УДК 004.415.2
DOI https://doi.org/10.32782/2663-5941/2023.1/19

Олещенко Л.М.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Вернік М.О.
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ПРИНЦИПИ РОЗРОБЛЕННЯ РЕКОМЕНДАЦІЙНИХ СИСТЕМ З ВИКОРИСТАННЯМ МЕТАЕВРЕСТИЧНОЇ ОПТИМІЗАЦІЇ

Більшість метаевристичних алгоритмів оптимізації засновані на ройовій поведінці тварин або біологічних взаємодіях живих організмів. Саме це надає метаевристичним алгоритмам простоту програмної реалізації, яка базується на ініціалізації початкової генерації, формування вектора, до якого застосовується цільова функція та створюється критерій зупинки. Перевага використання запропонованого підходу для розроблення рекомендаційних систем полягає в тому, що з часом дослідник розробляє кращі та досконаліші методи оптимізації, при цьому сама база буде оновлюватися шляхом додавання нових алгоритмів і, відповідно, не відбуватиметься процес стагнації системи рекомендацій. У даному дослідженні розглянуто алгоритми, які дозволяють точніше створювати рекомендаційні системи, а отже, й покращувати конверсію користувачів та їхню взаємодію з системою на основі отриманих даних, що дозволяє покращити точність аналізу даних для отримання цільової множини користувачів системи. Запропонований метод створення рекомендаційної системи з використанням методів метаевристичної оптимізації забезпечує більшу точність у порівнянні з класичною реалізацією у середньому на 4-6%. Модульна структура програмного забезпечення, модулі якої реалізують зазначені алгоритми метаевристичної оптимізації, полегшує реалізацію запропонованого програмного методу в цілому. Для апробації розробленої методики була створена клієнтська частина програмного забезпечення у вигляді мобільного додатку. Використано технології програмування для клієнтської частини програмного забезпечення запропонованої системи рекомендацій: Flutter, Firebase API, REST API, React JS. Для розробки серверної частини програмного забезпечення використані технології: .NET Core Web API, AZURE MS SQL, GOOGLE MAPS API, ASP.NET MVC, Python, Tensorflow, MOA, PyTorch, PySwarm. База даних запропонованої системи складається з 45 таблиць і займає ~500 Мб на жорсткому диску серверів US-1 у хмарній системі Azure. Кожен вибір, зроблений користувачем, зберігається в системі, що забезпечує процес навчання для нейронної мережі, тому система може надавати більш точні рекомендації, а також пропонувати людей відповідно з подібним психологічним профілем. Тестування методу проходило на вибірці даних для міста Київ (Україна) в кількості 6000 подій, які були отримані за допомогою сервісів Google Maps API.

Ключові слова: *MOA, система рекомендацій, нейронні мережі, матриця факторизацій, зворотнє поширення, вектор багатошарового персептрону, програмне забезпечення.*

Постановка проблеми. Метаевристичні методи оптимізації в нейронних мережах дають точніші результати в межах відносної похибки на 10% або на 1–2%, ніж класичні методи. В інтернет-магазинах використовується індивідуальний алгоритм надання рекомендацій для клієнта. У звичайному магазині особисті рекомендації надаються консультантами для зовнішньої оцінки потреб клієнтів, на основі яких вони пропонують рекомендації. Незважаючи на це, більшість людей роблять покупки через взаємодію з людиною та її емоційне забарвлення, а не через саму рекомендацію. Таким чином, актуальною є розробка програмного забезпечення системи рекомендацій, спрямованої на формування рекомендацій на основі нового методу оптимізації з урахуванням психологічної складової користувача.

Метою дослідження в цій статті є аналіз використання метаевристичних методів оптимізації для побудови системи рекомендацій у сфері відпочинку з побудовою психологічного портрета користувача за допомогою нейронної мережі та матриці факторизації.

Аналіз існуючих програмних рішень. Одним із підходів до проектування рекомендаційних систем (РС), який широко використовується,

є колаборативна фільтрація [1]. Знаходячи користувачів того самого віку або елементи з історією оцінок, подібною до поточного користувача або елемента, генеруються рекомендації з використанням цього сусідства. Методи фільтрації на основі вмісту використовують набір окремих атрибутів і ознак для характеристики елемента в системі. Широко використовуваним алгоритмом є представлення векторного простору, також використовуються методи, що включають аналіз тексту, пошук інформації, мультимодальний аналіз настрою та глибоке навчання.

