок, що вони можуть мати схожі вподобання щодо їжі.

Глибокі нейронні мережі: завдяки багат шаровим нейронним архітектурам NCF вивчає нелінійні взаємодії вищого порядку, які можуть бути пропущені лінійними методами. У контексті харчування це може відображати, як різні користувачі оцінюють смак, поживні речовини та дієтичні обмеження у складний, індивідуальний спосіб.

NeuMF реалізує ці ідеї, поєднуючи дві парадигми в NCF: гілку GMF, яка фіксує лінійну взаємодію користувача і рецепта за допомогою елементного продукту, та гілку MLP, яка вивчає більш багаті нелінійні перетворення [65]. Це поєднання дозволяє системі відображати як прямі сигнали про вподобання (наприклад, відома схильність користувача до вегетаріанських рецептів), так і приховані структури вподобань.

Архітектура NeuMF складається з двох основних компонентів: гілки GMF та гілки MLP.

Нехай: U та R – множини всіх користувачів та рецептів відповідно; $m = |U|$ і $n = |R|$ позначають загальну кількість користувачів і рецептів; $P \in \mathbb{R}^{(m \times k)}$ та $Q \in \mathbb{R}^{(n \times k)}$ – матриці латентних факторів для гілки GMF, де k – розмірність латентного простору; $W^{(l)}, b^{(l)}$ – параметри ваги та зміщення для l -го шару гілки MLP.

Гілка GMF моделює взаємодію користувача та рецепта за допомогою простого елементного добутку. Для користувача u та рецепта r вектори вбудовування GMF мають такий вигляд: $P_u \in \mathbb{R}^k$, $Q_r \in \mathbb{R}^k$, де P_u – u -й рядок P , а Q_r – r -й рядок Q .

Вихідні дані GMF є елементним добутком цих вбудовувань:

$$GMF_{u,r} = P_u \odot Q_r, \quad (2.36)$$

де $\odot$ позначає добуток Адамара [67].

Гілка MLP виходить за межі лінійних взаємодій, використовуючи кілька щільних шарів. Зазвичай латентні вектори користувача та рецепта об'єднуються:

$$x_{u,r} = [P'_u; Q'_r], \quad (2.37)$$

де $P'_u \in \mathbb{R}^k$ і $Q'_r \in \mathbb{R}^k$ представляють різні (або спільні, але відмінні) вкладення, що використовуються спеціально для гілки *MLP*, а $[\cdot]$ позначає з'єднання векторів. З'єднаний вектор $x_{u,r}$ потім подається в багатошарову нейронну мережу:

$$z^{(1)} = f(W^{(1)}x_{u,r} + b^{(1)}); \quad (2.38)$$

$$z^{(2)} = f(W^{(2)}z^{(1)} + b^{(2)}); \quad (2.39)$$

$$z^{(L)} = f(W^{(L)}z^{(L-1)} + b^{(L)}), \quad (2.40)$$

де L – кількість прихованих шарів, а $f(\cdot)$ – нелінійна функція активації (наприклад, ReLU). Останній шар $z^{(L)}$ представляє нелінійне високорівневе уявлення, що фіксує складні взаємодії між користувачем і рецептом.

На відміну від використання GMF або MLP окремо, NeuMF об'єднує результати обох гілок.

Нехай:

$$h_{GMF} = GMF_{u,r}; \quad (2.41)$$

$$h_{MLP} = z^{(L)}. \quad (2.42)$$

Остаточне «об'єднання» часто є з'єднанням цих двох векторів:

$$h_{NeuMF} = [h_{GMF}; h_{MLP}]. \quad (2.43)$$

Ця комбінована репрезентація, h_{NeuMF} , проходить через додатковий шар або просте лінійне перетворення для виведення прогнозованого рейтингу або ймовірності взаємодії:

$$\hat{y}_{u,r} = \sigma(W^F h_{NeuMF} + b^F), \quad (2.44)$$

де W^F і b^F – параметри шару злиття, а $\sigma(\cdot)$ часто позначає сигмоїдну функцію, якщо завдання прогнозування є неявним зворотним зв'язком (наприклад, прогнозування, чи зацікавить користувача рецепт), або лінійну функцію, якщо прогножуються явні рейтинги.

NeuMF зазвичай навчається шляхом мінімізації функції втрат. Для явного сценарію оцінки можна застосувати середньоквадратичну похибку (MSE) або середню абсолютну похибку (MAE):

$$L = \sum(u, r) (y_{u,r} - \hat{y}_{u,r})^2, \quad (2.45)$$

де $y_{u,r}$ – реальна оцінка, присвоєна користувачем u рецепту r , а $\hat{y}_{u,r}$ – прогноз моделі.

Для неявних налаштувань зворотного зв'язку (коли користувачі натискають або не натискають на рецепт) часто більш доречним є бінарна перехресна ентропія або парний рейтинг втрат:

$$L = -\sum(u, r) [y_{u,r} \log(\hat{y}_{u,r}) + (1 - y_{u,r}) \log(1 - \hat{y}_{u,r})]. \quad (2.46)$$

Для оновлення параметрів використовуються загальні методи оптимізації, такі як стохастичний градієнтний спуск (SGD), Adam або RMSProp.

Модель двох веж – нейронна архітектура, що призначена для вирішення задач рекомендацій, особливо при роботі з вбудованими даними користувачів і елементів (рецептів) у великомасштабних системах. Замість того, щоб з початку об'єднувати характеристики користувачів та елементів в одній мережі, у конфігурації з двома вежами навчаються дві окремі мережі кодування – одна для користувачів, а інша для рецептів. В подальшому ці кодування вирівнюються або порівнюються для отримання рейтингу релевантності або ранжування [68].

Цей підхід особливо корисний при обчисленні схожості або збігів у великих масштабах, оскільки він дозволяє ефективно здійснювати пошук за допомогою пошуку найближчих сусідів у спільному просторі вбудовування.

Вежа користувача: кодує характеристики користувача (наприклад, історію поведінки, харчові вподобання, обмеження щодо здоров'я) у вигляді щільного векторного представлення.

Вежа рецептів: кодує характеристики рецептів (інгредієнти, кухня, інформація про поживну цінність) у вигляді щільного векторного представлення.

- Функція схожості: Після того, як обидві вежі генерують свої відповідні вбудовування, застосовується міра схожості (часто скалярний добуток або косинусна схожість) для отримання остаточного балу, що визначає відповідність рецепту уподобанням і потребам користувача.

Завдяки окремому навчанню веж модель може вивчити потужні представлення користувачів і елементів, а крок виведення може бути прискорений шляхом індексації вбудовувань елементів для пошуку в режимі реального часу.

У вежі користувачів кожен користувач ($u \in U$) відображається на вектор вбудовування ($\mathbf{u}_u \in \mathbb{R}^k$) за допомогою нейронної мережі (або MLP).

Ці характеристики можуть бути введені в шар вбудовування (або шари, складені один на один) для отримання ($\mathbf{u}_u$). За наявності одного шару вбудовування, він може бути параметризований за допомогою ($\mathbf{P} \in \mathbb{R}^{|U| \times k}$), як було представлено раніше, але в більш просунутих конструкціях характеристики користувачів проходять через багат шарову мережу. Математично, для користувача u : $\mathbf{u}_u = f_{user}(u)$, де $f_{user}(\cdot)$ є нейронною мережею (наприклад, багат шаровим перцептроном, MLP), а $\mathbf{u}_u \in \mathbb{R}^k$ є результатом вбудовування користувача.

Вежа рецептів отримує вхідні дані, специфічні для рецептів, такі як список інгредієнтів, спосіб приготування, профіль макронутрієнтів або тип кухні. Ці характеристики перетворюються в інший k -вимірний вектор вбудовування ($\mathbf{v}_r \in \mathbb{R}^k$).

$$\mathbf{v}_r = f_{recipe}(r), \quad (2.47)$$

де $f_{recipe}(\cdot)$ конкретний рецепт нейронна мережа (часто має таку саму або схожу структуру, як і вежа користувачів), а $\mathbf{v}_r \in R^k$ – отримане вбудовування рецепта.

Оскільки дві вежі не переплітаються до моменту кодування, їхні результати $\mathbf{u}_u$ та $\mathbf{v}_r$ можна об'єднати за допомогою простої функції оцінювання, такої як скалярний добуток або косинусна подібність $\hat{y}_{u,r} = \mathbf{u}_u^T \mathbf{v}_r$.

Для інтелектуальної системи планування харчування цей скалярний добуток може приблизно оцінити, наскільки харчові потреби та смакові вподобання користувача відповідають поживній цінності та кулінарним характеристикам рецепта.

Хоча модель з двома вежами гарантує, що вбудовування користувачів і рецептів знаходяться в одному просторі вбудовування, пряме порівняння вбудовування одного користувача з усіма рецептами у величезному каталозі було б надто дорогим. Натомість система використовує ANN [69], що сприяють ефективному пошуку найкращих рецептів з мінімальною затримкою.

Перед наданням онлайн-рекомендацій вежа рецептів використовується для генерації та зберігання вбудованих даних для всіх рецептів.

Ці попередньо обчислені вбудовані дані утворюють велике сховище або індекс, які можуть зберігатися в індексній структурі ANN, що оптимізована для швидкого пошуку найближчих сусідів.

Під час запиту (коли користувач u входить у систему або запитує рекомендації щодо страв), вбудовування користувача $\mathbf{u}_u$ обчислюється в режимі реального часу, відображаючи останню інформацію або уподобання користувача (такі, як нові обмеження в харчуванні, останні взаємодії з рецептами).

Маючи $\mathbf{u}_u$, система запитує індекс ANN, шукаючи найближчі вбудовування $\mathbf{v}_r$

Wide&Deer є популярним при створенні рекомендаційних систем. У контексті рекомендацій страв використання Wide&Deer допомагає поєднати простіші, більш інтерпретовані закономірності – як перевірку безпосередніх зв'язків між потребами користувача та характеристиками рецепта – зі здатністю виявляти складніші вподобання, що стосуються смаку, поєднання інгредієнтів і поживних цілей [70].

Широкий компонент є, узагальненою лінійною моделлю, яка фокусується на запам'ятовуванні. Запам'ятовування особливо цінне в рекомендаціях рецептів для

безпосереднього використання співпадінь ознак (коли інгредієнти, що часто поєднуються, або певні атрибути користувачів корелюють з конкретними стравами).

Нехай $x_{wide} \in R^n$ представляють вхідні дані для широкої моделі, де (n) – кількість широких ознак (включаючи додаткові перехресні ознаки). Прогнозований вихід $\hat{y}_{wide}$ з широкого компонента можна представити як [68]:

$$\hat{y}_{wide} = w^T x_{wide} + b, \quad (2.48)$$

де $w \in R^n$ є вектор ваги, а b є членом зміщення.

У контексті рекомендацій рецептів x_{wide} може включати:

- Індикаторні ознаки, що представляють класи інгредієнтів («молочні продукти», «овочі», «зернові»).
- Перехресні ознаки демографічних даних користувачів («вегетаріанець» × «економний») з типом кухні («італійська», «китайська»).
- Інші категоріальні або одноразово закодовані ознаки, що відображають відомі асоціації в харчових звичках.

Якщо широкий компонент спрямовано на запам'ятовування, то глибокий компонент призначено для узагальнення шляхом вивчення високого рівня абстракцій із більш складних взаємодій. Ця глибока компонента зазвичай складається з декількох шарів, які можуть фіксувати нелінійні взаємозв'язки між вхідними та вихідними даними. Позначимо $x_{deep} \in R^m$ як вхідний вектор для глибокого модуля, який може включати щільні вбудовування користувачів, рецептів або інгредієнтів.

Загальна І-шарова нейронна мережа прямого поширення перетворює x_{deep} таким чином [71]:

$$h^{(1)} = f(W^{(1)}x_{de} + b^{(1)}); \quad (2.49)$$

$$h^{(2)} = f(W^{(2)}h^{(1)} + b^{(2)}); \quad (2.50)$$

$$h^{(l)} = f(W^{(l)}h^{(l-1)} + b^{(l)}), \quad (2.51)$$

де $W^{(k)}$ і $b^{(k)}$ – матриця ваг і зміщення для шару (k) відповідно; $f(\cdot)$ – нелінійна функція активації (ReLU або сигмоїда).

Вихід останнього прихованого шару $f(h^{(l)})$ агрегується в глибокий вихід $\hat{y}_{deep}$.

Для рекомендацій рецептів вектори вбудовування можуть кодувати [71]:

- вбудовування поживних речовин: фіксація макроелементів (білків, жирів, вуглеводів) та мікроелементів (вітамінів, мінералів).
- вбудовування інгредієнтів: представлення складних взаємозв'язків між елементами, такими як «часник добре поєднується з цибулею» або «лосось зазвичай супроводжує лимон».
- вбудовування уподобань користувача: відображення минулих моделей споживання, культурного фону та обмежень у харчуванні користувача.

Остаточний прогноз $\hat{y}$ моделі W&D зазвичай формується шляхом об'єднання результатів широких і глибоких компонентів. У багатьох реалізаціях перед обчисленням остаточного результату або ймовірності два прогнози об'єднуються за допомогою шлюзу або простої операції додавання.

Для завдань бінарної класифікації, таких як «Чи сподобається користувачеві цей рецепт?», загальний прогноз може використовувати сигмоїдну функцію $\sigma(\cdot)$ [71]:

$$\hat{y} = \sigma(\hat{y}_{wide} + \hat{y}_{deep}). \quad (2.52)$$

Підставляючи вирази для $\hat{y}_{wide}$ та $\hat{y}_{deep}$:

$$\hat{y} = \sigma(w^{Tx_{wd}} + b + u^{Th^{(l)}}), \quad (2.53)$$

де u – вектор ваг кінцевого вихідного шару, застосований до $h^{(l)}$ результату останнього прихованого шару глибокої мережі.

Для завдань рекомендації поширеним вибором є функція втрат перехресної ентропії в сценаріях класифікації або середньоквадратична похибка в налаштуваннях регресії.

Функція втрат перехресної ентропії ($\mathcal{L}$) з цільовим значенням $y \in (0,1)$ і прогнозом $\hat{y} \in [0,1]$ визначається за формулою:

$$\mathcal{L} = -[y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})]. \quad (2.54)$$

Параметри моделі $w, W^{(k)}, b^{(k)}, u$ визначаються шляхом мінімізації $\mathcal{L}$ у навчальному наборі даних за допомогою ітераційних методів оптимізації, таких як стохастичний градієнтний спуск або Adam.

2.4 Метод рекомендацій рецептів

Схему роботи трьох етапного методу рекомендації рецептів, що використовує методи ML для відбору рецептів і розробляється для формування рецептів страв різних категорій проїму їжі, показано на рис. 2.4.

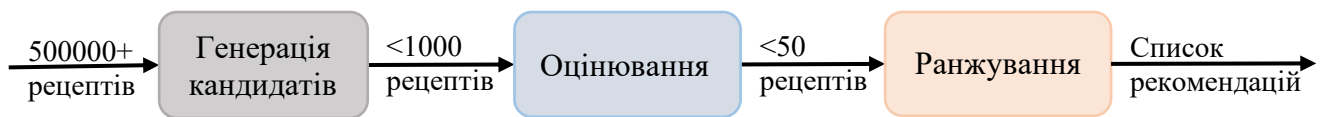


Рисунок 2.4. Схема роботи трьох етапного методу рекомендації рецептів [37]

Метод складається з етапів: генерація кандидатів, оцінювання і ранжування.

Схему генерації рецептів-кандидатів запропонованого методу показано на рис. 2.5.

На цьому етапі відбувається вибір підмножини рецептів з множини усіх наявних і фільтрація рецептів, що містять інгредієнти, які не підходять за обмеженнями користувача. Якісний генератор кандидатів може знайти 500 рецептів з показником відгуку $>90\%$. Це означає, що він успішно відображає найбільш релевантні позиції ціною включення деяких менш релевантних рецептів. Крім того, враховується охоплення різних показників інтересів користувача і різноманітність каталогу.

На етапі генерації кандидатів запропоновано застосовувати модель штучного інтелекту двох веж, де кожна вежа представлена нейронною мережею, що дозволяє

здійснити вкладання користувачів і рецептів в спільний векторний простір. При чому одна нейронна мережа кодує характеристики користувача, а інша – характеристики страв.

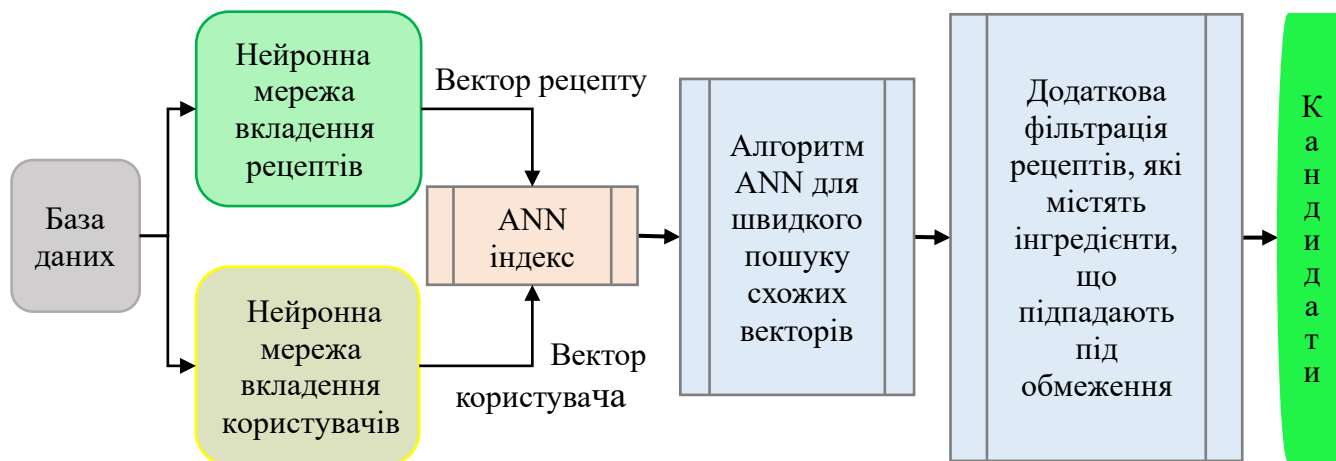


Рисунок 2.5 Схема роботи етапу генерації рецептів-кандидатів [72]

Генератор кандидатів зазвичай оцінюється за показником $Recall@N$. Забезпечення охоплення передбачає вибір кандидатів із різних категорій рецептів чи використання декількох генераторів.

Після обчислення вбудованих даних для швидкого отримання сету потенційних страв-кандидатів і формування плану збалансованого харчування із мільйонів варіантів використовується ANN.

При надходженні запиту на рекомендацію, вбудовування користувача генерується вежею користувача, а алгоритм ANN знаходить страви-кандидати, вбудовування яких найкраще відповідає вбудовуванню користувача.

ANN-індекс за мілісекунди повертає найкращі рецепти-кандидати, навіть якщо каталог містить мільйони позицій, завдяки розумному скороченню пошуку найближчих векторів medium.com.

Двохвежева архітектура сумісна з таким підходом до індексації, оскільки розділяє обчислення користувача та елемента так, що представлення елемента обчислюються один раз і використовуються повторно. Таке розділення суттєво прискорює роботу системи, оскільки перетворює проблему $O(U \times M)$ (порівняння всіх користувачів U з усіма рецептами M) на офлайн-обчислення $O(M)$ рецептів і

онлайн-пошук $O(U \times \log M)$, що робить досяжними рекомендації в реальному часі навіть для дуже великих M .

Додаткова фільтрація відсіває рецепти, які містять інгредієнти або техніки приготування, що підпадають під обмеження користувача.

Псевдокод для фільтрації рецептів згідно з (2.8), (2.9):

```
FUNCTION FILTER_CANDIDATE_RECIPES(  
    user_id,  
    meal_type,  
    raw_recipes  
) → filtered_recipes  
  
    lim ← DB.GET_USER_LIMITS(user_id)  
    kcal_daily ← lim.calories  
    ingredient_caps ← limits.ingredient_g  
    technique_caps ← limits.technique_g  
  
    meals_cfg ← DB.GET_MEALS_PER_DAY(user_id)  
    filtered_recipes ← []  
    FOR r IN raw_recipes DO  
        IF r.calories > kcal_daily THEN  
            CONTINUE  
        ENDIF  
        for_all ingr in r.ingredients :  
            IF ingr.g > ingredient_caps.get(ingr) THEN  
                CONTINUE  
            ENDIF  
        for_all tech in r.techniques :  
            IF tech.g > technique_caps.get(tech) THEN  
                CONTINUE  
            ENDIF  
  
        filtered_recipes.APPEND(r)  
    ENDFOR  
  
    RETURN filtered_recipes
```

Після відбору кандидатів на етапі оцінювання (рис. 2.6) використовуються методи машинного і глибокого навчання для коригування оцінок кожного елемента, що дає змогу підвищити якість рекомендацій, оскільки ступінь релевантності страв для кожного користувача визначається точніше.

