

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-57-11>

УДК 004.[4+8]

Копильчак Олег Андрійович, аспірант

<https://orcid.org/0009-0009-3295-6887>

Казимира Ірина Ярославівна, к.т.н., доцент

<https://orcid.org/0000-0003-1597-5647>

Національний університет «Львівська політехніка», м. Львів, Україна

## ГІБРИДНА МОДЕЛЬ ДЛЯ ЕФЕКТИВНОГО ФОРМУВАННЯ ПЕРСОНАЛІЗОВАНИХ РЕКОМЕНДАЦІЙ НАВЧАЛЬНИХ КУРСІВ

**Копильчак О.А., Казимира І.Я.** Гібридна модель для ефективного формування персоналізованих рекомендацій навчальних курсів. У цьому дослідженні представлено гібридну модель, яка поєднує колаборативну фільтрацію та глибокі нейронні мережі для підвищення ефективності навчальних рекомендацій. Такий підхід дозволяє оптимізувати процес генерації рекомендацій, забезпечуючи високу гнучкість і точність навіть при обмеженій кількості даних. У випадках, коли наявних даних недостатньо для ефективного навчання нейронної мережі, алгоритм автоматично переходить до використання колаборативної фільтрації, що зберігає якість рекомендацій і стабільність роботи моделі. Результати тестування на наборах даних показали, що гібридна модель підвищує точність і адаптивність рекомендаційної системи порівняно з традиційними методами. Поєднання двох методів дозволяє використати їхні переваги та мінімізувати недоліки, що робить модель універсальною в різних контекстах і з різними обсягами даних. Це особливо актуально у сфері освіти, де дані користувачів можуть бути обмеженими або неповними. Впровадження цієї моделі дає можливість створювати більш ефективні освітні платформи, які забезпечують персоналізоване навчання для широкої аудиторії, враховуючи індивідуальні потреби та вподобання користувачів. Важливою особливістю алгоритму є його здатність адаптуватися до змін у навчальному контенті та налаштовувати рекомендації відповідно до потреб кожного користувача. Це сприяє підвищенню задоволеності користувачів і покращує загальну якість освітніх послуг. Ефективна робота з обмеженими даними підвищує точність і адаптивність системи рекомендацій. Усі ці властивості роблять запропонований алгоритм перспективним інструментом для розвитку персоналізованих освітніх платформ, які краще відповідають індивідуальним потребам. Це сприяє ефективнішому навчанню та кращому засвоєнню матеріалу, що є важливим для розвитку сучасного інформаційного суспільства.

**Ключові слова:** персоналізоване планування навчання, глибоке навчання, колаборативна фільтрація, рекомендаційна система.

**Kopylchak O., Kazymyra I.** Hybrid model for the efficient formation of training courses personalized recommendations. In this study, a hybrid model that combines collaborative filtering and deep neural networks is presented to improve the effectiveness of learning recommendations. This approach allows to optimise the recommendation generation process, providing high flexibility and accuracy even with a limited amount of data. In cases where the available data is not enough to train the neural network efficiently, the algorithm automatically switches to collaborative filtering, which preserves the quality of recommendations and the stability of the model. The results of testing on datasets have shown that the hybrid model improves the accuracy and adaptability of the recommendation system compared to traditional methods. The combination of the two methods allows to use their advantages and minimise their disadvantages, making the model versatile in different contexts and with different amounts of data. This is especially relevant in the field of education, where user data may be limited or incomplete. Implementation of this model makes it possible to create more effective educational platforms that provide personalised learning for a wide audience, taking into account the individual needs and preferences of users. An important feature of the algorithm is its ability to adapt to changes in learning content and customise recommendations according to the needs of each user. This helps to increase user satisfaction and improves the overall quality of educational services. Effective work with limited data increases the accuracy and adaptability of the recommendation system. All of these properties make the proposed algorithm a promising tool for developing personalised educational platforms that better meet individual needs. This contributes to more efficient learning and better assimilation of the material, which is important for the development of the modern information society.

**Keywords:** personalized education planning, deep learning, collaborative filtering, recommender system.

**Постановка проблеми.** У сучасному світі, де обсяги інформації зростають експоненціально, автоматизація навчального процесу є необхідністю. Багато освітніх сервісів або не використовують рекомендаційні системи, або роблять це обмежено. Зі збільшенням обсягів даних користувачу стає все важче знаходити релевантну інформацію, тому автоматизовані системи рекомендацій стають особливо важливими. Основною проблемою є те, що більшість рекомендаційних систем потребують великої кількості даних для ефективного функціонування, що не завжди доступно в освітніх платформах [1]. Підхід, описаний у цій роботі, спрямований на вирішення цієї проблеми.

*Об'єктом дослідження* є процеси формування персоналізованих рекомендацій щодо навчальних курсів на освітніх платформах. *Предметом дослідження* є алгоритми, моделі і методи формування персоналізованих рекомендацій, зокрема колаборативна фільтрація, змістова фільтрація, глибокі нейронні мережі.