Системи рекомендацій на основі сеансу використовують взаємодію користувача під час сеансу. Такі системи використовуються на платформах YouTube та Amazon. Вони особливо корисні, коли історія користувача (минулі кліки та покупки) недоступна або не відповідає поточному сеансу користувача. Методи створення рекомендацій на основі сеансів переважно використовують генеративні послідовні моделі. Недоліками таких методів є обчислювальна складність обробки великих даних і низька точність існування розрідженості. Методи рекомендацій, засновані на навчанні з підкріпленням, дозволяють тренувати моделі та оптимізувати на основі показників залученості та зацікавленості користувачів рекомендаційні системи.

Мобільні дані для мобільних систем рекомендацій є більш складними, вони вимагають просторової та часової автокореляції та мають проблеми перевірки. Факторами, які можуть вплинути на мобільні рекомендаційні системи та точність результатів прогнозування, є контекст, метод рекомендації та конфіденційність. Uber і Lyft використовують цей підхід для створення маршрутів руху для водіїв таксі в місті, використовуючи дані GPS про маршрути, якими їдуть водії таксі під час роботи, які включають місцезнаходження (широту і довготу), позначки часу та робочий статус. Ці дані використовуються, щоб рекомендувати список пунктів посадки на маршруті для оптимізації часу завантаження та збільшення прибутку. Багато систем рекомендацій використовують гібридний підхід, використовуючи різні типи фільтрації. Прикладом використання гібридних рекомендаційних систем є Netflix. Деякі методи гібридизації включають:

1. Зважені методи: чисельна комбінація оцінок різних компонентів рекомендацій.

2. Перемикання: вибір компонентів рекомендації та застосування вибраного.

3. Змішані методи: рекомендації з різних настанов представлені разом, щоб дати рекомендацію.

На основі проведених досліджень виділено наступні переваги та недоліки розглянутих методів:

1. Персоналізовану РС складніше реалізувати порівняно з класичною РС, але в той же час персоналізована РС забезпечує більш точні результати, ніж класична РС.

2. Загальна тенденція використання РС – гібридні моделі.

3. Найбільша концентрація розробки на системах з усвідомленням ризиків.

У попередніх дослідженнях розглядалась класична модель побудови психологічного портрета користувача з використанням комбінації багатошарового персептрона і зворотного поширення [2-6].

Аналіз даних користувачів рекомендаційної системи

Дані користувача та події у даній роботі представлені як окремі вектори даних, де кількість активних користувачів системи становить 110, а кількість подій – 5370. Всього користувачами додано більше 2000 подій. Також понад 10 000 даних про дії користувачів, такі, як час перебування на певній сторінці, запрошені люди, параметри відбору, психологічний портрет. Таким чином, на виході ми маємо понад 12 000 рядків даних про користувачів та їх взаємодію в системі. Цей набір даних можна представити у вигляді 2 наборів даних для навчання та тестування, де 10 000 рядків є навчальними, а 2000 рядків – тестовими. Основою рекомендацій є дані користувача, рекомендаціями можуть бути будь-які характеристики (наприклад, побудована нейронна мережа психологічного портрета користувача для створення рекомендованого списку запрошень користувачеві). Запропонована система рекомендацій використовує модель нейроколаборативного фільтру з модифікацією та оптимізацією процесу навчання нейронної мережі за допомогою 5 метаевристичних алгоритмів оптимізації: Gray Wolf Optimization (GWO), Particle Swarm Optimization (PSO), Whale Optimization Algorithm (WOA), Bee Algorithm (BA), Genetic Algorithm (GA), які взаємодіють з нейронною мережею наступним чином:

1. Нейронна мережа розкладається на одновимірний вектор важелів нейронних зв'язків і важелів відхилень.

2. Вектор розкладеної нейронної мережі буде поколінням.

3. Кожен із 5 алгоритмів генеруватиме нове покоління.

4. Серед 5 поколінь вибирається найкраще покоління на основі середньоквадратичної помилки кінцевого результату.

5. Актуалізуються важелі нейронних зв'язків і важелі відхилень.

Система на вході отримує 2 вектори – «Users» та «Events», де вектор користувачів – дані психологічних портретів, а вектор подій – дані параметрів опису подій, відповідно, кожен із яких створює по 2 вектори:

1. MF Users (вектор матриці факторизації користувачів);

2. MF Events (вектор матриці факторизації подій);

3. MLP Users (вектор багатошарового персептрону користувачів);

4. MLP Events (вектор багатошарового персептрону подій).

Після обробки та розкладання векторів матрицю факторизації (MF) сформують вектори «MF Users» та «MF Events», у той же час в багатошаровий персептрон (MLP) використаються об'єднані вектори «MLP Users» та «MLP Events». На етапі отримання результатів як з матриці факторизації, так і з MLP, вони об'єднуються у вихідний результат, який буде відправлено в множину методів метаевристичної оптимізації, з яких буде вибрано найкраще покоління для отримання генерації нового покоління (оновлення вагів нейронної мережі), а також знаходження помилки відносно заздалегіть відомого результату вибору. Послідовність реалізації загального алгоритму побудови запропонованої РС наступна:

1. Побудова «дерева структури подій».