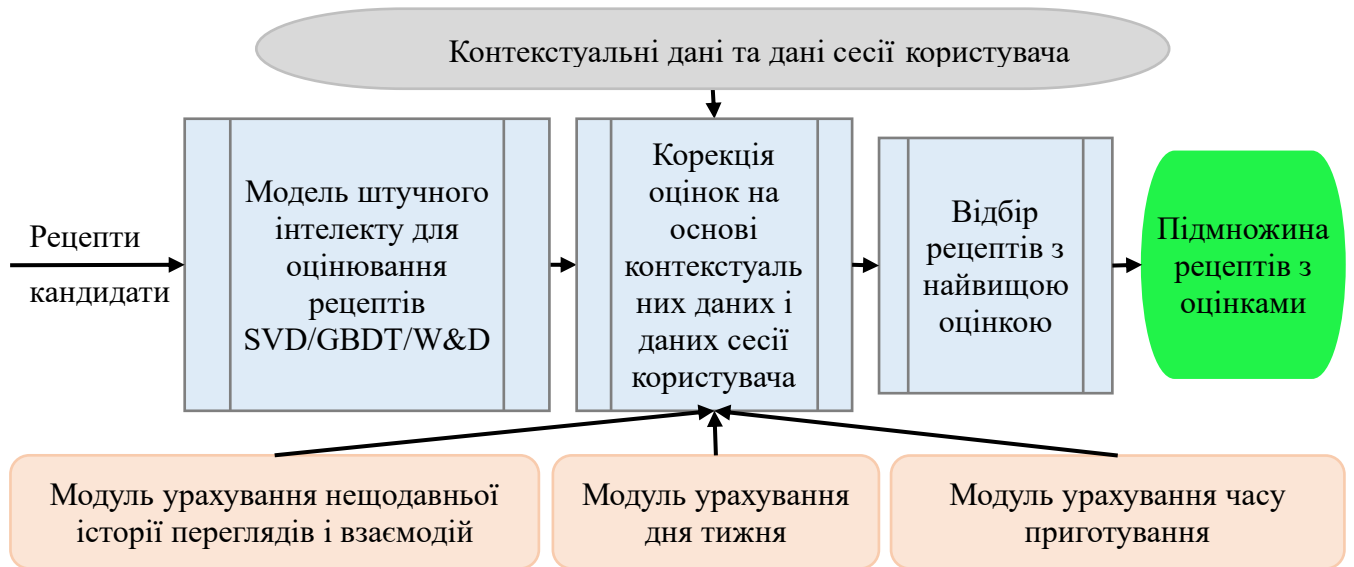


Рисунок 2.6. Схема роботи етапу оцінювання рецептів [72]

Ключовим нововведенням цієї роботи є впровадження модульної системи плагінів, що коригує оцінки моделі штучного інтелекту (SVD/GBDT/W&D), використовуючи контекстуальні дані та дані сесії користувача (2.55):

$$s'_{\{base\}(u,i)} = s_{\{base\}(u,i)} + \Delta p(u,i), \quad (2.55)$$

де: $s_{\{base\}(u,i)}$ – оцінка основної моделі рецепту i для користувача u , p – плагін, що створює зміщення $\Delta p(u,i)$.

В основу роботи запропонованих модулів урахування нещодавньої історії переглядів і взаємодій, дня тижня і часу приготування покладено (2.56) – (2.60):

$$\Delta p(u,i) = \alpha \cdot F_{clicks}(u,i) + \beta \cdot F_{views}(u,i) - \gamma \cdot F_{bounces}(u,i); \quad (2.56)$$

$$(\Delta t) = e^{-\lambda \Delta t}; \quad (2.57)$$

$$F_{click}(u, i) = \sum_{c \in C_{u,i}} w(t - t_c); \quad (2.58)$$

$$F_{views}(u, i) = \sum_{v \in V_{u,i}} w(t - t_v); \quad (2.59)$$

$$F_{bounces}(u, i) = \sum_{b \in B_{u,i}} w(t - t_b), \quad (2.60)$$

де: α, β, γ – параметри налаштування модулів, що вказують на вагу відповідного фактору F_{clicks} - т ; F_{views} - т ; $F_{bounces}$ - т ; F_{dow} - т ; $C_{u,i}$ — множина кліків на рецепт i користувачем u в сесії; $V_{u,i}$ — множина переглядів рецепту i користувачем u в сесії; $B_{u,i}$ — множина подій негативної взаємодії користувача u з рецептом i ; t_c, t_v, t_b — час відповідної події; λ – коефіцієнт забування.

Псевдокод для модулю урахування нещодавньої історії переглядів та взаємодій:

```

FUNCTION RESCORE_WITH_RECENT_HISTORY(
    user_id,
    candidate_recipes,
    horizon_days      = 30,
    half_life_days    = 7,
     $\alpha = 1.1$ ,  $\beta = 1.1$ ,  $\gamma = 1.2$ 
)  $\rightarrow$  rescored_recipes
 $\lambda \leftarrow \ln(2) / \text{half\_life\_days}$ 

recent  $\leftarrow$  SELECT recipe_id, action_type, timestamp, dwell_sec
                FROM activity_log
                WHERE user_id = user_id
                AND timestamp  $\geq$  NOW() - horizon_days

Fclicks  $\leftarrow$  DEFAULT_DICT(0)
Fviews  $\leftarrow$  DEFAULT_DICT(0)
Fbounces  $\leftarrow$  DEFAULT_DICT(0)

FOR ev IN recent DO
     $\Delta t\_days \leftarrow$  (NOW() - ev.timestamp)
     $w \leftarrow \text{EXP}(-\lambda * \Delta t\_days)$ 

    IF ev.action_type = "CLICK" THEN

```

```

        Fclicks[ev.recipe_id] ← Fclicks[ev.recipe_id] + w

    ELIF ev.action_type = "VIEW" THEN
        Fviews[ev.recipe_id] ← Fviews[ev.recipe_id] + w

    ELIF (ev.action_type = "VIEW" AND ev.dwell_sec < 5) OR
        (ev.action_type = "SKIP") THEN
        Fbounces[ev.recipe_id] ← Fbounces[ev.recipe_id] + w
    END FOR

    rescored_recipes ← []
    FOR c IN candidate_recipes DO
        fc ← Fclicks.get(c.recipe_id, 0)
        fv ← Fviews.get(c.recipe_id, 0)
        fb ← Fbounces.get(c.recipe_id, 0)

        Δp ← α * fc + β * fv - γ * fb
        new_score ← c.base_score + Δp

        rescored_recipes.APPEND({recipe_id: c.recipe_id,
                                score: new_score})
    END FOR

    rescored_recipes.SORT_BY(score DESC)
    RETURN rescored_recipes

```

Формула (2.63) представляє зміщення $\Delta_{p(u,i)}$ модулю врахування дня тижня.

$$P_{u,i,d} = \frac{n_{u,i,d} + \mu}{n_{u,i} + 7\mu}, \quad (2.61)$$

$$P_{u,d} = \frac{n_{u,d} + \mu}{N_u + 7\mu}; \quad (2.62)$$

$$\Delta_{p(u,i)} = \log \frac{P_{u,i,d}}{P_{u,d}}, \quad (2.63)$$

де: $d \in \{1, 2 \dots 7\}$ — день тижня; $n_{u,i,d}$ — кількість взаємодій користувача u з рецептом i в день d ; $n_{u,i}$ — кількість всіх взаємодій користувача u з рецептом i за весь

час; $n_{u,d}$ — кількість всіх взаємодій користувача u з усіма рецептами в день d ; N_u — загальна кількість взаємодій користувача u ; μ — коефіцієнт згладжування;

Під взаємодією розуміється додавання користувачем u рецепту i до плану харчування в минулому.

Псевдокод модулю врахування дня тижня:

```
FUNCTION RESCORE_DAY_OF_WEEK(  
    user_id,  
    candidate_recipes,  
    day_of_week,  
     $\mu = 1.0$   
) → rescored  
    N_u ← DB.GET_TOTAL_EVENTS(user_id)  
    n_u_d ← DB.GET_DAY_EVENTS (user_id, day_of_week)  
  
    recipe_ids ← [c.recipe_id FOR c IN candidate_recipes]  
    stats ← DB.BATCH_GET_RECIPE_STATS(user_id, recipe_ids, day_of_week)  
  
    rescored ← []  
    P_u_d ← (n_u_d +  $\mu$ ) / (N_u + 7* $\mu$ )  
  
    FOR c IN candidate_recipes DO  
        stat ← stats[c.recipe_id]  
        n_ui ← stat.n_ui IF stat ELSE 0  
        n_ui_d ← stat.n_ui_d IF stat ELSE 0  
  
        P_ui_d ← (n_ui_d +  $\mu$ ) / (n_ui + 7* $\mu$ )  
         $\Delta p$  ← LOG( P_ui_d / P_u_d )  
  
        rescored.APPEND({recipe_id: c.recipe_id,  
                        score: c.base_score +  $\Delta p$ })  
    ENDFOR  
  
    rescored.SORT_BY(score DESC)  
    RETURN rescored
```

Формула 2.64 представляє зміщення $\Delta_{p(u,i)}$ модулю часу для приготування.

$$\Delta_{p(u,i)} = \begin{cases} \lambda \cdot \left(1 - \frac{t_i}{T_u}\right), & t_i \leq T_u; \\ -\mu \cdot \left(\frac{t_i}{T_u} - 1\right), & t_i > T_u \end{cases} \quad (2.64)$$

де: t_i – час (в хвиликах), необхідний для приготування рецепту i , T_u – час (в хвиликах), що користувач вказав як наявний для приготування рецепту, λ, μ – параметри модулю.

Псевдокод для модулю врахування доступного часу для приготування:

```

FUNCTION RESCORE_COOK_TIME(
    user_id,
    meal_type,
    candidate_recipes,
     $\lambda = 0.30$ ,
     $\mu = 0.45$ 
) → rescored
    T_u ← DB.GET_COOK_TIME_LIMIT(user_id, meal_type)
    rescored ← []

    FOR c IN candidate_recipes DO
        t_i ← c.cook_min

        IF t_i ≤ T_u THEN
             $\Delta p \leftarrow \lambda * (1 - t_i / T_u)$ 
        ELSE
             $\Delta p \leftarrow -\mu * (t_i / T_u - 1)$ 
        ENDIF

        rescored.APPEND({recipe_id: c.recipe_id,
                        score:      c.base_score +  $\Delta p$ })
    ENDFOR

    rescored.SORT_BY(score DESC)
    RETURN rescored

```

На рис. 2.7 показано схему роботи етапу ранжування.

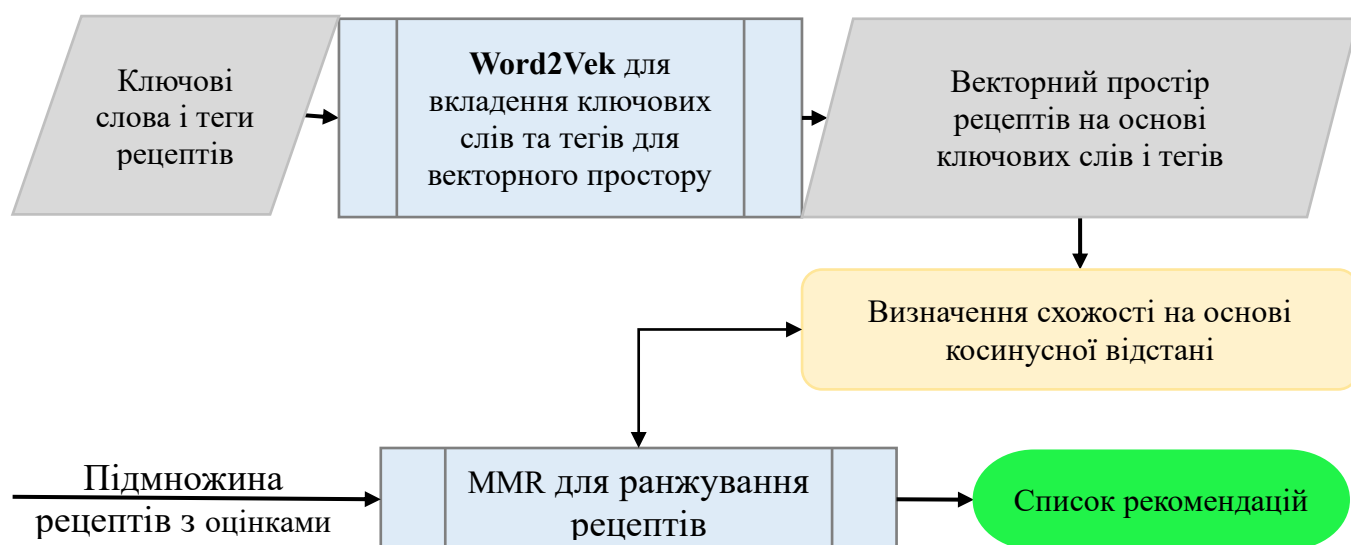


Рисунок 2.7 Схема роботи етапу ранжування [37]

На цьому етапі елементи впорядковуються відповідно до оцінок користувача, враховуючи додаткову бізнес-логіку чи обмеження (підтримка новизни, забезпечення різноманітності, дотримання рекламних пріоритетів).

В результаті формується оптимізований перелік рекомендацій, що відповідає контексту і уподобанням користувача.

Основна концепція максимальної граничної релевантності (MMR) полягає в ітеративному відборі страв не тільки на основі індивідуальної релевантності до уподобань користувача, а й на основі їх відмінності від вже рекомендованих елементів [73].

Цей подвійний критерій допомагає уникнути повторюваних рекомендацій, які з часом можуть знизити задоволеність користувачів. MMR працює шляхом обчислення комбінованого балу для кожної потенційної рекомендації.

Перша складова цього балу оцінює, відповідність уподобань користувача (на основі історії користувача і його харчових потреб або явних оцінок).

Друга складова оцінює відмінність від тих страв, що вже включені до набору рекомендацій, використовуючи показники схожості на основі поживної цінності, типу кухні, різноманітності інгредієнтів і смакових характеристик. Для етапу ранжування

для кожного рецепту ключові слова та теги вкладаються до векторного простору за допомогою алгоритму Word2Vec [74].

Векторне представлення рецептів страв за ключовими словами і тегами використовується для ранжування рецептів за допомогою алгоритму максимальної граничної релевантності MMR [75]:

$$MMR(i) = \lambda \cdot r(i) - (1 - \lambda) \cdot \max_{j \in S} Sim(i, j), \quad (2.65)$$

де $MMR(i)$ – оцінка рецепту i після ранжування; S – множина рецептів, що вже була поміщена в готовий список рекомендацій; $r(i)$ – оцінка рецепта i , що отримана з попереднього етапу, $Sim(i, j)$ – косинусна відстань між рецептом i та j , що отримана на основі ключових слів та тегів; λ – параметр, що при формуванні списку рекомендацій визначає важливість різноманітності.

Висновки до розділу 2

1. Запропоновано узагальнено модель інтелектуальної системи персоналізованого планування збалансованого раціону харчування юдини, яка послідовно поєднує етапи збирання даних, формування профілю користувача, синтезу меню та зворотного зв'язку, що забезпечує адаптацію раціону до динамічних потреб людини і дозволяє сформуванати цілісне теоретико-методичне підґрунття для подальшої реалізації програмного рішення.

2. Детально описано процес планування збалансованого харчування і сформалізовано модель збалансованого харчування у вигляді правило-базованої логіки, яка описує допустиму множину рецептів і дозволяє оперативно перевіряти відповідність меню дієтичним обмеженням та цілям користувача.

3. На основі дослідження класичних методів, методів на основі дерев рішень і нейронних мереж, обґрунтовано доцільність окремого навчання моделей для різних категорій прийому їжі (сніданок, обід, перекус, вечеря) та сформовано критерії вибору алгоритмів машинного навчання для подальших експериментальних досліджень.

4. Запропоновано і детально описано гібридний метод рекомендації рецептів, що інтегрує персональні вподобання, медичні та кулінарні обмеження для забезпечення різноманітності використано векторне представлення рецептів та алгоритм максимальної граничної релевантності.

РОЗДІЛ 3. РОЗРОБКА КОНЦЕПТУАЛЬНОЇ МОДЕЛІ ТА АРХІТЕКТУРИ СИСТЕМИ ПЛАНУВАННЯ ЗБАЛАНСОВАНОГО ХАРЧУВАННЯ ЛЮДИНИ

У третьому розділі запропоновано концептуальну модель загальнодоступної інтелектуальної системи планування збалансованого харчування людини, що використовує окремо навчені моделі машинного навчання, спеціалізовані для сніданків, обідів, перекусів і вечерь; описано архітектуру системи; розглянуто життєвий цикл моделей систем рекомендацій; обґрунтовано вибір інструментів для розробки системи і проведено аналіз необхідних для реалізації запропонованої системи компонент; проєктовано реляційну схему бази даних та інформаційні потоки, що забезпечують збереження профілів користувачів, рецептів, нутрієнтної інформації й історії взаємодій; описано модуль формування плану харчування, що перетворює вихід методу рекомендацій рецептів на цілісний меню план для заданого періоду часу, механізми зворотного зв'язку від користувача і вирішення проблеми холодного старту.

3.1 Концептуальна модель системи

На рис. 3.1 показано концептуальну модель системи персоналізованого планування збалансованого раціону харчування людини, що дозволяє використовувати окремо навчені системи рекомендації меню для кожного прийому їжі: сніданку, обіду, перекусу та вечері.

При розробці моделі висувається гіпотеза, що використання чотирьох окремо навчених моделей машинного навчання – спеціалізованих для сніданків, обідів, перекусів і вечерь – забезпечує статистично значуще покращення якості рекомендацій порівняно з єдиною універсальною моделлю і таким чином покращується ефективність роботи системи в цілому. Тому запропонована модель передбачає використання окремих, автономних моделей ML для використання в системах рекомендації рецептів.



Рисунок 3.1. Концептуальна модель системи персоналізованого планування збалансованого раціону харчування

Концептуальна модель об'єднує інформаційні потоки і функціональні підсистеми, необхідні для автоматизованого формування персоналізованих планів харчування.

Центральним вузлом моделі виступає підсистема формування плану харчування, а допоміжні модулі забезпечують збирання, зберігання та обробку даних про користувача та рецепти, а також забезпечують вибір найбільш релевантних рецептів за допомогою підсистеми рекомендації рецептів.

Модель передбачає запит користувача на формування плану збалансованого харчування на заданий ним період часу, де для кожного дня користувач надає кількість прийомів кожного типу їжі: сніданків, перекусів, обідів та вечері.

При цьому однією з вимог до систем рекомендацій є здатність виявляти загальні закономірності і складні взаємозв'язки в даних, оскільки це надає можливість урахувати приховані фактори, що лежать в основі вподобань користувачів і таким чином:

– запобігти невдоволенню системою при відхиленні персональних вподобань користувача від нормального розподілу вподобань інших користувачів;

– адаптувати рекомендації до динаміки вподобань користувача мінімізуючи його спілкування з системою.

Далі підсистема планування, використовуючи отримані від користувача відгуки, формує план харчування на заданий період. На цьому етапі система аналізує відгуки користувача про раніше запропоновані страви та їх рецепти. Врахування цих даних дає змогу зменшити рівень невизначеності й підвищити персоналізацію рекомендацій, оскільки план харчування коригується зі змінами уподобань і можливих додаткових обмежень користувача.

Підсистема формування профілю користувача та рецепту відповідає за зведення всіх вхідних даних до уніфікованого векторного простору. Вона нормалізує медичні та дієтичні обмеження та смакові вподобання користувача, а для кожного рецепту — інгредієнтний склад, нутрієнти та техніку приготування. Далі обидва типи даних перетворюються на латентні векторні репрезентації, що зберігаються у базі з можливістю швидкого оновлення при будь-яких змінах (нові алергії, редагування рецепта). Завдяки цьому метод рекомендації та ядро планування миттєво отримують числові описи «користувач ↔ страва» і можуть швидко обчислювати релевантність без повернення до сирих даних.

Підсистема збору інформації про поведінку користувача безперервно фіксує як явні дії (лайки, заміни страв, нотатки «з'їв/не з'їв»), так і неявні сигнали (час перегляду рецепта, повторні відкриття, дані з фітнес-трекерів). Події проходять очищення від шуму, агрегуються у показники «прилипання» до плану й конвертуються у поведінкові ознаки, які доповнюють користувацький вектор. Оновлені вектори негайно впливають на online-рекомендації, а історичні логи потрапляють до дата-сховища для регулярного перенавчання моделей, у такий спосіб замикаючи цикл самонавчання системи.

Результатом роботи системи є інтегрований план харчування з рецептами страв, що враховують калорійність і співвідношенням білків, жирів та вуглеводів, та не містить рецептів, що не відповідають дієтичним обмеженням користувача. Після отримання оцінки і відгуків користувача система адаптує майбутні пропозиції, з урахуванням динаміки його вподобань чи виявлення додаткових обмежень. Таким

чином, уся система спрямована на гнучке формування персоналізованого раціону користувача.

3.2 Архітектура інтелектуальної системи планування збалансованого харчування

На рис. 3.2 представлено архітектуру запропонованої інтелектуальної системи планування збалансованого харчування людини.

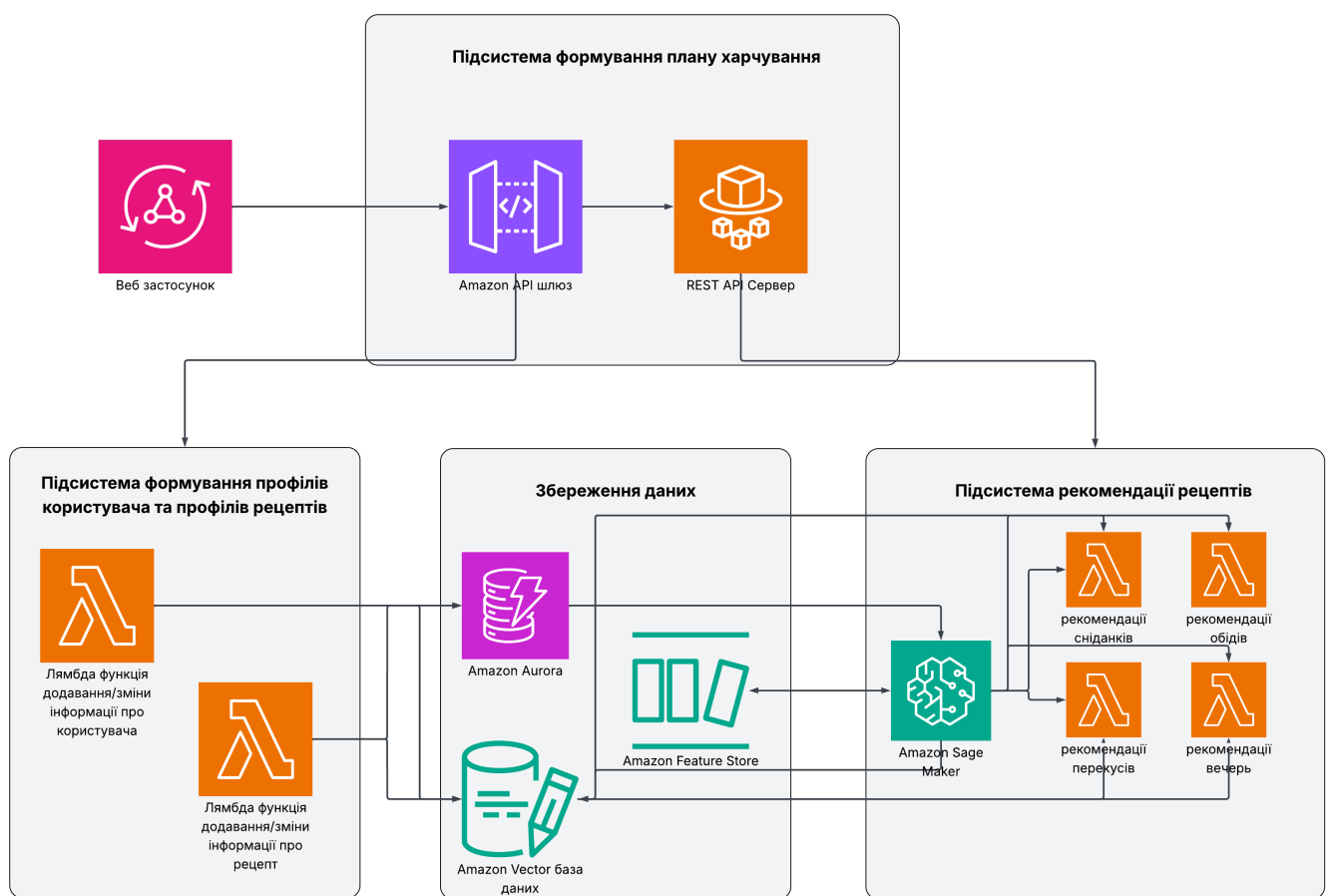


Рисунок 3.2 Архітектура інтелектуальної системи планування збалансованого харчування людини

Ця архітектура призначена для збору вимог користувачів, створення та коригування планів харчування, а також надання персоналізованих рекомендацій за допомогою керованих, масштабованих і безпечних служб AWS.