*Мета роботи* полягає в розробленні гібридної моделі формування персоналізованих рекомендацій навчальних курсів для подолання обмежень існуючих методів і забезпечення високої точності і гнучкості при різних обсягах даних.

*Науковою новизною дослідження* є розроблення гібридної моделі, яка, завдяки поєднанню колаборативної фільтрації та машинного навчання, покращує якість рекомендацій.

*Практична значущість отриманих результатів* – запропонована гібридна модель може бути легко інтегрована в будь-яку навчальну платформу, що дозволить більш ефективно генерувати рекомендації навчальних курсів системам з різним обсягом даних.

**Аналіз останніх досліджень та публікацій.** За останні роки попит на навчальні сервіси, як Coursera, Udeму чи інші сильно зросли, адже люди все більше часу приділяють навчанню для того, щоб бути конкурентоспроможними на ринку праці. Враховуючи, що користувачі не завжди знають, який курс для них підійде найкраще, дуже важливо впровадити систему рекомендацій, яка б пропонувала користувачам лише релевантну і потрібну для них інформацію.

Проаналізуємо методи, які зазвичай використовують для створення рекомендаційних систем, а також їх переваги та недоліки.

Колаборативна фільтрація (КФ) – це популярний алгоритм рекомендацій, який базує свої прогнози та рекомендації на рейтингах або поведінці інших користувачів у системі. Фундаментальне припущення цього методу полягає в тому, що думки інших користувачів можуть бути відібрані та агреговані таким чином, щоб забезпечити обґрунтоване передбачення вподобань активного користувача [2]. Частково даний метод буде використано у розроблюваній системі для того, щоб зменшити залежність від об'ємів навчальних даних, яка притаманна для моделей штучного інтелекту.

У статті [3] наведено систематичний огляд методів колаборативної фільтрації, включаючи традиційні підходи, такі як користувачко-орієнтована та елементно-орієнтована фільтрація, а також гібридні методи, що поєднують соціальні мережі для вирішення проблеми розрідженості даних та холодного старту. Особливу увагу приділено впливу соціальних факторів, таких як довіра між користувачами, на якість рекомендацій, а також ролі соціальних зв'язків у подоланні нестачі інформації про нових користувачів та елементи. Автори досліджують, як довіра та соціальна взаємодія можуть значно покращити якість рекомендацій у тих випадках, коли історичних даних недостатньо, і пропонують використовувати ці фактори як ключові параметри для подальшої персоналізації. Комбінування колаборативної фільтрації з іншими підходами, такими як фільтрація на основі змісту, дозволяє ефективно інтегрувати соціальні аспекти для покращення рекомендацій та забезпечення більшого охоплення і точності в рекомендаційних системах. Як основну метрику для визначення ступеня схожості між користувачами на основі їх спільних оцінок використовують косинусну подібність:

$$\text{sim}(u, v) = \frac{\sum_{i \in I_{uv}} r_{ui} * r_{vi}}{\sqrt{\sum_{i \in I_u} r_{ui}^2} * \sqrt{\sum_{i \in I_v} r_{vi}^2}}$$

Ця метрика дозволяє визначати, наскільки два користувачі мають схожі вподобання, що в свою чергу допомагає підвищити релевантність наданих рекомендацій.

Фільтрування, що базується на змісті, – ця методика використовує властивості елементів, щоб рекомендувати додаткові елементи, схожі на ті, що сподобалися користувачеві, на основі його попередніх дій або явного відгуку [4]. Даний підхід можна ефективно поєднати з моделями штучного інтелекту, наприклад, з нейронною мережею. У статті [5] представлено глибоку нейронну колаборативну фільтрацію (Deep Neural Collaborative Filtering, DDCF), яка об'єднує методи колаборативної фільтрації з глибокими нейронними мережами для покращення точності персоналізованих рекомендацій, досягаючи точності 0.85 та CTR (Click-Through Rate) 0.12. Для оцінки ефективності системи використовуються такі формули:

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}}$$

$$\text{Recall} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$

$$\text{Click – Through Rate (CTR)} = \frac{\text{Total clicks}}{\text{Total recommendations}}$$

У розроблюваній гібридній моделі також використовується поєднання змістового фільтрування з нейронною мережею, але акцент робиться на комбінованій обробці даних про курси та дії користувачів для точнішого прогнозування вподобань у контексті навчальних платформ. Відмінність у використанні власних моделей для аналізу взаємодій дозволяє краще адаптувати систему до освітнього середовища, в той час як підхід у статті [5] орієнтований на ширшу сферу електронної комерції.