2. Введення даних користувачів системи.

3. Застосування метаевристичних алгоритмів оптимізації.

4. Персоналізована рекомендаційна система з підбором подій та людей.

На рис. 1 зображено приклад скороченої структури подій в рекреаційній сфері.

Застосування метаевристичних алгоритмів оптимізації

Алгоритм Whale Optimization

Метаевристичний алгоритм оптимізації Whale Optimization Algorithm (WOA) імітує мисливську поведінку горбатих китів та полягає у моделюванні мисливської поведінки з випадковим або кращим пошуковим агентом для переслідування здобичі та використанням спіралі для моделювання пухирчастої сітки. Експлуатаційна модель [5]:

$$D_{-} = \left| C_{-}.X_{p-}(t) - X_{-}(t) \right|, \quad (1)$$

$$X_{-}(t+1) = \{ \vec{X}(t) - \vec{A}.\vec{D},\ p < 0.5 \quad (2)$$

$$\overrightarrow{D} * e^{bt} cos\cos(2\pi t) + \overrightarrow{X^*}(t),\ p \geq 0.5 \quad (3)$$

Пошукова модель [5]:

$$D_{-} = \left| C_{-}.X_{rand-} - X_{-} \right|, \quad (4)$$

$$X_{-}(t+1)_{-} = \left| C_{-}.X_{rand-} - A_{-}.D_{-} \right| \quad (5)$$

Алгоритм Genetic Algorithm

Генетичні алгоритми засновані на принципі природного відбору. Генетичні (або еволюційні) методи реалізують процес, пов'язаний з біологічною еволюцією, включають генерацію, рекомбінацію та мутацію хромосом, створення нової популяції з кращими генами та покращеною пристосованістю. Змінні моделі відповідають хромосомам популяції і намагаються через деяку кількість поколінь отримати найкращі хромосоми, що представляють оптимальне рішення проблеми.

Алгоритм Particle Swarm Optimization

Оптимізація зграї частинок базується на поведінці птахів, які збираються навколо джерел їжі. Коли один птах відчуває запах їжі, він голосно цвірінькає, а інші птахи розгойдуються в його напрямку. У контексті оптимізації розташування їжі вказує на вектор моделі для визначення оптимального значення цільової функції [3]:

$$X^{i}_{-}(t+1) = X^{i}_{-}(t) + V^{i}_{-}(t+1), \quad (6)$$

$$\overrightarrow{V_k^i}(t+1) = w * \overrightarrow{V_k^i}(t) + c_1 * r_2 \left(\overrightarrow{p^i_{k,best}} - \overrightarrow{p_k^i}(t) \right) + + c_2 * r_2 \left(\overrightarrow{g^i_{best}}(t) - \overrightarrow{p_k^i}(t) \right) \quad (7)$$