Розділення системи на підсистеми, кожна з яких має чіткі обов'язки, дає можливість гарантувати, що додаток залишатиметься гнучким, зручним в обслуговуванні та здатним розвиватися разом із дослідженнями в галузі харчування та потребами користувачів.

Виділено такі підсистеми: фронтент-шар, підсистема формування плану харчування, підсистема формування профілів користувачів та профілів рецептів, підсистема збереження та обробки даних, підсистема рекомендації рецептів.

Фронтенд-шар є підсистемою інтелектуальної системи збалансованого харчування, що призначена для безпосередньої взаємодії користувачів із системою. Основною метою цієї підсистеми є забезпечення ефективного збору і обробки персоналізованих даних про харчові обмеження та вподобання користувачів, зокрема інформації про алергічні реакції, дієтичні обмеження або специфічні запити щодо харчових наборів.

Для досягнення цієї мети фронтенд-підсистема реалізує зрозумілий, інтуїтивний та ергономічний інтерфейс користувача, що значно спрощує процес введення вихідних даних та отримання зворотного зв'язку у вигляді персоналізованих рекомендацій. Концепція відокремлення фронтенд-шару від інших компонентів архітектури має подвійне обґрунтування: по-перше, це дозволяє знизити складність бекенд-підсистем шляхом перенесення первинної валідації введених користувачем даних та взаємодії з користувачем на сторону клієнтського додатку; по-друге, це забезпечує універсальність системи, роблячи її доступною для інтеграції на різних типах платформ (веб-додатки, мобільні додатки, інтерактивні кіоски), що сприяє підвищенню зручності та охопленню ширшої аудиторії користувачів.

Підсистема формування плану харчування – функціональний модуль системи, основне призначення якого полягає у створенні персоналізованих щоденних або тижневих раціонів. Процес формування таких раціонів ґрунтується на заздалегідь визначеному наборі рецептів, дієтичних обмеженнях користувачів, а також рекомендованих показниках калорійності та розподілу макронутрієнтів. При цьому підсистема враховує індивідуальні вподобання користувача, формуючи узгоджені за структурою плани прийомів їжі.

Вбудована логіка гарантує виконання визначених правил та умов, уникнення продуктів з відомими алергенами або контроль за неперевищенням добової норми споживання певних речовин, таких як натрій. Централізація цих правил у окремій підсистемі дозволяє знизити складність інших модулів системи, забезпечує легкість їх супроводу та спрощує процес інтеграції додаткових вимог у майбутньому. У випадку появи нових дієтичних рекомендацій чи медичних обмежень, їх включення здійснюється саме в межах цієї підсистеми без негативного впливу на інші компоненти системи.

Підсистема управління профілями користувачів та даними рецептів забезпечує персоналізацію процесу планування харчування через зберігання та адміністрування необхідних для цього даних. Основними функціями цієї підсистеми є створення, читання, оновлення та видалення профілів користувачів, які містять інформацію щодо їх дієтичних обмежень, харчових уподобань, цільових показників споживання калорій, макроелементів, а також доступного кухонного обладнання. Крім того, до завдань підсистеми входить адміністрування бази даних рецептів, що включає метадані про інгредієнти, технологічні особливості приготування страв, поживну цінність тощо.

Завдяки виділенню операцій з управління користувацькими профілями та даними рецептів у окрему підсистему, архітектура системи залишається структурованою та гнучкою, а логіка формування планів харчування абстрагується від низькорівневих операцій зберігання та обробки даних. Таким чином, решта компонентів системи можуть використовувати стандартизовані формати даних, що полегшує їхню інтеграцію та подальший розвиток.

Підсистема зберігання та управління даними – фундаментальна складова будь-якої системи рекомендаційного характеру, включаючи платформу інтелектуального планування харчування. Вона відповідає за надійне збереження, обробку та оперативний доступ до різномірних типів даних. Для реалізації цих завдань використовуються спеціалізовані сервіси баз даних, такі як Amazon Aurora [75] та DynamoDB [76], призначені для зберігання структурованих транзакційних записів, а також сервіс Amazon S3 [77] для статичних об'єктів. Окрім цього, для ефективного

управління функціями моделей машинного навчання залучається спеціалізований модуль Amazon Feature Store, що є компонентом Amazon SageMaker [78, 79].

Використання різних сховищ даних дозволяє оптимізувати продуктивність пошуку та точність наданих рекомендацій, оскільки кожен тип бази даних відповідає специфічним вимогам щодо доступу та обробки інформації. Сервіси Aurora та DynamoDB забезпечують швидкий пошук користувацьких профілів і даних рецептів: Aurora підтримує транзакції з гарантією властивостей ACID [80], що важливо для збереження узгодженості даних, тоді як DynamoDB ідеально підходить для швидкої роботи з великими обсягами даних типу «ключ-значення». Водночас Amazon Feature Store інтегрує інформацію, необхідну для моделей машинного навчання, зокрема історичні дані харчових вподобань, агреговані нутрієнтні показники, а також векторні представлення рецептів, забезпечуючи послідовність процесів моделювання як у режимі офлайн, так і онлайн.

Підсистема рекомендації рецептів забезпечує персоналізовану фільтрацію та ранжування варіантів страв відповідно до індивідуальних уподобань користувача, враховуючи тип прийому їжі (сніданок, обід, перекус, вечеря). Основним завданням цієї підсистеми є вибір найбільш підходящих рецептів із великого масиву доступних варіантів, враховуючи як явні критерії користувача (дієтичні обмеження або алергії), так і неявні закономірності, що формуються внаслідок аналізу поведінкових даних (переваги щодо смакових характеристик, харчові звички та інше).

Для досягнення зазначеної мети підсистема рекомендації рецептів використовує запропонований метод рекомендації рецептів, що базується на використанні моделей та методів машинного навчання.

Загальний процес формування плану харчування є комплексним механізмом взаємодії підсистем інтелектуальної системи збалансованого харчування, що забезпечує послідовне проходження даних і ухвалення рішень на кожному етапі функціонування системи. Даний процес заслуговує окремого детального розгляду, оскільки визначає, як первинні дані користувача ініціюють послідовність взаємодій між підсистемами управління профілями, рекомендацій страв та безпосереднього формування плану харчування.

Організація цього робочого процесу сприяє запобіганню виникнення вузьких місць у функціонуванні системи, забезпечуючи рівномірний розподіл навантаження і гарантуючи здатність обробляти значні обсяги користувацьких запитів. Крім того, чітко визначений процес забезпечує прозорість формування плану, надаючи можливість адміністраторам відстежувати та аналізувати причини прийняття тих чи інших рішень.

Процес ініціюється на етапі авторизації користувача в системі або при запуску додатку, коли користувачу автоматично завантажуються базові або попередньо сформовані дані з баз даних Aurora або DynamoDB. У разі зміни користувацьких вподобань або обмежень (повідомлення користувачем нових медичних обмежень, таких як непереносимість лактози), ці оновлені параметри передаються через API Gateway до підсистеми формування планів харчування. Ця підсистема, у свою чергу, звертається до функцій AWS Lambda, що реалізують рекомендаційні алгоритми, з метою отримання актуальних рекомендацій рецептів [81, 82].

Отримані рекомендації надалі узагальнюються у комплексний план харчування, де перевіряється відповідність заданим критеріям щодо загальної кількості калорій та макронутрієнтів. За необхідності система виконує корекцію, вилучаючи або замінюючи окремі рецепти, аби забезпечити відповідність сформованого плану заданим обмеженням. Остаточний план записується назад у базу даних і стає доступним для подальшого редагування користувачем у реальному часі.

Таким чином, реалізація циклічного процесу зберігання, обробки, рекомендації та уточнення планів харчування гарантує адаптивність системи до динамічно змінних умов і забезпечує відповідність персоналізованих рекомендацій специфічним вимогам кожного користувача.

3.3 Вибір інструментів для розробки системи

Технологічний стек відіграє важливу роль у забезпеченні масштабованості, продуктивності та легкості підтримки всіх компонентів інтелектуальної системи збалансованого харчування, а обґрунтований підбір технологій дозволяє

оптимізувати розробку окремих підсистем з урахуванням специфіки їх функціональних задач.

Для фронтенд-сервісів та підсистеми формування планів харчування рекомендовано використання мови програмування C#, зокрема платформи .NET Core (.NET 7) [83]. Цей вибір обумовлений надійністю та продуктивністю даної платформи, її корпоративною орієнтованістю та інтегрованими інструментами розробки Visual Studio. Фронтенд-логіка реалізована як API-сервіси з ASP.NET Core або за допомогою ASP.NET MVC та Razor, що забезпечує єдине середовище розробки без додаткових клієнтських мов [83, 84]. Підсистема формування плану харчування виграє від сильної типізації мови C#, ефективної роботи з даними через LINQ, використання Entity Framework Core для взаємодії з реляційними базами даних та допоміжних бібліотек AutoMapper і Polly [85, 86].

Підсистеми рекомендацій та моделювання машинного навчання найбільш ефективно реалізуються мовою Python. Вибір Python обумовлений наявністю широкого спектра бібліотек для машинного навчання (NumPy [87], pandas [88], scikit-learn [89], TensorFlow [90], PyTorch [91]), що забезпечує швидке прототипування та легку інтеграцію нових алгоритмічних рішень. В AWS розгортання моделей та алгоритмів як функцій Lambda у Python дозволяє швидко інтегрувати рекомендаційні механізми у загальний робочий процес. Обробка даних, навчання та розгортання моделей виконується із залученням сервісів Amazon SageMaker.

Зберігання даних розподіляється між реляційними та NoSQL базами залежно від типу даних та завдань підсистеми. Реляційні бази даних забезпечують транзакційну цілісність та структурованість для планування харчування, управління сесіями користувачів та пов'язаними журналами. NoSQL-сховища підходять для зберігання гнучких структур даних, таких як інформація про рецепти та уподобання користувачів, з високою доступністю та масштабованістю. Amazon SageMaker Feature Store використовується для централізованого зберігання та версіювання специфічних для ML функцій.

3.4 Життєвий цикл моделі машинного навчання для методу рекомендацій

Модель машинного навчання методу рекомендацій зазвичай проходить ітеративний життєвий цикл, який включає збір даних, навчання моделі, розгортання та доопрацювання на основі зворотного зв'язку.

На рис. 3.3 показано схему життєвого циклу моделі системи рекомендацій, яка складається з офлайн і онлайн компонентів.

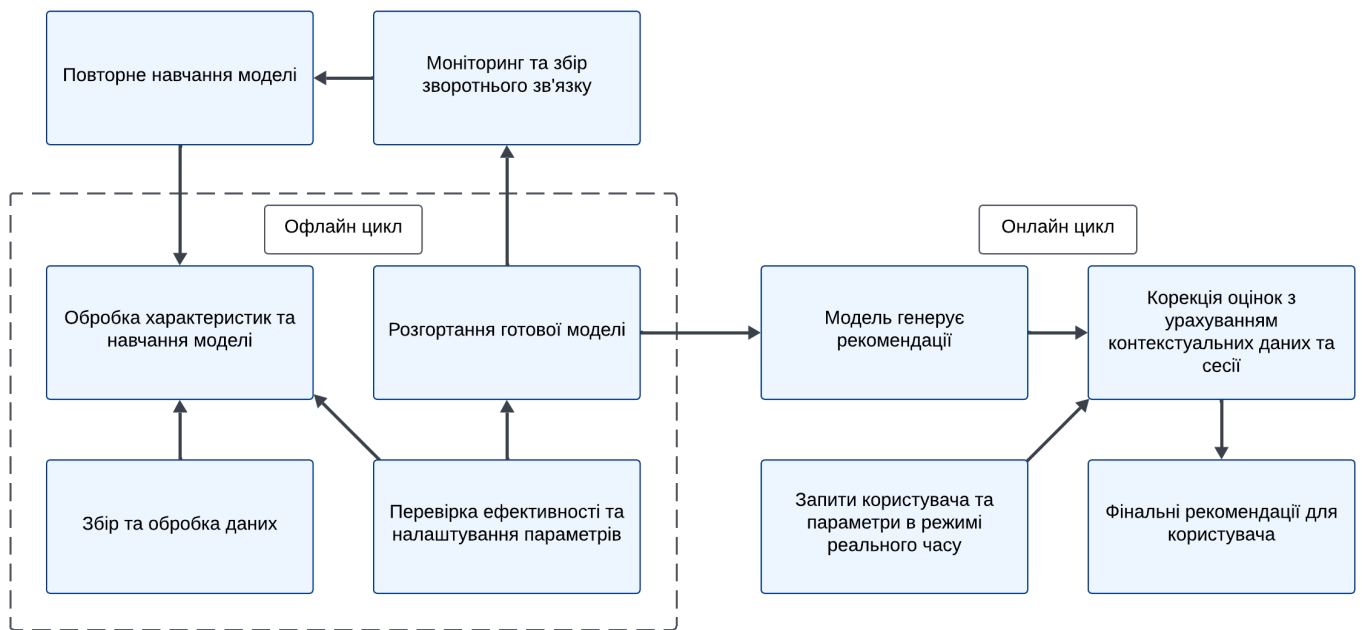


Рисунок 3.3 Схема життєвого циклу моделі машинного навчання методу рекомендацій
рецептів

Спочатку збираються історичні взаємодії і перетворюються на навчальні приклади за допомогою функціональної інженерії. Далі модель навчають передбачати реакцію користувача) для заданих елементів, застосовуючи відповідну регуляризацію та оптимізацію гіперпараметрів для покращення узагальнення.

Після успішної валідації на основі даних, що не були отримані, модель розгортається в реальних умовах і піддається постійному моніторингу, де відстежуються показники ефективності, щоб виявити потенційну деградацію або зміни у вподобаннях користувачів.

Подальше перенавчання з використанням останніх даних дозволяє моделі адаптуватися до еволюції поведінки та каталогів товарів, таким чином підтримуючи актуальність рекомендацій [92].

Хоча історичні дані лежать в основі фундаментального процесу навчання, використання сигналів у реальному часі дає можливість тонкої персоналізації, що виходить за рамки статичних уподобань, зафіксованих в основній моделі. Оновлення в реальному часі дозволяють негайно змінювати порядок рекомендованих елементів, гарантуючи, що система залишається чутливою до ситуативних факторів, таких як поточний настрій користувача, його місцезнаходження та часовий контекст. Поєднуючи довгострокове моделювання вподобань з короткостроковим контекстним зворотним зв'язком, система рекомендацій може пропонувати варіанти, які краще відображають нагальні потреби користувача, що в кінцевому підсумку підвищує задоволеність та залученість користувачів.

Описаний життєвий цикл моделі машинного навчання для рекомендацій рецептів чітко демонструє необхідність уважного проектування системи з метою мінімізації витрат на навчання та підтримку моделей. Ітеративний характер цього циклу, який передбачає регулярний збір нових даних, повторне навчання, постійний моніторинг та інтеграцію зворотного зв'язку в реальному часі, вказує на значні витрати, пов'язані з обчислювальними ресурсами та людськими зусиллями. Кожне перенавчання моделі потребує ресурсомістких етапів функціональної інженерії, налаштування параметрів та регуляризації, що створює постійне навантаження на обчислювальні потужності та збільшує операційні витрати.

Крім того, постійна потреба у моніторингу продуктивності моделей в режимі реального часу означає, що систему попотрібно спроектувати з урахуванням автоматизації таких процесів, щоб мінімізувати втручання людини і таким чином зменшити витрати. Забезпечення ефективності регулярних оновлень за допомогою автоматизованих засобів моніторингу та швидкого розгортання нових версій моделей є важливим фактором зниження загальної вартості користування системою [92].

Отже, життєвий цикл моделі, орієнтованої на персоналізацію рекомендацій, з необхідністю постійної адаптації до змінних переваг і контекстів користувачів,

підкреслює важливість проектування системи таким чином, щоб знижувати витрати на регулярне перенавчання, автоматизувати моніторинг і максимально спрощувати підтримку моделей. Це дозволить забезпечити довготривалу ефективність і економічну доцільність експлуатації інтелектуальної системи планування збалансованого харчування.

У рамках інтелектуальної системи планування харчування реалізовано підхід MLOps (Machine Learning Operations), який забезпечує автоматизацію та моніторинг всіх етапів життєвого циклу моделей. Архітектурно це досягається шляхом інтеграції служби Amazon SageMaker як платформи управління ML-процесами. SageMaker надає інструменти для побудови конвеєрів машинного навчання (SageMaker Pipelines) та реєстру моделей (Model Registry), що централізовано відстежує версії моделей і спрощує їх розгортання [93].

Зокрема, для підзадач рекомендації рецептів різних прийомів їжі налаштовано дві ML-моделі: модель колаборативної фільтрації на основі SVD для рекомендації сніданків та нейронну модель типу Wide&Deep – для обідів, вечерів і перекусів. Обидві моделі тренуються в автоматичному режимі в середовищі SageMaker, де забезпечено проведення експериментів з підбору оптимальних гіперпараметрів, фіксацію метрик якості та збереження артефактів моделей для подальшого використання [94]. Таким чином, архітектура системи охоплює повний цикл роботи з ML-моделями – від багаторазового експериментального навчання до стабільного деплойменту – за рахунок використання хмарних MLOps-інструментів.

Для забезпечення безперервного вдосконалення та оновлення моделей у системі впроваджено конвеєр CI/CD (Continuous Integration/Continuous Delivery), адаптований під потреби ML.

CI/CD охоплює автоматизовані кроки інтеграції змін і доставки оновлених моделей у продакшн-середовище. В репозиторії коду зберігаються версії як програмного коду, так і конфігурацій моделей та даних, що дозволяє відстежувати еволюцію системи. При надходженні нових даних або зміні налаштувань модельного алгоритму конвеєр CI/CD автоматично запускає повторне навчання відповідної моделі. Цей процес реалізовано за допомогою SageMaker Pipelines і служб AWS, таких

як AWS CodePipeline/CodeBuild, які виконують скрипти підготовки даних, тренування та тестування моделі [93].

Після завершення тренування модель проходить автоматизоване оцінювання на відкладеній вибірці; якщо якість моделі відповідає заданим критеріям, її версія реєструється у Model Registry. Версіонування моделей гарантує можливість простежити історію і порівняти ефективність різних ітерацій алгоритмів. Зареєстрована модель маркується у репозиторії як новий реліз і доступна для розгортання [94].

Розгортання реалізується через сервіси контейнеризації SageMaker: створюється оновлена служба рекомендацій (endpoint), доступна через REST API або gRPC. Це забезпечує стандартизований інтерфейс для звернення до моделей зі сторонніх додатків (веб- або мобільного застосунку системи харчування). Окрім основних моделей, конвеєр також оновлює допоміжні структури даних – зокрема, індекси векторних ознак.

В системі використовується AAN-індекс для швидкого пошуку найбільш схожих рецептів за їх векторними поданнями; після кожного переобучення моделі контент-фільтрації цей індекс автоматично перебудовується або оновлюється новими embeddings, щоб відображати актуальний простір ознак [94]. Усі кроки CI/CD конвеєра виконуються без втручання людини, що мінімізує ризик помилок та прискорює впровадження покращених моделей у експлуатацію. Після розгортання моделей рекомендацій критично важливо здійснювати їх безперервний моніторинг у реальному середовищі використання.

По-перше, контролюється якість рекомендаційної видачі за допомогою обраних метрик. Для оцінювання точності рекомендацій застосовується метрика

$Precision@K$ – частка релевантних об'єктів серед топ- K рекомендованих елементів. Ця метрика обчислюється на основі зворотного зв'язку від користувачів і агрегується для оцінки загальної ефективності алгоритму.

Додатково можуть розраховуватися $Recall@K$ та інші метрики ранжування для глибшого аналізу якості рекомендаційної моделі.

По-друге, організовано моніторинг стабільності вхідних даних і характеру запитів, що надходять до моделей. З часом статистичні властивості даних користувачів та рецептів можуть змінюватися (так званий «дрейф ознак»), що призводить до поступового погіршення точності передбачень моделі. Щоб вчасно виявити такі ефекти, система відстежує розподіли вхідних ознак у продакшн-середовищі та порівнює їх з базовими розподілами навчальних даних.

Виявлення значущих відхилень (зсувів) за допомогою статистичних тестів або відстаней між розподілами слугує сигналом про потенційний дрейф даних. При перевищенні порогових значень дрейфу автоматично активується процес перевірки моделі: обчислюється показник деградації якості, і за необхідності ініціюється позачергове перенавчання моделі на актуальних даних. Окрім дрейфу ознак, впроваджено механізми виявлення збоїв у профілюванні користувачів.

Йдеться про ситуації, коли профілі користувачів (їхні вподобання, історія вибору страв тощо) стають неактуальними або неповними, що може бути спричинено зміною поведінки користувача або помилками збору даних.

Система відстежує аномалії у даних профілів – нетипові або порожні профілі – шляхом порівняння статистик поточних запитів користувачів із трендовими значеннями. Якщо деякі сесії користувачів виходять за межі звичних параметрів (є outlier-випадками), то такі випадки логуються окремо для аналізу. Таким чином, багаторівневий моніторинг дозволяє контролювати як точність рекомендацій, так і відповідність вхідних даних припущенням моделі, забезпечуючи своєчасне виявлення проблем у продакшні. Максимальна автоматизація конвеєра машинного навчання дозволяє підтримувати надійність і адаптивність системи без ручного втручання на кожному етапі.

У впровадженій системі планування харчування реалізовано комплекс допоміжних механізмів, що підвищують прозорість та керованість ML-процесів.

По-перше, налагоджено розгалужену систему логування: всі ключові події конвеєра – від запуску тренувань до видачі рекомендацій – фіксуються у журналах (AWS CloudWatch та аналогах) з необхідними параметрами. Логи містять дані про

версії моделей, використані гіперпараметри, результати метрик і час відповіді сервісу, що дає змогу аналізувати історію роботи системи та швидко діагностувати проблеми.

По-друге, на кожному кроці конвеєра виконуються перевірки цілісності даних. Перед тренуванням моделі дані проходять автоматизовану валідацію: контроль діапазонів значень ознак, перевірку відсутності порушених або відсутніх записів, узгодженість форматів тощо. Це запобігає ситуаціям «garbage in – garbage out», коли неякісні вхідні дані могли б призвести до деградації моделі.

По-третє, конвеєр має інтеграцію із системою оповіщень: при настанні визначних подій автоматично генеруються сповіщення для команди супроводу (через електронну пошту, месенджер або дашборд). Завдяки цьому відповідальні особи негайно отримують інформацію про необхідність втручання або про проведені автоматичні заходи (автоперевчання моделі), що підвищує оперативність реагування.

Нарешті, критично важливим компонентом є зворотний зв'язок від кінцевих користувачів, інтегрований у цикл удосконалення моделей. Система збирає дані про взаємодію користувачів із рекомендаціями (вибір запропонованих рецептів, рейтинги страв, відгуки щодо меню). Ці дані зворотного зв'язку потрапляють до сховища та регулярно аналізуються: вони слугують як для обчислення онлайнових метрик ефективності, так і для збагачення тренувального датасету при наступних запусках навчання моделі.

Таким чином формується замкнений цикл навчання з підкріпленням від реального користувацького досвіду: автоматизовані механізми MLOps забезпечують безперервне оновлення моделей на основі нових даних та оцінок, що дозволяє системі рекомендацій адаптуватися до змін у вподобаннях користувачів і підтримувати високий рівень якості рекомендацій протягом тривалого часу.

3.5 Схеми бази даних

На рис. 3.4 показано схему реляційної бази даних системи планування збалансованого харчування людини, що розробляється.

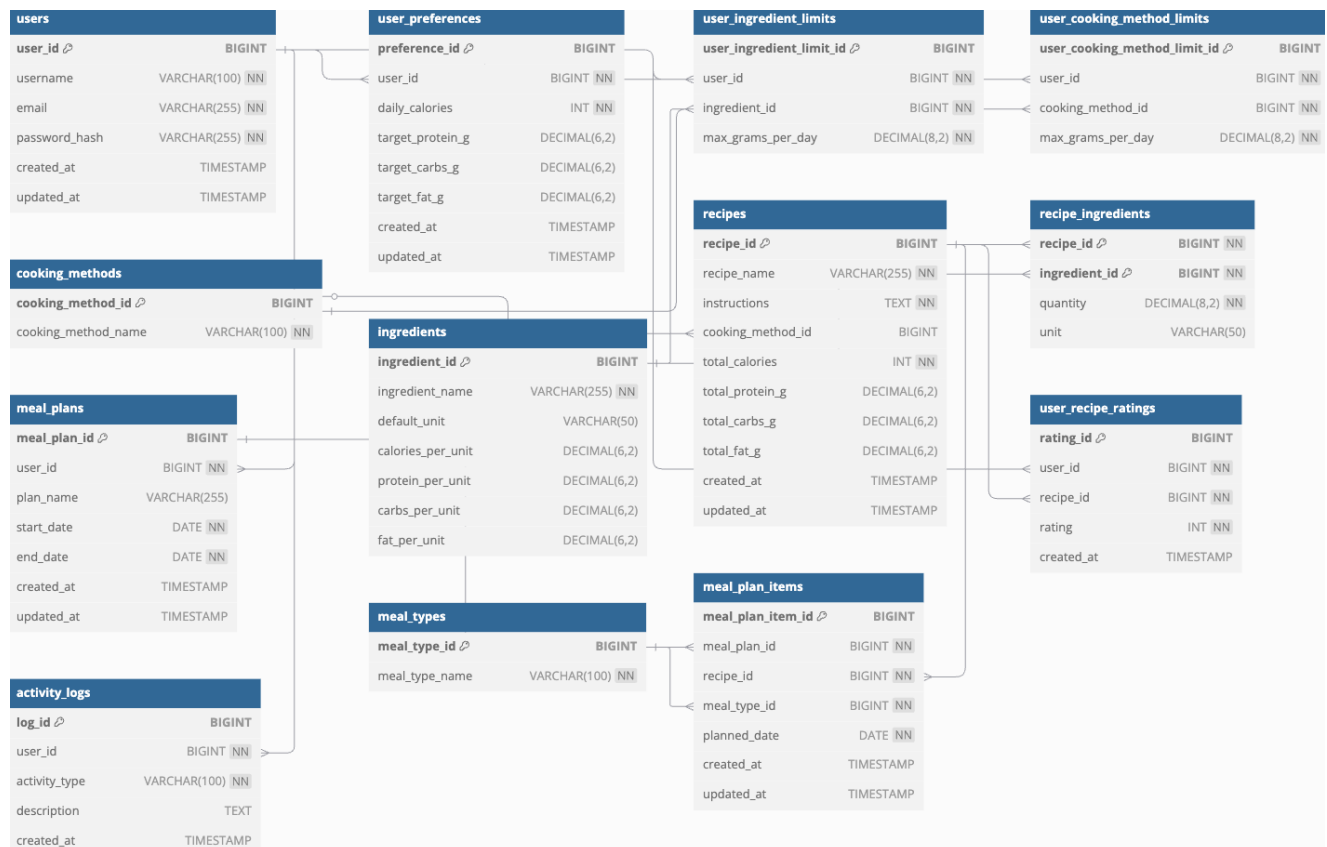


Рисунок 3.4 Схема бази даних системи планування збалансованого харчування
ЛЮДИНИ

Опис основних таблиць, представлених у формальному стилі, з метою забезпечення високого рівня цілісності та ефективності роботи системи:

- *users* – таблиця, що містить інформацію про користувачів системи, зокрема облікові дані, контактні дані, а також метадані, необхідні для процедур автентифікації та авторизації.
- *user_preferences* – зберігає числові цілі щодо харчових показників (добові норми калорій, макронутрієнтів) кожного користувача, що слугують орієнтиром при формуванні планів харчування.
- *user_ingredient_limits* – реєструє індивідуальні кількісні обмеження (у грамах на добу) щодо вживання конкретних інгредієнтів для користувачів, забезпечуючи дотримання персональних обмежень.
- *user_cooking_method_limits* – містить інформацію про кількісні обмеження щодо використання певних методів приготування їжі виражені у грамах на добу, що допомагає у дотриманні заданих обмежень користувача.

- *cooking_methods* – стандартизований список методів приготування кожен з яких має унікальний ідентифікатор для зручності використання у рецептах та обмеженнях.

- *ingredients* – веде детальну реєстрацію продуктів, включаючи їх назви та показники поживної цінності (калорії, білки, жири тощо) на стандартну одиницю виміру.

- *recipes* – таблиця, що містить інформацію про рецепти, включаючи назву, детальні інструкції приготування, загальні поживні характеристики та посилання на метод приготування.

- *recipe_ingredients* – сполучна таблиця, яка встановлює зв'язки між рецептами та інгредієнтами, визначаючи кількість і одиниці виміру кожного інгредієнта в рецепті.

- *meal_plans* – характеризує тимчасовий інтервал планування харчування (щоденний або тижневий), пов'язаний з конкретним користувачем, та містить описову інформацію плану.

- *meal_types* – стандартизований словник типів прийому їжі (сніданок, обід, перекус, вечеря) для уніфікації та ефективного пошуку рецептів у межах планів харчування.

- *meal_plan_items* – таблиця, що пов'язує рецепти з конкретними планами харчування, днями та типами страв, забезпечуючи можливість точного обліку обраних страв та агрегації поживних речовин за період.

- *user_recipe_ratings* (опціонально) – зберігає відгуки користувачів (рейтинги у вигляді оцінок), що сприяють удосконаленню моделей рекомендацій та підвищенню персоналізації планів харчування.

- *activity_logs* – веде журнал важливих користувацьких дій для потреб аудиту та поглибленого аналізу поведінкових тенденцій.

3.6 Модуль формування плану харчування

Наведений в цьому підрозділі псевдокод описує ядро системи, що перетворює вихід триетапного методу рекомендацій рецептів (окремі списки рецептів для сніданків, обідів, перекусів і вечерь) на цілісний меню-план для заданого періоду часу. Цей алгоритм гарантує виконання всіх денних обмежень користувача – калорійних, макронутрієнтних, інгредієнтних і технологічних – та дотримується потрібної структури прийомів їжі згідно з (2.7) – (2.9).

```
FUNCTION GENERATE_MEAL_PLAN(  
    user_id,  
    start_date,  
    end_date,  
    meals_per_day  
) → MealPlan  
  
    day_count ← DAYS_BETWEEN(start_date, end_date) + 1  
  
    limits ← DB.GET_USER_LIMITS(user_id)  
    kcal_daily      ← limits.calories  
    macro_targets   ← limits.macros  
    ingredient_caps ← limits.ingredient_g  
    technique_caps  ← limits.technique_g  
  
    rec_bf ← RECOMMENDER_BF(user_id, day_count * meals_per_day.Breakfast)  
    rec_lu ← RECOMMENDER_L (user_id, day_count * meals_per_day.Lunch)  
    rec_sn ← RECOMMENDER_S (user_id, day_count * meals_per_day.Snack)  
    rec_di ← RECOMMENDER_D (user_id, day_count * meals_per_day.Dinner)  
  
    MealPlan ← {}  
  
    FOR day in 0 .. day_count-1 DO  
        kcal_used      ← 0  
        macro_used     ← {P:0, F:0, C:0}  
        ingr_used      ← DEFAULT_DICT(0)  
        tech_used      ← DEFAULT_DICT(0)
```

```

menu_day    ← {Breakfast:[], Lunch:[], Snack:[], Dinner:[]}

FOR meal_type IN [Breakfast, Lunch, Snack, Dinner] DO
    need ← meals_per_day[meal_type]

    IF meal_type = Breakfast THEN pool ← rec_bf
    ELIF meal_type = Lunch      THEN pool ← rec_lu
    ELIF meal_type = Snack      THEN pool ← rec_sn
    ELSE                        pool ← rec_di
    ENDIF

    i ← 0
    WHILE menu_day[meal_type].size < need AND i < pool.size DO
        cand ← pool[i]

        ok ← kcal_used + cand.kcal ≤ kcal_daily AND
              macro_used.P + cand.P ≤ macro_targets.P AND
              macro_used.F + cand.F ≤ macro_targets.F AND
              macro_used.C + cand.C ≤ macro_targets.C AND
              for_all ingr in cand.ingredients :
                  ingr_used[ingr] + ingr.g ≤ ingredient_caps.get(ingr)

        AND

              for_all tech in cand.techniques :
                  tech_used[tech] + tech.g ≤ technique_caps.get(tech)

        IF ok THEN
            menu_day[meal_type].append(cand)
            kcal_used ← kcal_used + cand.kcal
            macro_used.P ← macro_used.P + cand.P
            macro_used.F ← macro_used.F + cand.F
            macro_used.C ← macro_used.C + cand.C
            FOR ingr in cand.ingredients : ingr_used[ingr] ←
ingr_used[ingr] + ingr.g
            FOR tech in cand.techniques : tech_used[tech] ← tech_used[tech]
+ tech.g
        ENDIF
        i ← i + 1
    ENDWHILE
ENDWHILE

```

```

ENDFOR

IF NOT DAY_IS_COMPLETE(menu_day , meals_per_day) THEN
    FILL_WITH_BEST_EFFORT(menu_day,
                           kcal_daily - kcal_used,
                           macro_targets - macro_used,
                           ingredient_caps - ingr_used,
                           technique_caps - tech_used)
ENDIF

MealPlan[day] ← menu_day
ENDFOR
RETURN MealPlan

```

Алгоритм працює в чотири логічні етапи. Спершу він ініціалізує всі добові ліміти користувача та формує багатий пул рекомендованих рецептів для кожного типу прийому їжі; обсяг пулу щонайменше в k разів перевищує потрібну кількість страв, що мінімізує ризик дефіциту після фільтрації. Далі для кожної календарної доби жадібно перебираються кандидати, починаючи з найрейтинговіших, і перевіряються шість груп обмежень: калорійність, білки, жири, вуглеводи, сумарна вага кожного інгредієнта та сумарна вага страв, приготованих певною технікою.

Рецепт додається до меню лише за повного виконання всіх умов, після чого денні лічильники оновлюються. Якщо після проходження всіх пулів якийсь прийом їжі лишається порожнім, викликається процедура «best effort»: вона послаблює м'які обмеження ($\pm 5\%$ калорій, частковий недобір за макроскладом) або підставляє шаблонні базові страви з низьким ризиком порушити жорсткі ліміти.

3.7 Реалізація механізмів зворотного зв'язку користувачів

У сучасних рекомендаційних системах персоналізованого харчування критично важливо передбачити та врахувати реакції користувачів на запропоновані рецепти. Для цього в системі реалізовано механізми збору як **явного**, так і **неявного** зворотного зв'язку від користувача, що дозволяє комплексно оцінити його вподобання

та поведінкові особливості. Явний зворотний зв'язок передбачає прямі дії користувача, які однозначно сигналізують про ставлення до контенту, тоді як неявний зворотний зв'язок охоплює автоматично відстежувані поведінкові сигнали, що опосередковано відображають інтереси користувача. Комбінація обох підходів (подекуди згадується як гібридний зворотний зв'язок) забезпечує більш надійну систему, мінімізуючи недоліки кожного з типів окремо [95].

Збір явного зворотного зв'язку. Явний зворотний зв'язок здійснюється через інтерфейс системи, де користувач може явно висловити свою думку про рекомендацію [95]. У розробленій системі до явних дій належать: вподобання рецепту, оцінювання (виставлення рейтингу), збереження в меню (додавання рецепту до персонального плану харчування чи списку покупок) та повідомлення про невідповідність (скарга на рецепт, що не відповідає заявленим уподобанням чи обмеженням користувача).

Кожна з цих дій явно вказує на позитивне чи негативне ставлення до певного рецепту. Такий зворотний зв'язок потребує певних дій користувача, але натомість надає високонадійні сигнали щодо його смаків. Зокрема, числові рейтинги та відмітки «подобається» безпосередньо відображають ступінь задоволеності користувача і використовуються для оновлення моделі рекомендацій.

Система передбачає збереження цих оцінок: таблиця *user_recipe_ratings* у базі даних (рис. 3.4) накопичує оцінки користувачів, що сприяє покращенню моделі рекомендацій та рівня персоналізації харчових планів. Явний негативний зворотний зв'язок, як-от скарги на невідповідність рецепту вимогам, є основою для виключення рекомендацій у майбутньому та вдосконалення правил відбору рецептів.

Збір неявного зворотного зв'язку. Нейвний (прихований) зворотний зв'язок фіксується без додаткових дій з боку користувача шляхом аналізу його поведінки на платформі. Система автоматично логує низку поведінкових сигналів: перегляди рецептів (кількість ідентифікованих відкриттів сторінок рецептів), тривалість перегляду (час, проведений користувачем на сторінці рецепту), глибину прокрутки (наскільки повно користувач прогортав сторінку з описом рецепту, що може свідчити про зацікавленість деталями приготування), пропуски або відмови (випадки, коли

користувач швидко закриває рецепт або ігнорує рекомендацію, що можна трактувати як ознаку невдоволення чи нецікавості) та послідовність взаємодій (порядок, у якому користувач переглядає чи обирає рекомендовані елементи). Ці сигнали в сукупності відображають приховані вподобання: довге читання рецепту або перегляд коментарів може вказувати на високу зацікавленість, тоді як швидкий скрол і закриття – на протилежне.

Незважаючи на потенційну нижчу точність порівняно з явними оцінками, неявний зворотний зв'язок особливо цінний своєю масовістю і безперервністю, тому система максимізує використання поведінкових даних для розуміння потреб користувачів [95]. Врахування навіть тонких сигналів (неявні ознаки зацікавленості конкретними інгредієнтами чи техніками приготування) дозволяє рекомендаційному алгоритму точніше моделювати смаки користувача на основі його щоденної активності.

Архітектурна реалізація логування взаємодій. Для обробки та збереження зворотного зв'язку користувачів система містить спеціалізований компонент логування, інтегрований у підсистему збереження і обробки даних. Ключовим елементом є реляційна таблиця *activity_log*, яка є центральним журналом подій взаємодії користувача з системою.

Кожен запис у цій таблиці відображає окрему дію користувача та має такі основні поля: *user_id* (ідентифікатор користувача, що здійснив дію, зовнішній ключ на таблицю користувачів), *recipe_id* (ідентифікатор рецепту або контенту, з яким пов'язана дія, зовнішній ключ на таблицю рецептів), *action_type* (тип здійсненої дії – VIEW, LIKE, RATING, SAVE, REPORT тощо), *timestamp* (мітка часу здійснення дії, що дозволяє відстежувати хронологію та сесійну активність) та *description* (поле для додаткових метаданих, яке зберігає контекстні деталі дії).

Архітектурно таблиця спроектована для ефективного накопичення великої кількості подій і гнучкого аналізу поведінки: поле *action_type* визначає категорію взаємодії, а *description* уніфіковано зберігає специфічні параметри.

Для дії типу RATING у *description* фіксується виставлена оцінка (числове значення рейтингу), для REPORT – причина скарги чи код типу невідповідності, для

VIEW – тривалість перегляду або глибина прокрутки сторінки, а для SAVE – можлива категорія списку (меню на тиждень, список покупок тощо). Такий гнучкий дизайн дозволяє уніфіковано обробляти всі види активностей та зберігати досить деталізовану інформацію про кожну подію.

Данні з *activity_log* використовуються як для аудиту роботи системи (відстеження ключових дій користувачів і виявлення аномалій), так і для глибокого аналізу поведінкових. Взаємозв'язок цієї таблиці з іншими (профілями користувачів, рецептами, рейтингами тощо) забезпечує можливість складання складних запитів для аналітики – виявляти найпопулярніші рецепти за кількістю переглядів, знаходити шаблони відмов від певних категорій страв чи відстежувати динаміку змін вподобань окремого користувача з часом. Логування реалізовано на рівні бекенд-сервісів: кожна взаємодія через фронтенд (веб-або мобільний застосунок) викликає відповідний API-запит, який фіксує подію в *activity_log* одразу при її виникненні.

Таким чином, база даних акумулює повну історію взаємодій, що є джерелом даних для навчання моделей та моніторингу роботи системи в реальному часі.

Використання та значимість зворотного зв'язку. Накопичений зворотний зв'язок користувачів є вирішальним ресурсом для підвищення якості рекомендацій та адаптивності системи.

По-перше, явні оцінки і приховані поведінкові сигнали виступають як навчальні дані для алгоритмів рекомендацій. Історичні взаємодії (класифікації, оцінки) агрегуються і перетворюються на вибірки для тренування моделей ML. На основі цих даних модель коригує свої параметри, щоб точніше передбачувати реакції користувачів на різні рецепти. Зібрані записи фактично формують матрицю «користувач-рецепт», яка оновлюється в міру надходження нового фідбеку; регулярне дозавантаження нових взаємодій і офлайн-навчання на них дозволяє моделі врахувати зміну смаків, появу нових рецептів та інші динамічні фактори.

По-друге, зворотний зв'язок підтримує персоналізацію на індивідуальному рівні: аналіз дій окремого користувача дає змогу тонше налаштувати рекомендації під його унікальний профіль. Якщо користувач систематично висловлює вподобання до вегетаріанських страв або ж часто переглядає рецепти з певним інгредієнтом, система

це враховує при формуванні наступних планів харчування, підвищуючи їх релевантність.

По-третє, онлайн-коригування рекомендацій у режимі реального часу стає можливим завдяки неявному фідбеку: миттєві сигнали від користувача можуть спричинити негайне переналаштування списку рекомендацій у поточній сесії. Система поєднує довгострокові вподобання, засвоєні під час офлайн-навчання, з короткостроковим контекстним зворотним зв'язком, щоб адаптивно реагувати на ситуаційні фактори. Це означає, що навіть після побудови початкового плану харчування платформа продовжує вчитися від користувача: кожен клік, перегляд чи оцінка негайно впливає на ранжування решти рекомендацій, гарантуючи їхню актуальність з урахуванням поточного настрою чи потреб. Такий підхід замикає цикл зворотного зв'язку: від рекомендації – до оцінки користувача – до оновленої рекомендації.

У результаті покращується точність і диверсифікованість персоналізованих планів, адже система оперативно відкидає невдалі варіанти та пропонує нові, більш відповідні альтернативи. Більш того, інтеграція зворотного зв'язку підвищує залученість та задоволеність користувачів: люди бачать, що їхні дії впливають на роботу системи, і отримують контент, який дедалі краще відповідає їхнім очікуванням.

3.8 Вирішення проблеми холодного старту

Проблема «холодного старту» виникає у двох основних випадках:

- коли в систему приходить новий користувач без будь-якої історії взаємодії⁴
- коли додається новий рецепт без даних про його популярність чи оцінки.

У запропонованій інтелектуальній системі механізми подолання холодного старту реалізовано на рівні двох підсистем: підсистеми формування профілю користувача та підсистеми формування профілю рецепту.

Нижче детально розглянуто підходи для обох сценаріїв.

У підсистемі профілю користувача реалізовано початкове анкетування нового користувача. Метою цього опитувальника при реєстрації є явне з'ясування харчових уподобань, потреб і обмежень користувача – так зване сканування вподобань (від англ. *preference elicitation* – метод первинного виявлення вподобань) [96].

Користувач заповнює персональну анкету, що дозволяє зібрати дані для ініціалізації його профілю. Зокрема, збирається така інформація:

- Демографічні дані: вік, стать та інші базові відомості користувача (ці дані враховуються для оцінки індивідуальних потреб, як-от розрахунок добової норми калорій).