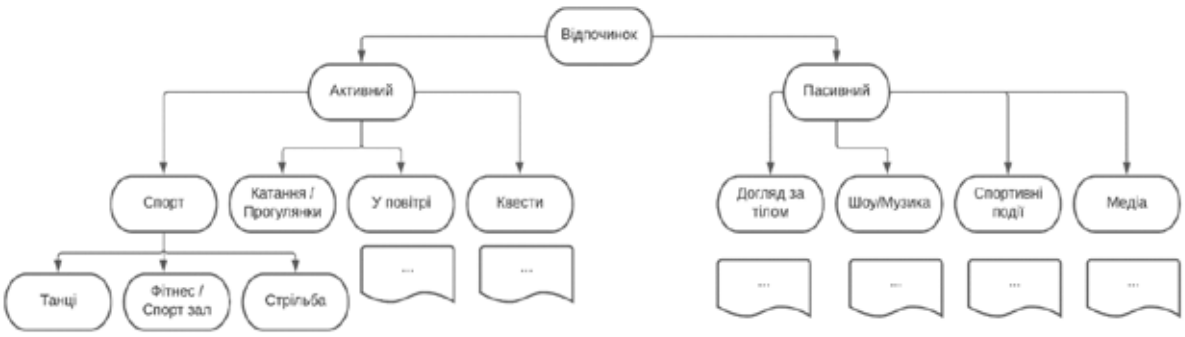

Рис. 1. Фрагмент дерева структури подій

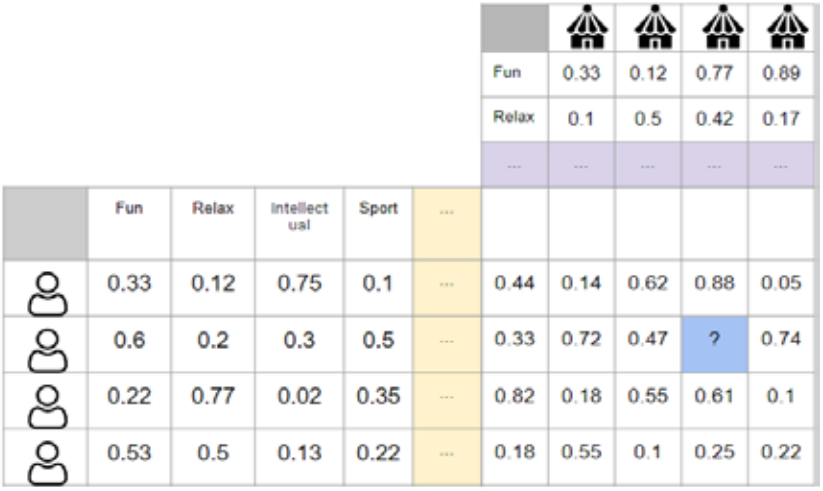

Рис. 2. Матриця факторизацій

У даній роботі розроблено рекомендаційну систему з матрицею факторизацій, де дані користувачів представляють собою психологічний портрет, а дані подій – параметри опису подій після співставлення обох матриць (рис. 2).

Більшість метаевристичних алгоритмів оптимізації засновані на ройовій поведінці тварин або біологічних взаємодіях живих організмів. Саме це надає метаевристичним алгоритмам простоту програмної реалізації, оскільки вони мають однакову реалізацію, яка базується на ініціалізації початкової генерації, яка розкладається на вектор, до якого застосовується цільова функція, створюється критерій зупинки. Перевага використання запропонованого підходу полягає в тому, що з часом дослідник розробляє кращі та досконаліші методи оптимізації, при цьому сама база буде оновлюватися шляхом додавання нових алгоритмів і, відповідно, не відбуватиметься процес стагнації рекомендаційної системи. Результати порівняння використання класичного методу зворотного поширення помилки (MLP + BP) та запропонованого методу з використанням метаевристичної оптимізації (MLP + MOA) наведено в табл. 1. Запропонований метод має перевагу за рахунок того, що метаевристичні алгоритми оптимізації надають нейронній мережі більший пошуковий простір і мають можливість виходити з локальних пасток, на відміну від використання таких класичних методів, як стохастичний градієнт.

Програмна реалізація запропонованої рекомендаційної системи

Для апробації розробленої методики була створена відповідна клієнтська частина програмного забезпечення у вигляді мобільного додатку. Використано наступні технології для створення клієнтської частини програмного забезпечення запропонованої системи рекомендацій:

1. Flutter – для реалізації в Google Play, App Store;
2. Firebase API – для сповіщень системи;
3. REST API – для зв'язку з серверами;
4. React JS – для створення вебдодатку.

Для розробки серверної частини програмного забезпечення використані наступні технології:

1. .NET Core Web API є основним сервером;
2. AZURE MS SQL – для створення бази даних;
3. GOOGLE MAPS API – для відтворення розташування зон відпочинку;
4. ASP.NET MVC – панель адміністратора;
5. Python, Tensorflow, MOA, PyTorch, PySwarm – для створення психологічного портрета користувача системи рекомендацій.

Таблиця 1

Порівняння застосування класичного підходу з використанням зворотного поширення та запропонованого методу з використанням алгоритмів метаевристичної оптимізації

Метод	Точність
Класичний (MLP + BP)	76.68%
Запропонований (MLP + MOA)	81.37%

База даних запропонованої рекомендаційної системи складається з 45 таблиць і займає ~500 Мб на жорсткому диску серверів US-1 у хмарній системі Azure. Після додавання користувача в систему рекомендацій вступає в дію алгоритм формування психологічного портрета користувача та алгоритм рекомендацій, де базовий психологічний портрет користувача будується на основі початкового вибору. Система рекомендацій приймає події, відвідані користувачем, параметри, вибрані користувачем, а також згенерований психологічний портрет як параметри для генерації

рекомендованих подій. Кожен вибір, зроблений користувачем, зберігається в системі, що забезпечує процес навчання для нейронної мережі, тому система може надавати більш точні рекомендації, а також пропонувати людей відповідно з подібним психологічним профілем. Тестування проходило на вибірці даних для міста Київ (Україна) в кількості 6000 подій, які були отримані за допомогою сервісів Google Maps API та додано >30 місць і подій через створене адміністрування даних системи рекомендацій «Планувальник розваг».