- Дієтичні обмеження: медичні або етичні обмеження в харчуванні, такі як харчові алергії, непереносимість продуктів, вегетаріанство/веганство чи інші типи дієт.

- Цілі харчування: індивідуальні цілі та пріоритети, пов'язані з раціоном – зниження ваги, набір м'язової маси, підтримання здорового стану тощо. Ці цілі надалі дозволяють підлаштовувати рекомендації під бажаний.

- Смакові вподобання: уподобані або небажані типи кухні та інгредієнтів. Користувач може зазначити улюблені кухні (національні або регіональні) чи страви, а також продукти, які йому не до вподоби.

Зібрані через анкету дані слугують основою для первинної ініціалізації профілю користувача в системі. На їх основі підсистема профілю користувача створює початковий вектор або набір ознак профілю, який далі використовується рекомендаційним алгоритмом для генерування стартових рекомендацій.

Для нового користувача, що ще не має історії, система комбінує кілька стратегій початкової рекомендації:

- Контентно-орієнтований підбір. На основі явних вподобань, вказаних користувачем, здійснюється підбір рецептів за їх контентними ознаками. Алгоритм зіставляє зібраний профіль користувача з характеристиками рецептів – рекомендує страви з тієї кухні чи з тими інгредієнтами, які відповідають вподобанням користувача [97]. Такий підхід спирається на вміст рецептів і дозволяє надати релевантні рекомендації навіть без поведінкових даних.

– Популярні рецепти. Паралельно застосовується стратегія ненаскрізної популярності: новому користувачу пропонуються рецепти, що мають високий загальний рейтинг або частоту вибору серед усіх користувачів. Такі популярні страви є безпечним вибором на початковому етапі, оскільки мають широке схвалення аудиторії [98]. Це дозволяє відразу залучити користувача, поки система ще не має персональних даних про нього.

– Колаборативна фільтрація за схожими профілями. Система може застосувати знання про уподобання існуючих користувачів, схожих за профілем до нового. Використовуючи метадані анкети, система знаходить сусідів – групу наявних користувачів з подібними демографічними чи дієтичними характеристиками – і рекомендує новачкові рецепти, які були популярними серед цих схожих користувачів. Таким чином реалізується принцип колаборативного підходу: якщо користувач А подібний до користувача В, то вподобані В рецепти можуть бути релевантними для А [99].

Передбачено, що початкові рекомендації будуть поступово уточнюватися у міру надходження даних про поведінку нового користувача. Як тільки користувач почне взаємодіяти з системою – переглядати, вподобати чи оцінювати рецепти – його профіль автоматично оновлюється. Після накопичення достатньої інформації відбувається поступовий перехід від загальних популярних рекомендацій до дедалі більш персоналізованих [100].

Для нових рецептів, які ще не мають історії переглядів чи оцінок, проблема холодного старту вирішується засобами підсистеми профілю рецепту, що відповідає за представлення та зберігання ознак контенту. Кожен рецепт у системі описується набором ключових атрибутів (метаданих), які формують його контентний профіль. При додаванні нового рецепту в базу виконується автоматичний аналіз його вмісту з витягом таких ознак, як:

– Кухня походження: категорія національної або регіональної кухні, до якої належить страва (напр., італійська, азіатська, середземноморська тощо). Ця класифікація дозволяє врахувати культурні смакові уподобання – рекомендувати любителям італійської кухні нові італійські рецепти.

– Склад інгредієнтів: перелік основних інгредієнтів, що входять до рецепту, та їх категорії. Система фіксує, що рецепт містить курятину і помідори, або що в ньому відсутні глютені продукти. Інформація про інгредієнти використовується для контентної фільтрації – щоб підбирати рецепти під наявні вподобання чи виключати ті, що містять алергени або небажані продукти.

– Метод приготування: технологія та техніка, за якою готується страва (запікання, варіння, гриль, смаження, тощо). Ця ознака допомагає врахувати практичні вподобання користувача – якщо користувач надає перевагу стравам на грилі або шукає рецепти без смаження, система може відповідно фільтрувати пропозиції.

– Дієтичні та нутріціологічні властивості: категорії на кшталт вегетаріанська/веганська страва, низьковуглеводна або високобілкова страва, десерт, основна страва тощо. Такі теги відображають, для якої дієти чи ситуації придатний рецепт (напр., «для кето-дієти», «пісна кухня» чи «низькокалорійний обід»). Вони дозволяють співставити рецепт із цілями та обмеженнями користувача – скажімо, новий рецепт з тегом «без цукру» буде одразу доступний у рекомендаціях людям, які дотримуються низьковуглеводної дієти.

На основі виділених контентних ознак підсистема профілю рецепту формує векторне представлення рецепту у багатовимірному просторі ознак. Новому рецепту, навіть без жодного рейтингу, присвоюється змістовний вектор – фактично, його «адреса» в просторі кулінарних уподобань. Це гарантує, що щойно доданий рецепт одразу потрапляє в поле зору рекомендаційного механізму: його можна порівнювати з профілями користувачів та іншими рецептами за схожістю контенту. Таким чином, навіть без історичних даних новий рецепт може бути рекомендований відповідним користувачам, якщо його контентні характеристики відповідають їхнім вподобанням. Якщо рецепт має тег «тайська кухня, гостре, веганське», система знайде користувачів, чий профілі вказують інтерес до тайських веганських страв, і запропонує їм цей новий рецепт. Таким чином, використовуючи контентну схожість, нову страву можна рекомендувати користувачам, які вподобали інші рецепти зі схожими інгредієнтами чи стилем приготування.

Для інтеграції нових рецептів у загальний рейтинг рекомендацій система застосовує гібридний підхід, поєднуючи контентні та колаборативні алгоритми. На початковому етапі переважає контентна складова – адже тільки вона доступна для cold-start рецепту. Проте підсистема профілю рецепту продовжує відслідковувати подальшу взаємодію користувачів з новою стравою. У міру того, як рецепт починає отримувати перегляди, оцінки чи інші сигнали популярності, ці поведінкові дані включаються до його профілю. Зокрема, оновлюється векторна модель рецепту: контентне представлення може бути доповнене або переважене поведінковими ознаками. Таким чином з часом накопичення достатньої кількості даних і колаборативна фільтрація зможе врахувати новий рецепт в стандартних системах рекомендацій ін «теплішає».

В запропнованій системі це досягається завдяки спільній роботі підсистем:

- підсистеми формування профілю рецепту, що забезпечує контентну базу для нового елемента;
- загального рекомендаційного двигуна, що використовує і профілі користувачів, поступово включає новий рецепт до персоналізованих видач по мірі надходження статистики.

Такий гібридний підхід гарантує баланс між експлуатацією перевірених популярних рекомендацій та включенням новинок, забезпечуючи користувачам різноманітність і враховуючи їхні індивідуальні запити навіть для раніше невідомих системі рецептів.

Висновки до розділу 3

1.

Запропоновано і детально описано концептуальну модель та мікросервісну архітектуру інтелектуальної системи персоналізованого планування збалансованого раціону, що підтримує незалежне навчання систем рекомендацій для сніданків, обідів, перекусів і вечерь. Такий поділ систем рекомендацій за категоріями прийому їжі підвищує релевантність рекомендацій і масштабує систему під різні сценарії використання.

2. Описано життєвий цикл моделей машинного навчання, включно з етапами підготовки даних, окремого тренування для кожного прийому їжі, онлайнового моніторингу метрик Precision@K і процедурою перетренування при деградації якості.

3. Спроектовано реляційну схему бази даних та інформаційні потоки, що забезпечують збереження профілів користувачів, рецептів, нутрієнтної інформації й історії взаємодій, мінімізуючи дублювання даних і спрощуючи аналітичні запити.

4. Розроблено модуль формування плану харчування, який перетворює вихід рекомендаційного двигуна на цілісний меню-план для певного періоду та надає користувачеві можливість точкового редагування з автоматичним перерахунком поживної цінності.

5. Передбачено механізм зворотного зв'язку для безперервного покращення якості рекомендацій і реалізовано механізми подолання холодного старту на рівні підсистем формування профілю користувача та формування профілю рецепту.

РОЗДІЛ 4. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ І ВПРОВАДЖЕННЯ МОДЕЛЕЙ МАШИНОГО НАВЧАННЯ В СИСТЕМУ ПЛАНУВАННЯ ХАРЧУВАННЯ

У четвертому розділі описано даних для експериментального дослідження, методологію і середовище експериментів; проаналізовано результати експериментального дослідження перевірки гіпотези щодо доцільності використання окремо навчених моделей машинного навчання для рекомендації різних категорій прийомів їжі; обґрунтовано доцільність використання моделі сингулярного розкладання матриці для генерації меню сніданків і моделі широкого та глибокого навчання для генерації меню перекусів, обідів; представлені опис роботи розробленої системи планування харчування та надані інструкції щодо взаємодії користувача з системою.

4.1. Опис даних для експериментального дослідження

Для навчання систем рекомендацій персоналізованого меню в цьому дослідженні використано набір вхідних даних Food.com [101], який надано веб-сервісом рецептів Food.Com, містить понад:

- 500 000 рецептів від користувачів;
- 1 400 000 користувацьких оцінок рецептів.

Вибір Food.com обґрунтовано його доступністю для широкого кола користувачів і відповідністю для практичних і експериментальних досліджень за вимогам, що описані у [102]. Основний недолік Food.com, що призводить до проблем, однією з яких є неможливість оцінки системи рекомендацій на великій кількості прогнозів через обмеженість кількості оцінок рецептів, що доступні кожному користувачу, – розрідженість даних. Тому розподіл меню на сніданки, обіди, вечері і закуски здійснювався з 100 000 найбільш популярних наборів даних.

При цьому допускалось, що один рецепт може належати до кількох категорій одночасно (рис. 4.1), і певний відсоток рецептів, які не належать до жодної з категорій, визначався як інші.

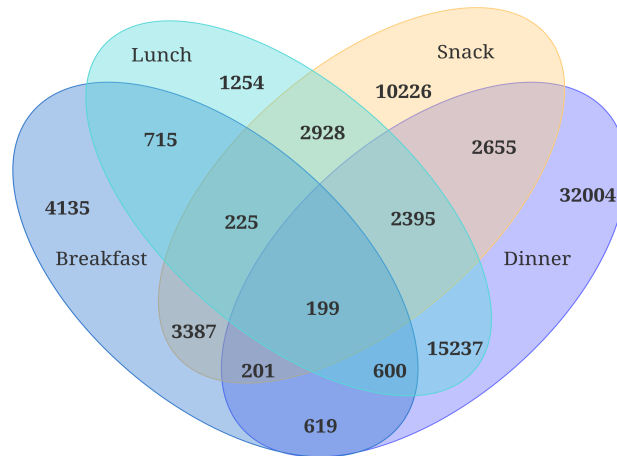


Рисунок 4.1 Діаграма Венна розподілу набору даних за категоріями страв [10],

З 100 000 найбільш популярних наборів даних отримано:

- 7133 рецепти, які були позначені як сніданок;
- 15872 рецепти, які були позначені як обід;
- 15579 рецептів, які були позначені як закуски;
- 42165 рецептів, які були позначені як вечеря.

Однак, незважаючи на те, що ці категорії дещо перетинаються, значна кількість рецептів належить лише до однієї категорії, і це свідчить про те, що кожен тип їжі значною мірою відмінний від інших (рис. 4.1).

Розподіл рецептів за прийомами їжі дозволяє сформувати підмножини даних для окремого навчання систем рекомендації для кожного прийому їжі.

Для кожного з рецептів було визначено набір тегів, які вказують на приналежність рецепту до певного типу харчування. Наразі розглядались такі типи: {веганське, вегетаріанське, без цукру, без лактози, без глютену, кето, палео, з низьким вмістом жиру, низьковуглеводне, з високим вмістом білка}.

Також для кожного з рецептів були визначені алергени, присутні в переліку: {глютен, арахіс, деревні горіхи, молочні продукти, яйця, соя, пшениця, риба, молюски}.

Крім того, набір даних Food.com містить інші дані щодо рецептів, які були корисними при побудові системи рекомендацій: {час на приготування, калорійність та техніка, необхідна для приготування страви}.

Набір даних, що використовувався в цьому дослідженні, містить рецепти та оцінки користувачів, що зібрані протягом 18 років.

Дані були розділені на навчальну (70 %) і тестову (30 %) вибірки випадковим чином. До обох вибірок відбирались тільки ті рецепти, що мали щонайменше 20 відгуків від користувачів. Серед користувачів відбирались тільки ті, що оцінили щонайменше 20 рецептів з навчальної та тестової вибірок. Це дозволяє впевнитись, що для кожного користувача залишається можливість перевірки точності роботи системи на тестовій вибірці. Всі записи в навчальній та тестових вибірках є повними і перевірялись на відсутність протиріч.

Запис набору даних відгуків користувачів, що є вхідним вектором експериментального дослідження, має вигляд: {RecipeId, UserId, Rating, Techniques, Ingredients, IsBreakfast, IsLunch, IsSnack, IsDinner}, де:

- RecipeId та UserId – ідентифікатори рецепту та користувача що залишив відгук відповідно;

- Rating – оцінка користувача;

- IsBreakfast, IsLunch, IsSnack та IsDinner – булеві значення, що вказують належність рецепту до відповідного прийому їжі;

- Techniques – JSON-рядок, що містить масив булевих значень, що вказують чи застосовується відповідна техніка приготування в даному рецепті.

- Ingredients – масив чисельних векторів, що репрезентує вектори інгредієнтів страви.

Набори даних відгуків користувачів для сніданків, обідів, перекусів і вечерь є підмножинами загального набору даних відгуків користувачів.

Набір даних відгуків користувачів має 325568 записів, для 49251 унікальних рецептів і 99338 унікальних користувачів, при чому:

- набір для сніданків містить 47981;

- набір для обідів містить 90398;

- набір для перекусів містить 86762;
- набір для вечерь містить 120427 записів.

В наборі даних представлені оцінки всі оцінки за шкалою від 1 до 5. При цьому враховано те, що різні користувачі мають різне розуміння шкали оцінювання і схильність до завищення чи заниження оцінок, і для мінімізації впливу суб'єктивізму було проведено нормалізацію оцінок страв користувачами.

На рис. 4.2 показано схему потоку даних при експериментальному дослідженні.

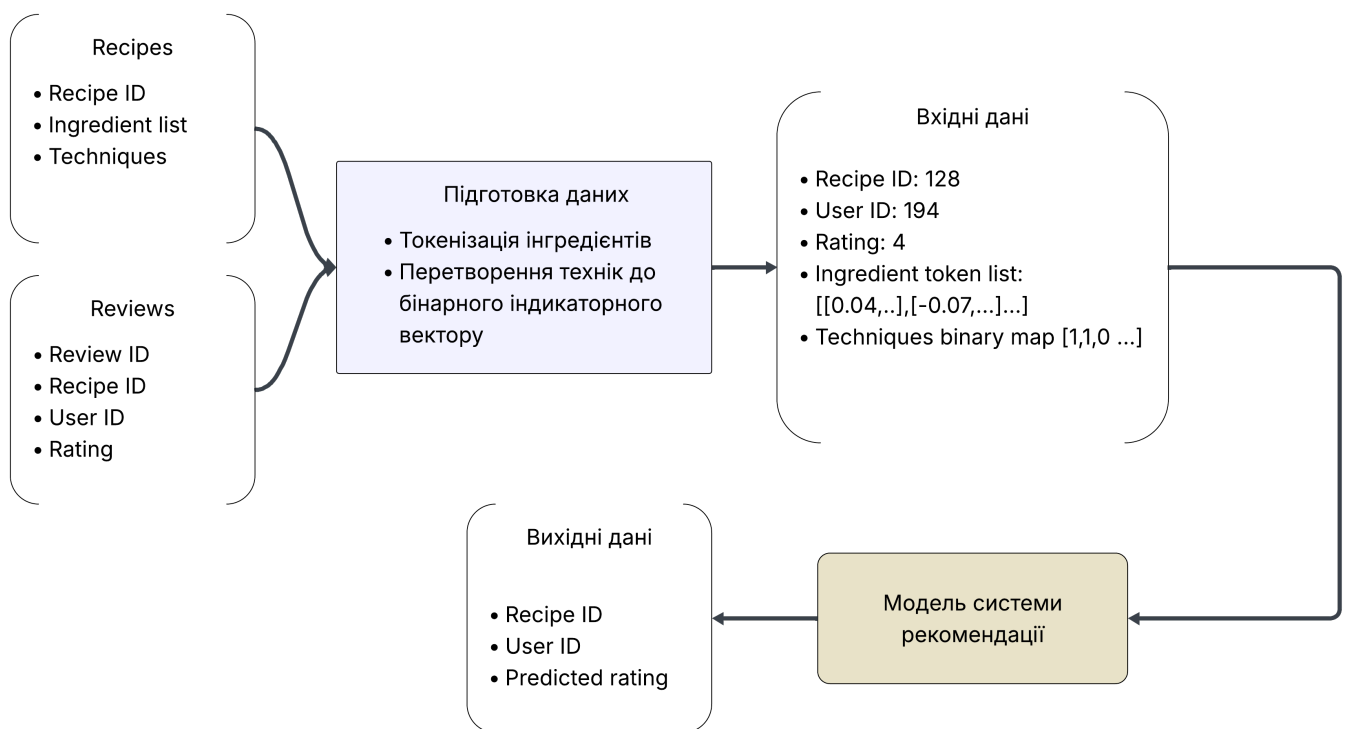


Рисунок 4.2 схема потоку даних при експериментальному дослідженні моделей ML

4.2. Методологія та середовище експериментів

У цьому експерименті перевіряється гіпотеза, що використання окремо навчених оцінювальних моделей для кожної категорії прийому їжі (сніданок, обід, перекус, вечеря) забезпечує вищу якість рекомендацій порівняно з єдиною універсальною моделлю, яка однаково обробляє всі позиції меню. Для перевірки гіпотези, перевірялась лише модель машинного навчання, відповідальна за етап «оцінювання», залишаючи без змін генерацію кандидатів і фінальне ранжування.

Такий підхід забезпечує чистоту експерименту, адже дозволяє однозначно приписати всі відмінності у показниках точності саме зміні ядра етапу оцінювання. Генератор кандидатів слугує переважно recall-орієнтованим фільтром: він швидко відбирає потенційно релевантні рецепти спільним embedding-пошуком і майже не використовує персональних характеристик; тому приріст якості завдяки його перенавчанню був би мінімальним при суттєвих обчислювальних витратах. Натомість оцінювальний шар є саме тим місцем, де моделі враховують неявні вподобання та категорію прийому їжі.

Етап ранжування, своєю чергою, є детермінованим: рецепти сортуються за обчисленим скором і проходять жорсткий нутрієнтний фільтр, тому він не містить параметрів, які можна було б оптимізувати навчанням. Також зосередження на моделі машинного навчання для етапу оцінювання мінімізує затрати часу на перенавчання, зберігає низьку затримку у продакшн-середовищі й прямо відповідає концепції трьох етапного методу рекомендації рецептів [10].

Всі моделі машинного навчання навчались і тестувались в хмарному середовищі Google Colab [103], що дає змогу оперативно використовувати апаратне прискорення без додаткових витрат на розгортання власних обчислювальних серверів.

Для розробки системи рекомендацій використано бібліотеки surprise [104], LightGBM [57] і tensorflow [90] і такі вхідні дані, як вподобання користувача, особливості продуктів і контекстна інформація.

В процесі експериментів:

- основною мовою програмування була Python версії 3.11;
- застосовувався графічний процесор NVIDIA T4 GPU з 16 GB GDDR6 оперативної пам'яті, що забезпечує достатній запас обчислювальної потужності навіть для моделей глибокого навчання;
- для роботи з даними використовувались бібліотеки pandas та numpy;
- весь програмний код був структурований в JupiterNotebooks [105].

4.3. Параметри моделей

Модель SVD навчається протягом заданої кількості епох, що є одним з ключових гіперпараметрів моделі. Модель SVD може зупинити навчання раніше, якщо протягом кількох епох метрика RMSE не покращується.

В цій роботі під час проведення експериментального дослідження кількість епох для всіх навчених моделей SVD становила 30, а кількість латентних факторів d становила 100 [10].

Одним з ключових гіперпараметрів моделі GBDT є кількість епох (M), під час яких вона навчається. В цій роботі під час проведення експериментального дослідження для всіх моделей GBDT становила $M=200$, але модель може зупинити навчання раніше, якщо протягом кількох епох метрика RMSE не покращується. Параметр усадки $\gamma=0.05$ [10].

Структуру широкої компоненти моделі Wide&Deep, яку вибрано для експериментального дослідження, показано на рис. 4.3.

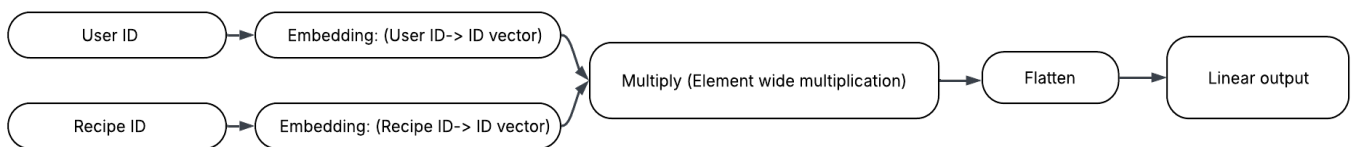


Рисунок 4.3 Структура широкої компоненти моделі Wide&Deep.

Вхідний вектор широкої компоненти складається з двох основних вхідних даних: `UserId` та `RecipeId`, кожен з яких відображається на вбудовування, що перетворюють ці категоріальні змінні на щільні одновимірні вектори. Потім ці вбудовування піддаються поелементному множенню, фіксуючи взаємодію між ознаками. Отриманий результат згладжується в єдиний вектор, який пропускається через лінійний шар [10]. Цей шар обчислює зважену суму вхідних даних, виводячи скалярне значення. Простота цієї архітектури дозволяє компоненту Wide ефективно моделювати прямі асоціації між характеристиками користувача і рецепта.

При навчанні моделі Wide&Deep забезпечення надійності прогнозування рейтингу передбачає мінімізацію розбіжності між прогнозованим і фактичним рейтингами за допомогою MSE(4) з L2 регуляризацією, що використовується для уникнення перенавчання [10].

Структуру глибокої компоненти моделі Wide&Deep, яку вибрано для експериментального дослідження, показано на рис. 4.4.

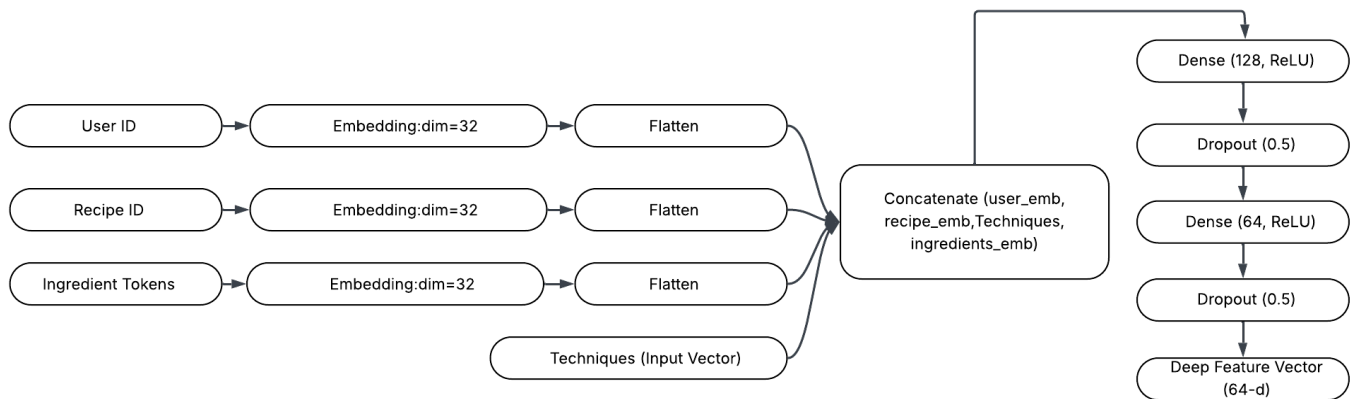


Рисунок 4.4 Структура глибокої компоненти моделі Wide&Deep

Вхідний вектор глибокої компоненти Wide&Deep складається з: UserId, RecipeId, Ingredients і Techniques, які представляють:

- унікальні ідентифікатори авторів і рецептів;
- вектори інгредієнтів а також вектор характеристик технік, що використовуються для приготування страв.

Для категоріальних вхідних даних UserId і RecipeId ця компонента Wide&Deep використовує окремі шари вбудовування, що перетворюють ці розріджені ідентифікатори в щільні 32-вимірні представлення [10].

Після кожного шару вбудовування виконується операція згладжування, яка перетворює двовимірні результати вбудовування в одновимірні вектори.

Вхідні дані необроблених технік для приготування рецептів не вбудовуються, а використовуються безпосередньо, як категоріальні дані.

Згладжені вбудовування і вектор методів об'єднуються за допомогою конкатенації, формуючи єдиний вектор ознак, який містить всю необхідну інформацію [10].

Цей об'єднаний вектор ознак подається на серію прихованих шарів, де:

- проходить через щільний шар зі 128 нейронами з використанням функції активації ReLU, яка фіксує нелінійні зв'язки між ознаками;
- проходить шар відсіву з коефіцієнтом відсіву 0.5, щоб зменшити надмірне пристосування шляхом випадкової деактивації половини нейронів під час навчання;
- вихідні дані першого шару відсіву обробляються другим щільним шаром з 64 нейронами, також з використанням активації ReLU, за яким слідує шар відсіву з коефіцієнтом відсіву 0.5;
- кінцевим результатом глибинної складової є 64-вимірний вектор ознак, який представляє вивчені взаємодії ознак високого порядку. Ці ознаки сприяють загальному прогнозу при об'єднанні з широким компонентом.

Wide&Deer навчається протягом M епох, що є одним з гіперпараметрів моделі. В цій роботі під час проведення експериментального дослідження $M=20$, але модель може зупинити навчання раніше, якщо протягом кількох ітерацій метрика RMSE не покращується [10].

4.4 Експериментальне дослідження моделей для кожного прийому їжі

Для експериментального дослідження моделей-кандидатів в системи рекомендації кожного виду прийому їжі були вибрані моделі SVD, GBDT і Wide&Deer, оскільки вони [10]:

- здатні виявляти складні зв'язки між характеристиками страв і вподобаннями користувачів, що дає можливість формувати рекомендації навіть якщо індивідуальні смаки відхиляються від математичних сподівань параметрів, виявлених у загальному масиві даних;
- здатні ефективно працювати з такою складною і розрідженою інформацією про страви, як використання різних інгредієнтів і способів приготування, враховуючи при цьому суттєві особливості структури даних;
- охоплюють різні методи машинного навчання: матричну факторизацію, деревоподібні методи і глибинне навчання.

Крім того, наявність популярних фрейм ворків спрощує їх впровадження, налаштування і усунення несправностей. Нарешті, обчислювальні ресурси цих моделей дозволяють відносно гнучко налаштовувати гіперпараметри під час експериментів, не перевищуючи доступні апаратні обмеження.

Модель, яка навчалась на наборі даних, що містить всі рецепти та оцінки користувачів, в цій роботі називається універсальною конфігурацією [10].

Модель, яка навчалась на наборі даних, що містить тільки рецепти сніданків, обідів, перекусів чи вечерь, в цій роботі називається спеціалізованою конфігурацією [10].

Для тренування універсальної конфігурації використовувався набір, що містив усі рецепти і оцінки з [101], а для тренування спеціалізованою конфігурації використовувались набори, які включали лише рецепти для однієї категорії прийому їжі.

Для перевірки гіпотези щодо покращення ефективності роботи систем рекомендації при окремому навчанні проведено дослідження, в ході якого [10]:

- кожна з моделей SVD, GBDT і Wide&Deer була натренована на 5 наборах даних;

- для страв сніданків, обідів, перекусів і вечерь усіма конфігураціями SVD, GBDT і Wide&Deer були сформовані таблиці персональних рекомендацій.

Кожен із наборів даних був розділений на тренувальний (70 % даних) і тестовий (30 % даних). Усі моделі було натреновано на навчальних, а потім перевірено їх ефективність на тестових.

Щоб зрозуміти наскільки релевантні перші 10 рекомендацій в середньому:

- після навчання усіх конфігурацій SVD, GBDT і Wide&Deer було оцінено Precision@10 кожної моделі для всіх користувачів;

- на основі цих оцінок запропоновано рекомендувати кращу систему для страви кожної категорії прийомів їжі.

На рис. 4.5 показано динаміку метрики MSE при навчанні універсальної і спеціалізованих конфігурацій моделі SVD для сніданків, обідів, перекусів і вечерь [10].

Динаміка навчання різних конфігурацій SVD показує, що [10]:

- моделі всіх конфігурацій для сніданку, обіду, перекусу закінчили навчання на 15-ій епосі;
- моделі обох конфігурації для вечері закінчили навчання на 20-ій епосі.

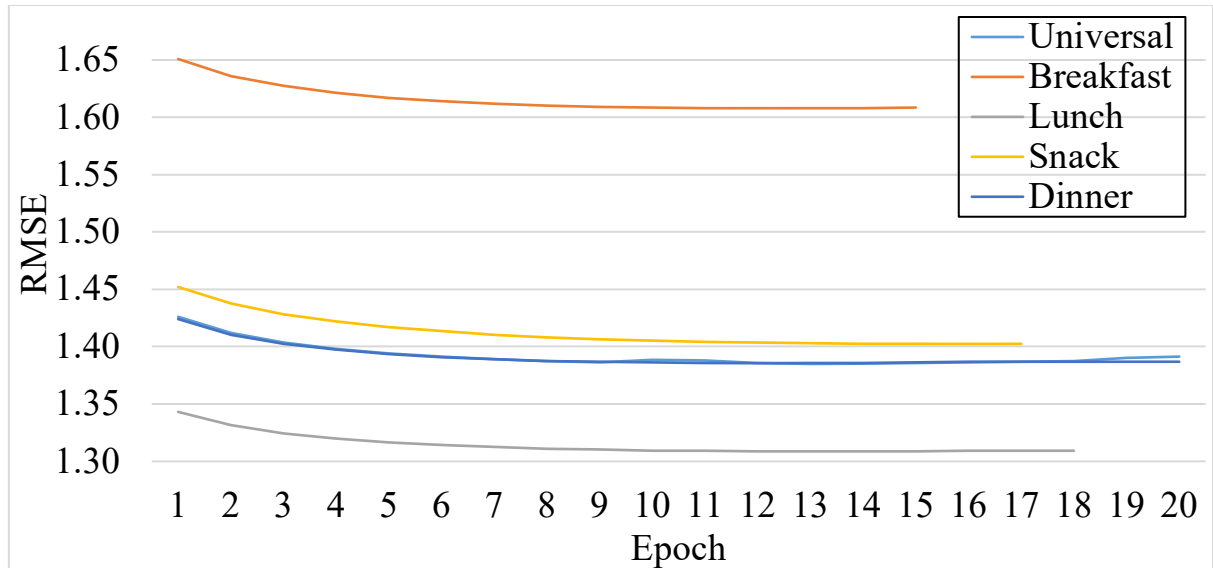


Рисунок 4.5 RMSE при навчанні універсальної і спеціалізованих конфігурацій моделі SVD [10]

На прикладі моделі SVD також показано перші 10 персональних рекомендацій і оцінки страв сніданку, що надані користувачеві з id=169430 універсальними і спеціалізованими моделями різних конфігурацій (табл. 4.1) [10].

Дані табл. 4.1 показують, що при заданому порозі релевантності 4,0 [10]:

- універсальна модель рекомендувала 8 з 10 релевантних страв, оскільки реальну оцінку меншу, ніж 4.0 мають страви під номером 9 і 10;
- спеціалізована модель рекомендувала 9 з 10 релевантних страв, оскільки реальну оцінку меншу, ніж 4.0 має тільки страву під номером 9.

Таким чином Precision@10 універсальної і спеціалізованої моделей для страв сніданку, запропонованих користувачеві з id=169430, становлять 0,8 і 0,9, відповідно. Аналогічно були сформовані таблиці для обідів, перекусів чи вечері, що надані універсальною і спеціалізованою моделлю SVD [10].

Середні значення Precision@10 універсальних і спеціалізованих SVD, що навчені для рекомендації страв різних категорій їжі, надані в табл. 4.2 [10].

Таблиця 4.1

Приклад страв сніданку, що надані користувачеві з id=169430, універсальною і спеціалізованою конфігураціями SVD [10]

Конфігурація SVD	№	ID страви	Оцінка за 5-бальною шкалою	
			Реальна	Передбачена
Універсальна	1	Banana nut bread	5.0	5.0
	2	Creamy scrambled eggs	5.0	5.0
	3	Raisin spice hot cereal with quinoa	5.0	5.0
	4	Onion bread for the abm	5.0	4.9787
	5	Nutty banana oatmeal	5.0	4.9404
	6	Low fat banana buttermilk muffins	5.0	4.9281
	7	Doctoredup scrambled eggs	5.0	4.8805
	8	Country cornbread	5.0	4.7166
	9	<i>Chocolate fudge power oatmeal</i>	3.0	4.5393
	10	<i>Cheese bacon breakfast muffins</i>	3.0	4.3954
Спеціалізована	1	Creamy scrambled eggs	5.0	5.0
	2	Low fat banana buttermilk muffins	5.0	5.0
	3	Bacon egg and cheese biscuit	5.0	4.9888
	4	Baked sicilian frittata	5.0	4.9470
	5	Tabasco cheddar biscuits	5.0	4.8864
	6	Broccoli cheese frittata	5.0	4.8736
	7	Orange almond scones	5.0	4.8683
	8	Banana nutbread	5.0	4.8253
	9	<i>Scrambled egg wrap</i>	3.0	4.3265
	10	Country cornbread	5.0	4.2023

Порівняння середнього Precision@10 для різних конфігурацій SVD [10]

Категорія прийому їжі	Середнє Precision@10 моделі SVD (pp/np)		
	Універсальна конфігурація	Спеціалізована конфігурація	Покращення, %
Сніданок	0.7951	0.9423	18
Обід	0.8058	0.9046	12
Перекус	0.8069	0.8939	10
Вечеря	0.7652	0.8856	15

Такі самі експериментальні дослідження для всіх категорій прийомів їжі були проведені для всіх конфігурацій GBDT і Wide&Deep. При чому у процес побудови рекомендацій цими моделями включено різні техніки приготування їжі і оцінка їх впливу на вподобання [10].

Рис. 4.7 показує динаміку метрики RMSE на кожній епосі навчання універсальної і спеціалізованих конфігурацій моделі GBDT для сніданків, обідів, перекусів і вечерь [10].

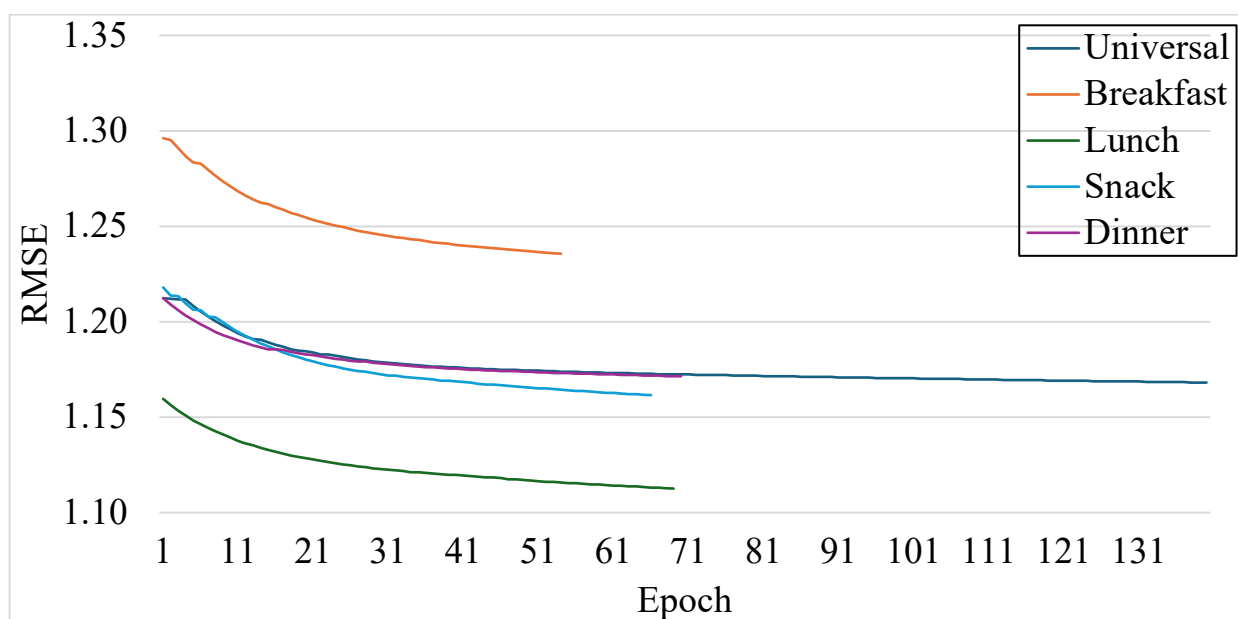


Рисунок 4.7 RMSE при навчанні універсальної і спеціалізованих конфігурацій моделі GBDT [10]

Динаміка навчання різних конфігурацій GBDT показує, що [10]:

– модель з конфігурацією для сніданків, показала найбільшу похибку RMSE серед всіх конфігурацій, включаючи універсальну, при цьому модель для обідів показала найнижчу похибку;

– спеціалізовані моделі навчаються значно швидше, закінчуючи навчання передчасно в околі 50-ти епох, тоді як універсальна модель закінчила навчання в околиці 130 епох, оскільки метрика RMSE перестала покращуватись [10].

Середні значення Precision@10 універсальних і спеціалізованих GBDT, які навчені для рекомендації страв різних категорій їжі, надані в табл. 4.3 [10].

Таблиця 4.3

Порівняння середнього Precision@10 для різних конфігурацій GBDT [10]

Категорія прийому їжі	Середнє Precision@10 моделі GBDT (pp/np)		
	Універсальна конфігурація	Спеціалізована конфігурація	Покращення, %
Сніданок	0.7115	0.8613	21
Обід	0.6592	0.8331	26
Перекус	0.7257	0.8479	17
Вечеря	0.6372	0.7528	18

Рис. 4.8 показує динаміку метрики RMSE при навчанні універсальної і спеціалізованих конфігурацій моделі Wide&Deer для сніданків, обідів, перекусів і вечерь [10].

Динаміка навчання різних конфігурацій Wide&Deer показує, що:

– всі конфігурації закінчили навчання під час 11 епохи;

– конфігурація для сніданків показала найвищий показник похибки RMSE, тоді як конфігурація для вечерь і універсальна конфігурація показала найнижчі значення цієї метрики [10].

Середні значення Precision@10 універсальних і спеціалізованих Wide&Deer, що навчені для рекомендації страв різних категорій їжі, надані в табл. 4.4 [10].

Таким чином в результаті окремого навчання для кожного прийому їжі ефективність роботи за метрикою Precision@10 покращилась для усіх моделей.

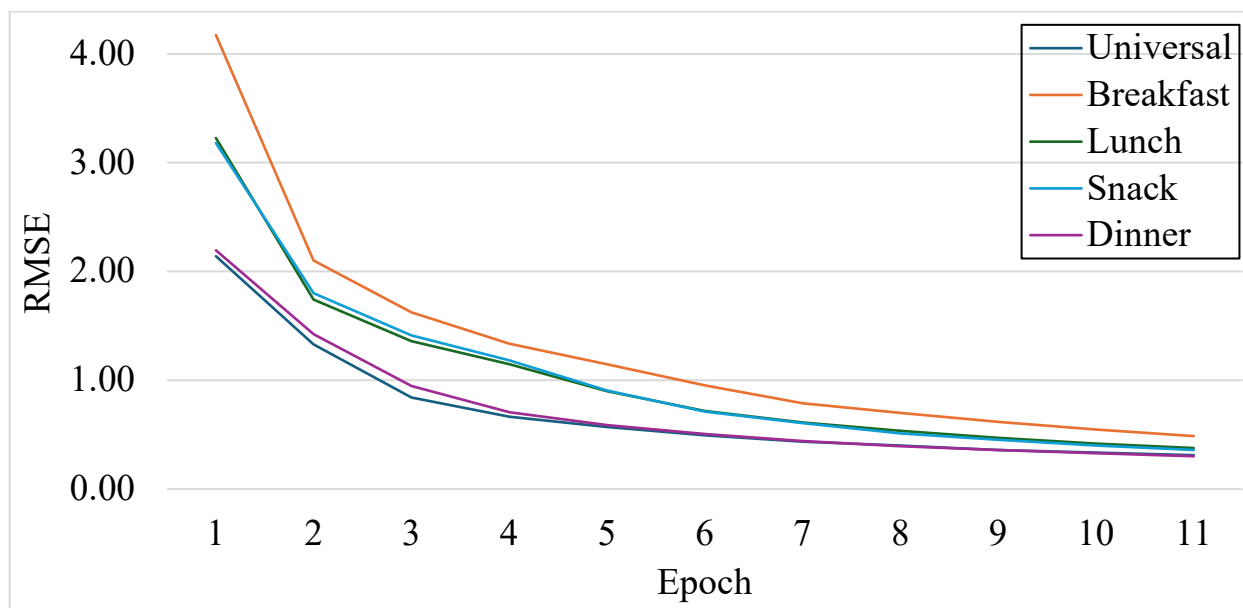


Рисунок 4.8 RMSE при навчанні універсальної і спеціалізованих конфігурацій моделі Wide&Deep [10]

Таблиця 4.4

Порівняння середнього Precision@10 для різних конфігурацій Wide&Deep [10].

Категорія прийому їжі	Середнє Precision@10 моделі Wide&Deep (pp/np)		
	Універсальна конфігурація	Спеціалізована конфігурація	Покращення, %
Сніданок	0.7631	0.9210	20
Обід	0.8155	0.9609	18
Перекус	0.8154	0.9768	19
Вечеря	0.7719	0.9509	23

4.4 Обґрунтування вибору моделей систем рекомендації рецептів

На рис. 4.9 показано стовпчикові діаграми, що по осі ординат показують середнє значення Precision@10 для моделей SVD, GBDT і Wide&Deep [10].