Висновки та подальша робота. У даному дослідженні розглянуто алгоритми, які дозволяють точніше створювати рекомендаційні системи, а отже, й покращувати конверсію користувачів та їхню взаємодію з тією чи іншою системою на основі отриманих даних, що, в свою чергу, дозволяє покращити точність аналізу даних для отримання цільової множини користувачів системи. Запропонований програмний метод створення рекомендаційної системи з використанням методів метаевристичної оптимізації забезпечує більшу точність у порівнянні з класичною реалізацією у середньому на 4–6%. Модульна структура програмного забезпечення, модулі якої реалізують зазначені алгоритми метаевристичної оптимізації, полегшує реалізацію запропонованого програмного методу в цілому.

Список літератури:

1. Vasileios Perifanis, Pavlos S. Efraimidis, Federated Neural Collaborative Filtering, Knowledge-Based Systems, Volume 242, 2022, 108441, ISSN 950-7051, https://doi.org/10.1016/j.knosys.2022.108441.

2. Mirjalili, Seyedali & Mirjalili, Seyed & Lewis, Andrew. (2014). Grey Wolf Optimizer. Advances in Engineering Software. 69. 46–61. 10.1016/j.advengsoft.2013.12.007.

3. M. Cells, B. Rylander, Neural network learning using particle swarm optimization, Advances in Information Science and Soft Computing, 2002, pp. 224– 226.

4. Y. Tian, S. Peng, X. Zhang, T. Rodemann, K. C. Tan and Y. Jin. A Recommender System for Metaheuristic Algorithms for Continuous Optimization Based on Deep Recurrent Neural Networks. IEEE Transactions on Artificial Intelligence, vol. 1, no. 1, pp. 5-18, Aug. 2020, doi: 10.1109/TAI.2020.3022339.

5. Mirjalili, S. and A. Lewis. The Whale Optimization Algorithm. Advances in Engineering Software, 95 (2016) 51–67.

6. L.D. Mech. Alpha status, dominance, and division of labor in wolf packs. Canadian Journal of Zoology, vol. 77, pp. 1196- 1203, 1999.

Oleshchenko L.M., Vernik M.O. PRINCIPLES OF RECOMMENDER SYSTEMS DEVELOPMENT USING METAHEURISTIC OPTIMIZATION

Most metaheuristic optimization algorithms are based on swarm behavior of animals or biological interactions of living organisms. This is what gives metaheuristic algorithms the simplicity of software implementation, which is based on the initialization of the initial generation, the formation of a vector to which the objective function is applied and the stopping criterion is created. The advantage of using the proposed approach for the recommender systems development is that over time the researcher develops better and more advanced optimization methods, while the base itself will be updated by adding new algorithms and, accordingly, the process of stagnation of the recommendation system will not occur. Algorithms that allow for more accurate creation of recommender systems and, therefore, to improve user conversion and their interaction with a particular system based on the obtained data, are considered in this study, which allows to improve the accuracy of data analysis to obtain the target set of system users. The proposed method of creating a recommender system using metaheuristic optimization methods provides greater accuracy compared to the classical implementation by an average of 4-6%. The modular structure of the software, the modules of which implement the specified metaheuristic optimization algorithms, facilitates the implementation of the proposed software method as a whole. To test the developed methodology, the client part of the software was created in the form of a mobile application. Programming technologies were used for the client part of the software of the proposed recommendation system: Flutter, Firebase API, REST API, React JS. The following technologies were used to develop the server part of the software: .NET Core Web API, AZURE MS SQL, GOOGLE MAPS API, ASP.NET MVC, Python, Tensorflow, MOA, PyTorch, PySwarm. The database of the proposed system consists of 45 tables and occupies ~500 MB on the hard disk of US-1 servers in the Azure cloud system. Each choice made by the user is stored in the system, which provides a learning process for the neural network, so the system can provide more accurate recommendations and also suggest people with a similar psychological profile accordingly. The method was tested on a sample of data for the city of Kyiv (Ukraine) in the amount of 6,000 events, which were obtained using Google Maps API services.

Key words: *MOA, recommendation system, neural networks, factorization matrix, backpropagation, multilayer perceptron vector, software.*