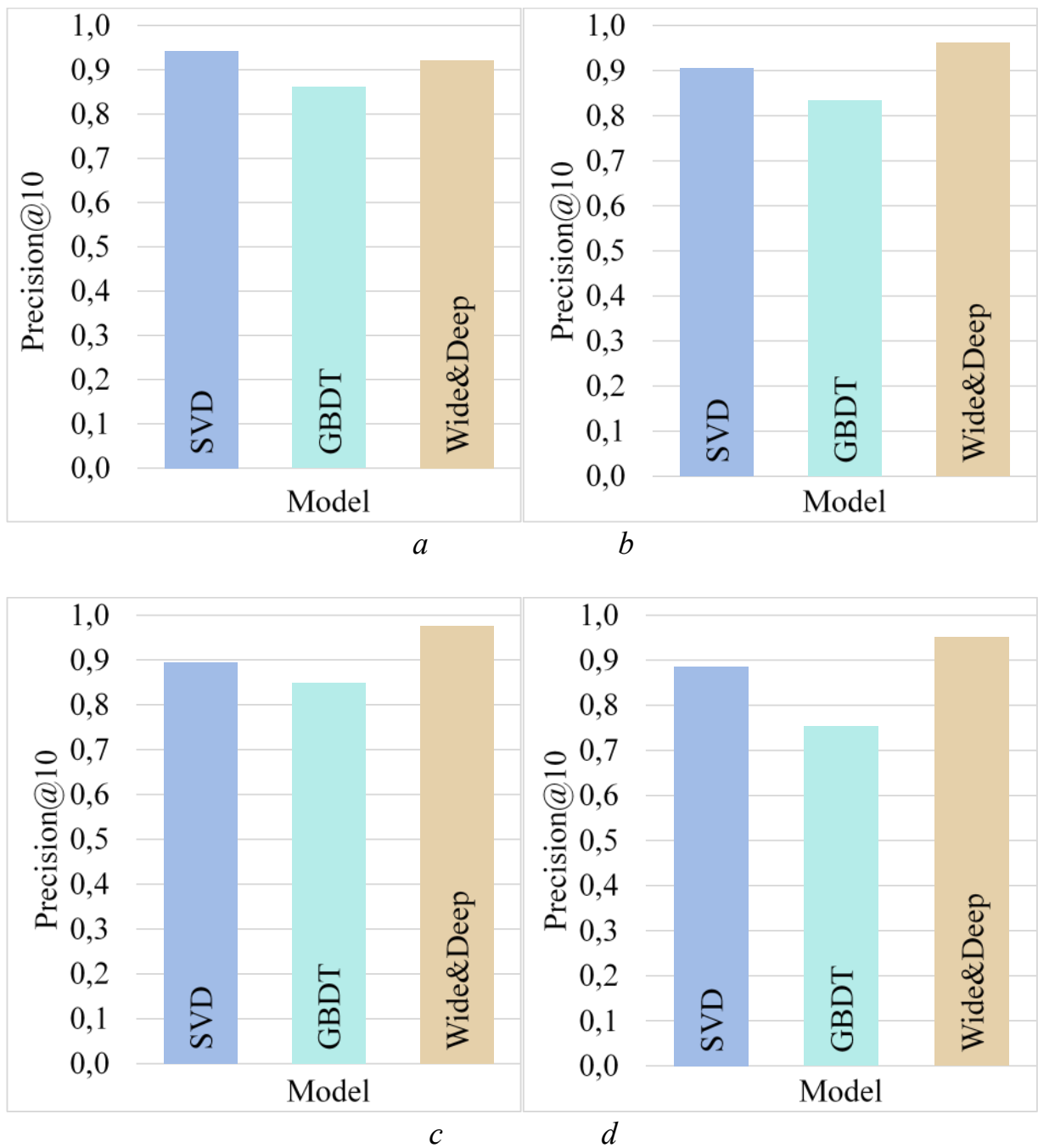


Рисунок 4.9 Діаграми середнього Precision@10 спеціалізованих конфігурацій SVD, GBDT і Wide&Deep для: *a* – сніданків, *b* – обідів, *c* – перекусів, *d* – вечерь [10]

Ці діаграми показують, що за метрикою Precision@10 GBDT демонструє найгірші результати: тільки близько 86 % перших 10 рекомендованих користувачеві страв сніданків є релевантними, тоді як моделі SVD і Wide&Deep покращують цю метрику для сніданків до 94 % і 92 % відповідно.

Модель Wide&Deer показала найбільшу ефективність за Precision@K для обідів, перекусів і вечерь, рекомендуючи в середньому близько 96 % релевантних страв для обідів, 97 % для перекусів та 95 % для вечерь [10].

Таким чином, результати експериментального дослідження надають підстави стверджувати, що доцільність вибору моделі SVD і Wide&Deer для подальшого вдосконалення інтелектуальної системи персоналізованого планування збалансованого раціону харчування є достатньо обґрунтованою [10].

4.5 Результати та аналіз експериментального дослідження моделей-кандидатів

Результати процесу окремого навчання моделей-кандидатів для систем рекомендації кожного виду прийому їжі показали, що [10]:

- спеціалізовані моделі SVD і GBDT навчаються впродовж меншої кількості епох, ніж універсальна система, і показують не гірше фінальне значення метрик після навчання для всіх прийомів їжі, окрім сніданків і перекусів для SVD та сніданків для GBDT;

- універсальна і спеціалізовані моделі Wide&Deer навчалися впродовж однакової кількості епох і показали схожі значення фінальних метрик.

Результати експериментального дослідження моделей-кандидатів для систем рекомендації кожного виду прийому їжі показали, що окремо навчені спеціалізовані системи для сніданків, обідів, перекусів і вечерь показують збільшення ефективності роботи за метрикою Precision@10 в порівнянні з універсальною системою [10]:.

Покращення склали від 10 % до 26 %. Таке зростання пояснюється тим, що спеціалізовані моделі навчаються на більш однорідними наборами прикладів, завдяки чому краще виявляють закономірності, характерні саме для відповідного типу страв. Обмежений контекст означає меншу кількість конфліктної чи зайвої інформації, що полегшує задачі пошуку і ранжування рецептів за релевантністю [10]:.

Водночас універсальна модель, охоплюючи всі різновиди страв, може отримувати переваги від більшого обсягу даних, однак втрачає частину точності через неоднорідність, що виникає через змішування різних категорій рецептів [10].

За результатами порівняння ефективності моделей SVD, GBDT і Wide&Deep за метрикою Precision@10 для рекомендації сніданків, обідів, перекусів та вечерь, обґрунтовано доцільність використання [10]:

- конфігурацію SVD для систем рекомендації сніданків;
- конфігурації Wide&Deep для систем рекомендації обідів, перекусів і вечерь.

Краща ефективність за метрикою Precision@10 систем рекомендації сніданків на основі моделі SVD пояснюються тим, що способи приготування блюд, які наявні в наборі даних [97], менше впливають на вподобання користувачів. А те, що для інших категорій кращі результати показала система на основі Wide&Deep, пояснюються тим, що поєднання узагальнення і запам'ятовування позитивно впливає на продуктивність систем рекомендацій [10].

4.6 Взаємодія користувача з системою

Інтерфейс інтелектуальної системи збалансованого харчування розроблений з урахуванням інтуїтивної зрозумілості, проте нижче представлено детальні інструкції щодо основних функцій та взаємодії з системою (рис. 4.10).

Реєстрація та налаштування профілю користувача

- Реєстрація: Нові користувачі створюють особистий обліковий запис, надаючи базову інформацію (електронну адресу, пароль, ім'я).
- Авторизація: Зареєстровані користувачі вводять свої облікові дані для доступу до системи.
- Профіль користувача: Після авторизації користувачі вводять інформацію про вік, вагу, зріст та стать для визначення індивідуальних калорійних потреб.
- Рівень активності та цілі: Обирається рівень фізичної активності (низький, помірний, високий) та визначається основна мета харчування (зниження ваги,

підтримка ваги або набір м'язової маси). На основі цих даних система розраховує щоденні потреби в макроелементах.

Профіль користувача

Ім'я: Людмила Вік: 28 Вага (кг): 55 Зріст (см): 164

Стать: Жінка Рівень активності: Низька активність

Алергії

Кунжут Видалити

Кунжут Додати

Дієтичні обмеження

Немає Додати

Обмеження щодо здоров'я

Немає Додати

Зберегти

Рисунок 4.10 Знімок екрана з формою редагування профілю користувача, що містить такі поля, як вік, вага, зріст, рівень активності та обмеження в харчуванні (алергії, кулінарні вподобання).

Встановлення обмежень у харчуванні:

- Інгредієнти: У розділі «Обмеження» користувачі зазначають максимальну добову кількість інгредієнтів («цукор ≤ 30 г/день»).
- Методи приготування: Користувачі можуть встановити добові межі для використання певних методів приготування їжі («смаження ≤ 100 г/день»).
- Алергії та медичні обмеження: Відповідні обмеження (алергії на горіхи чи медичні дієтичні рекомендації) додаються у спеціальному розділі системи.

Запит на формування плану харчування (рис. 4.11):

- Частота прийомів їжі: На сторінці «Створення плану» користувачі вказують кількість прийомів їжі певного типу протягом дня.
- Тривалість плану: Обирається період планування (зазвичай тиждень, але можливі інші часові інтервали).

- Створення плану: Після натискання кнопки «Створити план харчування» система аналізує користувацькі дані, враховує встановлені обмеження та генерує рекомендації рецептів, формуючи збалансований план.

Рисунок 4.11 Знімок екрана, що ілюструє поля введення кількості сніданків, обідів, вечерь та перекусів, які користувач бажає отримати

Перегляд та редагування плану харчування показано на рис. 4.12.

За планом у кожен день вказано назви страв та їхній поживний склад:

- Тижневий огляд: Візуалізується сформований план із зазначенням харчової цінності (калорії, білки, вуглеводи, жири).

- Редагування страв: Користувачі можуть замінювати запропоновані рецепти на альтернативні, при цьому система автоматично перераховує показники харчової цінності відповідно до актуальних обмежень.

- Детальна інформація про рецепти: Кожен рецепт супроводжується детальними інструкціями, переліком необхідних інгредієнтів та додатковими застереженнями щодо здоров'я.

Управління сповіщеннями та оновленнями

- Сповіщення: Система інформує користувачів про майбутні прийоми їжі або необхідність поповнення запасів інгредієнтів через електронну пошту або push-сповіщення.

Оновлення профілю: Користувачі можуть регулярно переглядати та коригувати свій профіль та обмеження. Будь-які зміни автоматично враховуються під час наступного оновлення плану харчування.

План харчування на тиждень

Понеділок 1595 kcal Carbs: 28% Protein: 25% Fat: 42%	Вівторок 1620 kcal Carbs: 43% Protein: 15% Fat: 36%	Середа 1800 kcal Carbs: 43% Protein: 21% Fat: 29%
Сніданок Oatmeal Bowl Warm oats with fruits and nuts 350 kcal C: 60 g P: 10 g F: 8 g	Сніданок Pancakes Fluffy pancakes with maple syrup 400 kcal C: 70 g P: 8 g F: 10 g	Сніданок Yogurt Parfait Greek yogurt with berries and granola 300 kcal C: 45 g P: 15 g F: 5 g
Перекуси Apple Slices Fresh apple with cinnamon 95 kcal C: 25 g P: 0 g F: 0 g	Перекуси Mixed Nuts Almonds, walnuts, and cashews 200 kcal C: 6 g P: 5 g F: 18 g	Перекуси Protein Bar Chocolate protein bar 220 kcal C: 20 g P: 15 g F: 7 g
Міксовані горіхи Almonds, walnuts, and cashews 200 kcal C: 6 g P: 5 g F: 18 g	Протеїновий батончик Chocolate protein bar 220 kcal C: 20 g P: 15 g F: 7 g	Хумус та овочі Carrots and celery with hummus 180 kcal C: 15 g P: 5 g F: 10 g
Обід Grilled Chicken Salad Mixed greens with chicken and vinaigrette 450 kcal C: 20 g P: 40 g F: 18 g	Обід Veggie Wrap Whole wheat wrap with veggies and hummus 400 kcal C: 50 g P: 12 g F: 12 g	Обід Tuna Sandwich Tuna salad on whole grain bread 500 kcal C: 40 g P: 35 g F: 20 g
Вечеря Baked Salmon Salmon fillet with lemon and herbs 500 kcal C: 0 g P: 45 g F: 30 g	Вечеря Tofu Stir-Fry Tofu with mixed vegetables 400 kcal C: 30 g P: 20 g F: 18 g	Вечеря Spaghetti Bolognese Pasta with meat sauce 600 kcal C: 75 g P: 25 g F: 15 g
Четвер 1610 kcal Carbs: 46% Protein: 18% Fat: 30%	П'ятниця 1570 kcal Carbs: 42% Protein: 15% Fat: 38%	Субота 1870 kcal Carbs: 31% Protein: 24% Fat: 47%

Рисунок 4.12 Знімок екрана з добре структурованим тижневим планом харчування.

Запропонована мікросервісна архітектура і спеціалізовані моделі рекомендацій надають можливість формувати персоналізовані збалансовані плани харчування для користувачів з урахуванням їх обмежень. Подальші дослідження можуть покращити систему в наступних напрямках:

1. Інтеграція з сервісами ціноутворення та оптимізація вартості меню.

Підключення API торговельних мереж і служб доставки продуктів забезпечить отримання актуальних цін, знижок і наявності товару. Це дозволить вирішувати задачу підбору меню як багатокритеріальну оптимізацію «поживна цінність + вартість + обмеження». Доцільно оцінити як цінові стимули впливають на прихильність користувача до плану та розробити функції втрат, що мінімізують одночасно відхилення від нутрієнтних норм і перевищення бюджету.

2. Генерація меню для групи осіб.

Якщо рішення приймає сім'я, колектив чи дієтолог, то модель профілю слід розширити до множини користувачів і дослідити групові підходи до агрегації переваг та балансування суперечливих обмежень. Перспективними є мультиагентні моделі, де кожен агент-користувач торгується за «свої» страви, а координатор гарантує задоволення групи з урахуванням калорійних норм різного віку.

3. Включення страв із ресторанів та служб доставки.

Імпорт цифрових меню за допомогою OpenMenu чи Menu-XML та автоматичне доповнення відсутніх нутрієнтів NLP- і CV-моделями вимагатиме:

- верифікації харчової цінності, заявленої закладом;
- оцінки впливу кулінарних технік ресторану;
- моделювання компромісу «час/ціна/користь», коли користувач комбінує домашні страви з готовими позиціями.

Такий функціонал підвищить різноманітність меню й адаптацію системи до міського ритму .

Для підтримки перелічених розширень необхідно:

- удосконалити MLOps-конвеєр для безперервного розгортання моделей;
- забезпечити конфіденційність цінових і медичних даних;
- розробити пояснювані інтерфейси, що показують, як ціна та здоров'я вплинули на вибір страв.

Така інтеграція економічних, соціальних і кулінарних факторів перетворить систему на цифрового помічника, здатного підтримувати як одну людину, так і родину у щоденному прийнятті збалансованих та фінансово обґрунтованих рішень щодо харчування.

Висновки до розділу 4

1. Експериментальні дослідження підтвердили гіпотезу щодо підвищення ефективності роботи системи в цілому за рахунок окремого навчання моделей машинного навчання. Спеціалізовані конфігурації SVD, GBDT та Wide&Deep, натреновані окремо для генерації меню різних категорій прийомів їжі продемонстрували приріст Precision@10 від 10 % до 26 % порівняно з універсальною моделлю, яка навчалась на наборі даних, що охоплює всі категорії прийомів їжі.

2. Реалізовано інтерфейс, що надає користувачеві тижневий план харчування з детальним поживним складом, дозволяє редагувати рецепти та надає користувачу список покупок для приготування всіх страв плану харчування.

ВИСНОВКИ

1. На основі системного аналізу сучасних систем рекомендацій щодо харчування людини рішень з'ясовано актуальність розробки інтелектуальної систем планування збалансованого харчування людини на певний період і систем рекомендації рецептів, які зможуть допомогти планувати раціон харчування широкому колу користувачів, що прагнуть самостійно врегулювати особисті дієтичні обмеження, харчові потреби та уподобання.

2. Сформовано модель збалансованого харчування людини з урахуванням дієтичних обмежень, на основі цієї моделі реалізовано додаткову фільтрацію, що відсіває рецепти, які містять інгредієнти або техніки приготування, що підпадають під обмеження користувача.

3. Розроблено метод рекомендації рецептів для системи планування планування збалансованого харчування людини, що використовує методи машинного навчання для формування рецептів страв різних категорій проїму їжі. Ключовим нововведенням цієї роботи є впровадження модульної системи плагінів, що коригує оцінки моделі штучного інтелекту, використовуючи контекстуальні дані та дані сесії користувача

4. Запропоновано концептуальну модель інтелектуальної системи планування збалансованого харчування людини, що підтримує незалежне навчання систем рекомендацій для проїмів їжі різних категорій. реалізація циклічного процесу зберігання, обробки, рекомендації та уточнення планів харчування гарантує адаптивність системи і забезпечує відповідність персоналізованих рекомендацій специфічним вимогам кожного користувача.

5. Експериментально за метрикою Precision@10 обґрунтовано вибір моделі SVD для систем рекомендації рецептів сніданків ($\text{Precision@10} \approx 0,94$) і моделі Wide and Deep для системи рекомендацій рецептів обідів ($\approx 0,96$), перекусів ($\approx 0,98$) і вечір ($\approx 0,95$).

6. Розроблено інтелектуальну систему планування збалансованого харчування що використовує запропонований метод рекомендації і враховує обмеження користувача згідно з моделлю збалансованого харчування людини.

СПИСОК ЛІТЕРАТУРИ

1. Resnick P., Varian H. R. Recommender systems. *Communications of the ACM*. 1997. T. 40, № 3. С. 56–58. DOI: <https://doi.org/10.1145/245108.245121>.
2. Burke R. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*. 2002. Vol. 12, no. 4. P. 331–370. DOI: <https://doi.org/10.1023/a:1021240730564>
3. Adomavicius G., Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*. 2005. Vol. 17, no. 6. P. 734–749. DOI: <https://doi.org/10.1109/tkde.2005.99>
4. Burke R., Felfernig A., Göker M. H. Recommender Systems: An Overview. *AI Magazine*. 2011. Vol. 32, no. 3. P. 13–18. DOI: <https://doi.org/10.1609/aimag.v32i3.2361>
5. de Gemmis M., Lops P., Polignano M. Recommender Systems, Basics of. *Encyclopedia of Social Network Analysis and Mining*. New York, NY, 2018. P. 2125–2137. DOI: https://doi.org/10.1007/978-1-4939-7131-2_110158
6. Felfernig A., Friedrich G., Schmidt-Thieme L. Guest Editors' Introduction: Recommender Systems. *IEEE Intelligent Systems*. 2007. Vol. 22, no. 3. P. 18–21. DOI: <https://doi.org/10.1109/mis.2007.52>
7. Ricci F., Rokach L., Shapira B. Introduction to Recommender Systems Handbook. *Recommender Systems Handbook*. Boston, MA, 2010. P. 1–35. DOI: https://doi.org/10.1007/978-0-387-85820-3_1
8. Vo, Ace and Plachkinova, Miloslava, "Application of Social Matching Recommender Systems in Healthcare" 2022. *SAIS 2022 Proceedings*. 2. URL: <https://aisel.aisnet.org/sais2022/2>
9. Raza, S., Rahman, M., Kamawal, S., Toroghi, A., Raval, A., Navah, F., Kazemeini, A. A Comprehensive Review of Recommender Systems: Transitioning from Theory to Practice. 2024. DOI: 10.48550/arXiv.2407.13699.
10. Ladyzhets, V., Terenchuk, S., Aznaurian, I., Makhinya, A. Improving of intelligent decision support systems for planning a balanced diet, *Eastern-European Journal*

of *Enterprise Technologies*. 2025. Vol. 1, no. 3 (133). P. 37–47. DOI: <https://doi.org/10.15587/1729-4061.2025.322316>

11. Pu L., Faltings B. Understanding and improving relational matrix factorization in recommender systems. *RecSys '13: Seventh ACM Conference on Recommender Systems*, Hong Kong China. New York, NY, USA, 2013. DOI: <https://doi.org/10.1145/2507157.2507178>

12. Dong, L., Jin, R., Jing, G., & Li, Z. On Sampling Top-K Recommendation Evaluation. *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Virtual Event CA USA. New York, NY, USA, 2020. DOI: <https://doi.org/10.1145/3394486.3403262>

13. Papastratis, I., Konstantinidis, D., Daras, P. *et al.* AI nutrition recommendation using a deep generative model and ChatGPT. *Scientific Reports*. 2024. Vol. 14, no. 1. DOI: <https://doi.org/10.1038/s41598-024-65438-x>

14. Stefanidis, K. *et al.* PROTEIN AI Advisor: A Knowledge-Based Recommendation Framework Using Expert-Validated Meals for Healthy Diets. *Nutrients*. 2022. Vol. 14, no. 20. P. 4435. DOI: <https://doi.org/10.3390/nu14204435>

15. Zioutos K., Kondylakis H., Stefanidis K. Healthy Personalized Recipe Recommendations for Weekly Meal Planning. *Computers*. 2023. Vol. 13, no. 1. P. 1. DOI: <https://doi.org/10.3390/computers13010001>

16. Majumder, B. P., Li, S., Ni, J., & McAuley, J. Generating Personalized Recipes from Historical User Preferences. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China. Stroudsburg, PA, USA, 2019. DOI: <https://doi.org/10.18653/v1/d19-1613>

17. Mohan A., Singh S., Kalpana D. A. V. Meal Plan Monitoring and Recommendation System. *International Conference on Recent Trends in Data Science and its Applications (ICRTDA 2023)*, Kattankulathur, Tamil Nadu, India, 30–31 March 2023. Denmark, 2023. DOI: <https://doi.org/10.13052/rp-9788770040723.117>

18. Zhang, S., Yao, L., Sun, A., & Tay, Y. Deep Learning Based Recommender System. *ACM Computing Surveys*. 2019. Vol. 52, no. 1. P. 1–38. DOI: <https://doi.org/10.1145/3285029>
19. Viktor L., Yeremenko B., Terenchuk S. Candidate Generation for Meal Recommendation System. *2024 IEEE 4th International Conference on Smart Information Systems and Technologies (SIST)*, Astana, Kazakhstan, 15–17 May 2024. 2024. P. 560–564. DOI: <https://doi.org/10.1109/sist61555.2024.10629517>
20. Journal I. Meal Map Pro. *Interantional journal of scientific research in engineering and management*. 2023. Vol. 07, no. 11. P. 1–11. DOI: <https://doi.org/10.55041/ijssrem27174>
21. Elswailer D., Harvey M. Towards Automatic Meal Plan Recommendations for Balanced Nutrition. *RecSys '15: Ninth ACM Conference on Recommender Systems*, Vienna Austria. New York, NY, USA, 2015. DOI: <https://doi.org/10.1145/2792838.2799665>
22. Liu C., Jin H., Wang J. Investigating the Moderating Effects of Context-Aware Recommendations on the Relationship Between Knowledge Search and Decision Quality. *Journal of Organizational and End User Computing*. 2024. Vol. 36, no. 1. P. 1—1. DOI: <https://doi.org/10.4018/joeuc.345930>
23. Gao, X., Feng, F., Huang, H., Mao, X.-L., Lan, T., Chi, Z. Food recommendation with graph convolutional network. *Information Sciences*. 2022. Vol. 584. P. 170–183. DOI: <https://doi.org/10.1016/j.ins.2021.10.040>
24. Ладижець В. І., Теренчук С. А. Моделі, методи і засоби планування раціону харчування людини. Управління розвитком складних систем. Київ, 2023. № 53. С. 39 – 44, DOI: <https://doi.org/10.32347/2412-9933.2023.53.39-44>
25. Naik, P. A. Intelligent food recommendation system using machine learning. *International Journal of Innovative Science and Research Technology*, 2020, Vol. 5 no. 8, P. 616–619. DOI: <https://doi.org/10.38124/IJISRT20AUG414>
26. Oh Y., Choi A., Woo W. u-BabSang: a context-aware food recommendation system. *The Journal of Supercomputing*. 2009. Vol. 54, no. 1. P. 61–81. DOI: <https://doi.org/10.1007/s11227-009-0314-5>

27. Meng L., Feng F., He X., Gao X., Chua T.-S. Heterogeneous Fusion of Semantic and Collaborative Information for Visually-Aware Food Recommendation. *MM '20: The 28th ACM International Conference on Multimedia*, Seattle WA USA. New York, NY, USA, 2020. DOI: <https://doi.org/10.1145/3394171.3413598>
28. Eurostat. Overweight and obesity – BMI statistics URL: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Overweight_and_obesity_-_BMI_statistics
29. Misra, A., Jayawardena, R., & Anoop, S. Obesity in South Asia: Phenotype, Morbidities, and Mitigation. *Current Obesity Reports*, 2019 Vol. 8 no. 1, p. 43–52. DOI: <https://doi.org/10.1007/s13679-019-0328-0>
30. Powell-Wiley, T. M., Poirier, P., Burke, L. E., Després, J.-P., Gordon-Larsen, P., Lavie, C. J., Lear, S. A., Ndumele, C. E., Neeland, I. J., Sanders, P., & St-Onge, M.-P. (). Obesity and Cardiovascular Disease: A Scientific Statement From the American Heart Association. *Circulation*, 2021 Vol. 143 no. 21. DOI: <https://doi.org/10.1161/cir.0000000000000973>
31. World Health Organization. (2020, April 29). *Healthy diet* [Fact sheet]. URL: <https://www.who.int/news-room/fact-sheets/detail/healthy-diet>
32. Sharifi, M. H., Izadpanah, P., Hosseini, M. M., & Vojoudi, M. (). *Relationship between dietary variety, adequacy, moderation, and balanced diet and cardiovascular risk factors*. *BMC Nutrition*, 2022, Vol. 8 no. 1, Article 20. DOI: <https://doi.org/10.1186/s40795-022-00514-x>
33. Karagun, B., & Baklaci, N. Comparative analysis of basal metabolic rate measurement methods in overweight and obese individuals: A retrospective study. *Medicine*, 2024, August 30 Vol. 103 no. 35, e39542. DOI: <https://doi.org/10.1097/MD.00000000000039542>
34. L. M. Redman et al. Energy requirements in nonobese men and women: results from CALERIE, *The American Journal of Clinical Nutrition*. 2013. Vol. 99, no. 1. P. 71–78. DOI: <https://doi.org/10.3945/ajcn.113.065631>
35. Centre for Food Safety. (n.d.). Eating a healthy diet [Multimedia publication FSF29_02]. The Government of the Hong Kong Special Administrative Region. URL:

https://www.cfs.gov.hk/english/multimedia/multimedia_pub/multimedia_pub_fsf_29_02.html

36. Hestaras, M. (n.d.). The 10 most common dietary restrictions. *Healthline*. URL: <https://www.healthline.com/nutrition/most-common-dietary-restrictions>
37. Ладижець, В., & Теренчук, С. Проектування методу рекомендації для інформаційної системи планування харчування. *Шляхи підвищення ефективності будівництва*, №53 (2), С. 377–388. DOI: [https://doi.org/10.32347/2707-501x.2024.53\(2\).377-388](https://doi.org/10.32347/2707-501x.2024.53(2).377-388)
38. Amiri M., Sarani Rad F., Li J. Delighting Palates with AI: Reinforcement Learning's Triumph in Crafting Personalized Meal Plans with High User Acceptance. *Nutrients*. 2024. Vol. 16, no. 3. P. 346. DOI: <https://doi.org/10.3390/nu16030346>
39. Abdul Hussien F. T., Rahma A. M. S., Abdul Wahab H. B. Recommendation Systems For E-commerce Systems An Overview. *Journal of Physics: Conference Series*. 2021. Vol. 1897, no. 1. P. 012024. DOI: <https://doi.org/10.1088/1742-6596/1897/1/012024>
40. Deldjoo, Y., Schedl, M., Cremonesi, P., & Pasi, G. Recommender Systems Leveraging Multimedia Content. *ACM Computing Surveys*. 2020. Vol. 53, no. 5. P. 1–38. DOI: <https://doi.org/10.1145/3407190>
41. Nakamura, M. T., & Alfouzan, N. W. Reduced food cravings correlated with a 24-month period of weight loss and weight maintenance. *Physiology & Behavior*. Advance online publication, 2025, DOI: <https://doi.org/10.1016/j.physbeh.2025.114813>
42. Rajarajeswari, S., Naik, S., Srikant, S., Sai Prakash, M.K., Uday, P. Movie Recommendation System. *Emerging Research in Computing, Information, Communication and Applications. Advances in Intelligent Systems and Computing*, 2019, vol 882. Springer, Singapore. DOI: https://doi.org/10.1007/978-981-13-5953-8_28
43. Hu, J., & Zhang, B. Product Recommendation System (CS224W project report). 2012, December 10, Stanford Network Analysis Project.
44. Papadakis, H., Papagrigoriou, A., Panagiotakis, C., Kosmas, E., & Fragopoulou, P. Collaborative filtering recommender systems taxonomy. *Knowledge and Information Systems*, 2022, Vol. 64, no. 1, P. 35–74. DOI: <https://doi.org/10.1007/s10115-021-01628-7>

45. Ekstrand, M. D., Riedl, J. T., & Konstan, J. A. 2011. Collaborative filtering recommender systems. *Foundations and Trends® in Human–Computer Interaction*, Vol. 4, no. 2, P. 81–173. DOI: <https://doi.org/10.1561/11000000009>
46. Sreeja, Y., & Sheela, J. J. J. Dietary analysis among diabetic patients using gradient boosting over decision tree. *Proceedings of the 2024 15th International Conference on Communications (COMM)*, 2024, P. 1–4. DOI: <https://doi.org/10.1109/COMM62355.2024.10741449>
47. Godbole, A., Shukla, G., Sharma, Y., Joshi, H., & Shaikh, S., AI powered diet planning application using XGBoost, visual nutritional analysis, and recipe recommendation. 2024, *Zenodo*. DOI: <https://doi.org/10.5281/zenodo.15490512>
48. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, 2017, Vol. 30, pp. 3146–3154
49. Maji, S., Maity, S., Das, J., & Majumder, S. *An improved recommendation system based on neural matrix factorization*. In *Proceedings of the 2024 4th International Conference on Intelligent Technologies (CONIT)* 2024, June, DOI: <https://doi.org/10.1109/CONIT61985.2024.10627601>
50. Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., & Shah, H., Wide & Deep learning for recommender systems. *Proceedings of the 2016 ACM Conference on Knowledge Discovery and Data Mining*, p. 2826–2834. DOI: <https://doi.org/10.1145/2988450.2988454>
51. Abdi, H., Singular value decomposition (SVD) and generalized singular value decomposition. *Encyclopedia of measurement and statistics*, 2007, Vol. 907(912), P. 44.
52. Singh, R. H., Maurya, S., Tripathi, T., Narula, T., & Srivastav, G. Movie recommendation system using cosine similarity and KNN. *International Journal of Engineering and Advanced Technology*, 2020, Vol. 9 no. 5, p. 556-559. DOI: <https://doi.org/10.35940/ijeat.e9666.069520>
53. Luo, X., Zhou, M., Xia, Y., & Zhu, Q., An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE*

Transactions on Industrial Informatics, 2014, Vol. 10, no. 2, P. 1273–1284. DOI: <https://doi.org/10.1109/tii.2014.2308433>

54. H. Khatter et al., Movie Recommendation System using Cosine Similarity with Sentiment Analysis, *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2–4 September 2021. 2021. DOI: <https://doi.org/10.1109/icirca51532.2021.9544794>

55. Zheng, S., Ding, C., & Nie, F., Regularized singular value decomposition and application to recommender system. 2018 DOI: <https://doi.org/10.48550/arXiv.1804.05090>

56. Li, T., Wang, J., Chen, H., Feng, X., & Ye, F. A NMF-based Collaborative Filtering Recommendation Algorithm, *2006 6th World Congress on Intelligent Control and Automation*, Dalian, China. 2006. DOI: <https://doi.org/10.1109/wcica.2006.1714249>

57. Lord W. M., Sun J., Bollt E. M. Geometric k-nearest neighbor estimation of entropy and mutual information. *Chaos: An Interdisciplinary Journal of Nonlinear Science*. 2018. Vol. 28, no. 3. P. 033114. DOI: <https://doi.org/10.1063/1.5011683> .

58. L. Breiman et al., Classification And Regression Trees, *Routledge*, 2017. DOI: <https://doi.org/10.1201/9781315139470>

59. Breiman, L., Random forests. *Machine Learning*, 2001 Vol. 45 no. 1, p. 5–32. DOI: <https://doi.org/10.1023/A:1010933404324>

60. Friedman, J. H., Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 2001, Vol. 29 no. 5, p. 1189–1232. DOI: <https://doi.org/10.1214/aos/1013203451>

61. Chen, W., Cai, F., Chen, H., & Rijke, M. D., Joint Neural Collaborative Filtering for Recommender Systems, *ACM Transactions on Information Systems*. 2019. Vol. 37, no. 4. P. 1–30. DOI: <https://doi.org/10.1145/3343117>

62. Dai H., Zhu M., Gui X., An Improved NCF Model in Federated Recommendation Systems. *2023 China Automation Congress (CAC)*, Chongqing, China, 17–19 November 2023. DOI: <https://doi.org/10.1109/cac59555.2023.10450566>

63. Popescu, M. C., Balas, V. E., Perescu-Popescu, L., & Mastorakis, N., Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems*, 2009, Vol. 8(7), p. 579-588.

64. Horn, R. A., The hadamard product. In *Proc. symp. appl. math* 1990 May, Vol. 40, pp. 87-169.
65. Yu, Y., Wang, W., Feng, Z., & Xue, D. (2021). A dual augmented two-tower model for online large-scale recommendation. *DLP-KDD*.
66. Chen, R., Liu, B., Zhu, H., Wang, Y., Li, Q., Ma, B., ... & Zheng, B., Approximate nearest neighbor search under neural similarity metric for large-scale recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* 2022, pp. 3013-3022, DOI: <https://doi.org/10.1145/3511808.3557098>
67. Lin H., Ji H. Improved Wide&Deep Recommendation Algorithm for User Behavioural Characteristics. *2024 13th International Conference of Information and Communication Technology (ICTech)*, Xiamen, China, 12–14 April 2024. P. 234–238. DOI: <https://doi.org/10.1109/icttech63197.2024.00052>
68. Guo, H., Tang, R., Ye, Y., Li, Z., & He, X., DeepFM: An End-to-End Wide & Deep Learning Architecture for CTR Prediction, *Information Retrieval*, 2017, DOI: <https://doi.org/10.48550/arXiv.1804.04950>
69. Xia, L., Xu, J., Lan, Y., Guo, J., & Cheng, X. Learning Maximal Marginal Relevance Model via Directly Optimizing Diversity Evaluation Measures, *SIGIR '15: The 38th International ACM SIGIR conference on research and development in Information Retrieval*, Santiago Chile. New York, NY, USA, 2015. DOI: <https://doi.org/10.1145/2766462.2767710>
70. CHURCH K. W., Word2Vec. *Natural Language Engineering*. 2016. Vol. 23, no. 1. P. 155–162. DOI: <https://doi.org/10.1017/s1351324916000334>
71. Verbitski, A., Gupta, A., Saha, S., et al. Amazon Aurora, *SIGMOD/PODS'17: International Conference on Management of Data*, Chicago Illinois USA. New York, NY, USA, 2017. DOI: <https://doi.org/10.1145/3035918.3056101>
72. DeCandia, G., Hastorun, D., Jampani, M., et al., Dynamo: Amazon's highly available key-value store. In *Proceedings of Twenty-First ACM SIGOPS Symposium on Operating Systems Principles (SOSP '07)*, 2007, P. 205–220 DOI: <https://doi.org/10.1145/1294261.1294281>

73. M. R. Palankar et al., Amazon S3 for science grids, *The 2008 international workshop*, Boston, MA, USA, 24 June 2008. New York, New York, USA, 2008. DOI: <https://doi.org/10.1145/1383519.1383526>
74. Amazon Web Services. (2023). *Amazon SageMaker Developer Guide*. AWS. URL: <https://docs.aws.amazon.com/sagemaker/latest/dg/whatis.html>
75. Amazon Web Services. (2023). *Amazon SageMaker Feature Store documentation*. AWS. URL: <https://docs.aws.amazon.com/sagemaker/latest/dg/feature-store.html>
76. Haerder T., Reuter A. Principles of transaction-oriented database recovery. *ACM Computing Surveys*. 1983. Vol. 15, no. 4. P. 287–317. DOI: <https://doi.org/10.1145/289.291>
77. Amazon Web Services. (2023). *Amazon API Gateway Developer Guide*. AWS. URL: <https://docs.aws.amazon.com/apigateway>
78. Spillner, J., Mateos, C., & Monge, D. A., FaaSification: From monolithic to serverless applications. In *2017 IEEE International Conference on Software Architecture Workshops*, 2019, pp. 284–293, DOI: <https://doi.org/10.1109/ICSAW.2017.65>
79. Freeman, A., Pro ASP.NET Core 7: Develop cloud-ready web applications using MVC, Blazor, and Razor Pages (10th ed.). Apress. 2023, DOI: <https://doi.org/10.1007/978-1-4842-8574-2>
80. Price, J., & Freeman, A. (2023). *Beginning Visual C# and .NET: Learn to program with C# and the .NET platform* (2nd ed.). Apress.
81. Miller, J. (2022). *AutoMapper documentation*. GitHub. URL: <https://github.com/AutoMapper/AutoMapper>
82. Newton-King, J., & Stegman, M. (2023). Polly: .NET resilience and transient-fault-handling library [Software]. GitHub. URL: <https://github.com/App-vNext/Polly>
83. Harris, C. R., Millman, K. J., van der Walt, S. J., et al., Array programming with NumPy. *Nature*, 2020, Vol. 585 no. 7825, P. 357–362. DOI: <https://doi.org/10.1038/s41586-020-2649-2>

84. McKinney, W. Data structures for statistical computing in Python. *Proceedings of the 9th Python in Science Conference* 2010, P. 51–56, DOI: <https://doi.org/10.25080/Majora-92bf1922-00a>
85. Pedregosa, F., Varoquaux, G., Gramfort, A., et al., Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 2011, Vol. 12, P. 2825–2830. URL: <http://jmlr.org/papers/v12/pedregosa11a.html>
86. Abadi, M., Barham, P., Chen, J., et al., TensorFlow: A system for large-scale machine learning. *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 2016)* P. 265–283, URL: <https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi>
87. Paszke, A., Gross, S., Massa, F., et al., PyTorch: An imperative style, high-performance deep learning library, *Advances in Neural Information Processing Systems (NeurIPS 2019)*, 2019, Vol. 32, P. 8024–8035. URL: https://papers.nips.cc/paper_files/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html
88. Xie, Y., Cruz, L., Heck, P., & Rellermeier, J. S., Systematic mapping study on the machine learning lifecycle. *2021 IEEE/ACM 1st workshop on AI engineering-software engineering for AI (WAIN)* 2021, May P. 70-73, DOI: <https://doi.org/10.1109/wain52551.2021.00017>
89. Amazon Web Services. Build an end-to-end MLOps pipeline using Amazon SageMaker Pipelines, GitHub, and GitHub Actions. *AWS Machine Learning Blog*. 2021, February 18, URL: <https://aws.amazon.com/blogs/machine-learning/build-an-end-to-end-mlops-pipeline-using-amazon-sagemaker-pipelines-github-and-github-actions/>
90. Choudhary, P., Do's and don'ts for effective MLOps: Machine learning and data engineering. *Level Up (via Medium)*. 2021, October 4, URL: <https://levelup.gitconnected.com/dos-and-don-ts-for-effective-mlops-machine-learning-and-data-engineering-34f1b6a2fa6a>
91. Кравченко, М. (2024). Гібридний зворотній зв'язок в циклі мішення [Hybrid feedback in the milling cycle] (кандидатська дисертація, Київський політехнічний інститут імені Ігоря Сікорського). Архів КПІ.

92. Ramaswamy, A., Recommendations for the AI cold start problem. *Airbyte Blog*. 2024, February 12, URL: <https://airbyte.com/blog/recommendations-for-the-ai-cold-start-problem>
93. Yuan H., Hernandez A. A. User Cold Start Problem in Recommendation Systems: A Systematic Review. *IEEE Access*. 2023. P. 36958-136977. DOI: <https://doi.org/10.1109/access.2023.3338705>
94. Bianchi, M., Cesaro, F., Ciceri, F., Dagrada, M., Gasparin, A., Grattarola, D., ... & Cella, L. Content-Based approaches for Cold-Start Job Recommendations, *Recommender Systems Challenge 2017*, Como, Italy, 27 August 2017. New York, USA, 2017. DOI: <https://doi.org/10.1145/3124791.3124793>
95. Miao, Z., Yan, J., Chen, K., Yang, X., Zha, H., & Zhang, W., Joint prediction of rating and popularity for cold-start item by sentinel user selection. *IEEE Access*, 2016, 4, P. 8500-8513, DOI: <https://doi.org/10.1109/access.2016.2633282>
96. Vairachilai S. Collaborative Filtering Recommender System: Comparative Survey on Cold-Start Issue. *Indian Journal of Science and Technology*. 2018. Vol. 11, no. 22. P. 1–13. DOI: <https://doi.org/10.17485/ijst/2018/v11i20/116392>
97. Majumder, B. P., Li, S., Ni, J., & McAuley, J. Generating Personalized Recipes from Historical User Preferences, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China. Stroudsburg, PA, USA, 2019. DOI: <https://doi.org/10.18653/v1/d19-1613>
98. Bisong, E. Google Colaboratory. In Building Machine Learning and Deep Learning Models on Google Cloud Platform 2019, P. 59–64, *Apress*. DOI: https://doi.org/10.1007/978-1-4842-4470-1_7
99. Hug, N. Surprise: A Python library for recommender systems. 2017, URL: <http://surpriselib.com>
100. Kluyver Thomas et al., Jupyter Notebooks – a publishing format for reproducible computational workflows, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*. 2016. DOI: <https://doi.org/10.3233/978-1-61499-649-1-87>
101. Food.com - Recipes, Food Ideas and Videos. URL: <https://www.food.com/>

102. Majumder, B. P., Li, S., Ni, J., & McAuley, J. Generating Personalized Recipes from Historical User Preferences, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China. Stroudsburg, PA, USA, 2019. DOI: <https://doi.org/10.18653/v1/d19-1613>
103. Bisong, E. Google Colaboratory. *In Building Machine Learning and Deep Learning Models on Google Cloud Platform* 2019, P. 59–64, *Apress*. DOI: https://doi.org/10.1007/978-1-4842-4470-1_7
104. Hug, N. Surprise: A Python library for recommender systems. 2017, URL: <http://surpriselib.com>
105. Kluyver Thomas et al., Jupyter Notebooks – a publishing format for reproducible computational workflows, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*. 2016. DOI: <https://doi.org/10.3233/978-1-61499-649-1-87>

ДОДАТКИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

ФАКУЛЬТЕТ АВТОМАТИЗАЦІЇ І ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

«07» квітня 2025 р.

ДОВІДКА

про впровадження результатів дисертаційного дослідження **ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПЛАНУВАННЯ ЗБАЛАНСОВАНОГО ХАРЧУВАННЯ ЛЮДИНИ**

Запропонована в дисертаційній роботі Ладижця В.І. інтелектуальна система поєднує високу персоналізацію та адаптивність до особистих вподобань і обмежень користувача. Ця робота є прикладом моделювання інформаційних процесів і систем різного призначення та їх реалізації з використанням машинного навчання, глибокого навчання і штучних нейронних мереж.

Впровадження в навчальний процес досліджених в роботі моделей, методів і технологій побудови архітектури інформаційних систем надає можливість здобувачам освіти галузі знань 12-«Інформаційні технології» отримати досвід розробки систем різного призначення. Це підтверджує актуальність, теоретичну і практичну значимість роботи в процесі підготовки майбутніх спеціалістів галузі.

декан факультету АІТ,
доктор технічних наук, професор



Олександр ТЕРЕНТЬЄВ



УКРАЇНА

ШЕВЧЕНКІВСЬКА РАЙОННА В МІСТІ КИСВІ
ДЕРЖАВНА АДМІНІСТРАЦІЯ
УПРАВЛІННЯ ОСВІТИ

вул. Зоологічна, 6-А, м. Київ, 04119, тел. (044) 489-02-50, тел./факс 489-02-50
E-mail: priyom.shevrno@kyivcity.gov.ua, код ЄДРПОУ 37470086

10.06.2025 № 109/33/08-1249/1

на № _____ від _____

ДОВІДКА

про апробацію результатів дисертаційного дослідження
«ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПЛАНУВАННЯ ЗБАЛАНСОВАНОГО
ХАРЧУВАННЯ»
ЛАДИЖЦЯ ВІКТОРА ІВАНОВИЧА

Цією довідкою засвідчується направленість дисертаційного дослідження «Інтелектуальна система планування збалансованого харчування» на реалізацію завдань Стратегії реформування системи шкільного харчування на період до 2027 року, схваленої розпорядженням Кабінету Міністрів України від 27 жовтня 2023 р. № 990-р, в частині запровадження доступних та ефективних інтелектуальних систем, які дозволяють користувачам формувати плани збалансованого, якісного, безпечного та повноцінного харчування, зокрема харчування дітей у закладах дошкільної та загальної середньої освіти, з урахуванням норм, індивідуальних особливостей, стану здоров'я та харчових уподобань.

Апробація матеріалів дисертаційного дослідження підтвердила здатність запропонованої автором інтелектуальної системи планування збалансованого харчування забезпечувати науково-обґрунтовану підтримку прийняття рішень у сфері вибору оптимального раціону харчування, зокрема шкільного харчування. Система дозволяє автоматично формувати плани харчування, які відповідають сучасним нормам і рекомендаціям щодо безпечного та повноцінного харчування, адаптуючи їх під потреби користувачів.

Запропоновані автором моделі штучного інтелекту, методи та алгоритми створюють необхідні передумови для практичного впровадження автоматизованих засобів контролю та підтримки здорового способу життя, що особливо актуально для забезпечення фізичного здоров'я дітей і підлітків в умовах повномасштабної війни. Це сприятиме поширенню культури здорового харчування серед молодого покоління та покращенню загального рівня здоров'я суспільства.

Наукові дослідження, виконані Ладижцем В. І. у рамках дисертаційного дослідження, є актуальними, своєчасними та економічно доцільними.

Начальник

Євгенія ЯРОВА