Гібридні методи поєднують два або більше методів рекомендацій для отримання кращої продуктивності з меншою кількістю недоліків кожного з них. Найчастіше колаборативну фільтрацію поєднують з іншими методами, щоб уникнути проблеми нарощування продуктивності [6]. Існують різні підходи до комбінування методів: ваги – у кожного з методу є певна вага і в результаті усі з них впливають на кінцеву рекомендацію відповідно до своєї ваги; перемикання – вибір методу залежить від конкретної ситуації (наприклад, об'єму наявних даних); поєднання – підхід, коли кілька рекомендаційних методів використовуються одночасно; каскад – перший метод покращує рекомендації другого (може використовуватись з двома і більше методами). Далі в дослідженні буде використано саме гібридну модель, яка комбінуватиме колаборативну фільтрацію разом з фільтруванням за змістом задля досягнення оптимальних результатів з різними об'ємами даних.

Переваги та недоліки основних підходів представлено у таблиці 1.

Таблиця 1. Порівняння основних підходів у рекомендаційних системах

| Підхід                        | Переваги  | Недоліки   |
|-------------------------------|---|--|
| Колаборативна фільтрація (КФ) | <ul style="list-style-type: none"> <li>• Доменна інформація не потрібна</li> <li>• Ефективність зростає з часом</li> <li>• Може визначати крос-жанрові ніші</li> <li>• Досить неявного зворотнього зв'язку</li> </ul> | <ul style="list-style-type: none"> <li>• Проблема «сірої вівці»</li> <li>• Проблема холостого старту для нових користувачів</li> <li>• Потрібно обирати між стабільністю та гнучкістю</li> </ul>                                     |
| Змістова фільтрація           | <ul style="list-style-type: none"> <li>• Доменна інформація не потрібна</li> <li>• Ефективність зростає з часом</li> <li>• Досить неявного зворотнього зв'язку</li> </ul>   | <ul style="list-style-type: none"> <li>• Проблема холостого старту для нових користувачів</li> <li>• Ефективність залежить від великого обсягу історичних даних</li> <li>• Потрібно обирати між стабільністю та гнучкістю</li> </ul> |

Як можемо бачити з таблиці, обидва підходи мають свої плюси та мінуси. Перевага колаборативної фільтрації над змістовою у тому, що вона дозволяє досить ефективно рекомендувати елементи навіть, якщо в системі присутні дані з різних ніш (наприклад, навчальні курси та музика), також даний підхід краще працює в умовах обмеженої кількості даних, проте може стикатись з проблемою холостого старту для нових елементів. При використанні змістової фільтрації відсутня проблема «сірої вівці», тобто проблема, коли для користувача неможливо підібрати достатньо хорошу рекомендацію. Далі розглянемо підходи для вирішення деяких з цих проблем шляхом комбінування двох підходів.

Для побудови моделі змістової фільтрації, як однієї зі складових гібридної моделі, що розробляється, використано глибоку нейронну мережу – це один із найбільш потужних підходів у рекомендаційних системах. У статті [7] підкреслено здатність глибоких нейронних мереж до навчання нелінійних взаємозв'язків між користувачами і елементами, а також можливість об'єднання різноманітних типів даних, таких як текстові, візуальні та контекстуальні. На відміну від підходу, який розглядається в даній роботі, що фокусується на рекомендаціях для навчальних курсів, стаття [7] розглядає широке застосування в електронній комерції, що дозволяє використовувати мультимодальні дані для покращення якості рекомендацій. Ця стаття також

підкреслює ключові переваги глибоких нейронних мереж, такі як нелінійні перетворення та представлення даних, що дозволяють ефективніше виявляти складні взаємозв'язки.

Проаналізуємо підхід до рекомендацій, який використовує UdeMy – один з найбільших сервісів на ринку, що надає доступ до великої кількості навчальних курсів. Цей сервіс обрано для аналізу через його популярність та високу ефективність.

UdeMy – це платформа онлайн-навчання, орієнтована на професіоналів та студентів, розроблена у травні 2010 року. Станом на січень 2020 року платформа налічувала понад 50 мільйонів студентів та 57 000 викладачів, які викладають курси більш ніж 65 мовами. Було зареєстровано понад 295 мільйонів курсів [8]. Система рекомендацій UdeMy призначена для двох основних типів користувачів: тих, хто вивчає нові теми і тих, хто прагне вдосконалити існуючі навички. Мета полягає в тому, щоб надати високоякісні, актуальні пропозиції курсів за розумними цінами, які відповідають рівню зацікавленості користувача та глибині предмету.

Система використовує як пакетні, так і потокові процеси обробки даних, щоб пропонувати рекомендації в реальному часі. Для нових користувачів або тих, хто не зареєструвався, пропозиції курсів надаються на основі широкої відповідності пошуковим запитам, надаючи пріоритет високорейтинговим курсам з численними рейтингами. Коли користувач натискає на курс, система пропонує додаткові рекомендації на основі того, що інші студенти придбали разом з цим курсом, використовуючи історичні дані. Для повернення користувачів з історією покупок система рекомендацій теоретично повинна персоналізувати пропозиції на основі минулих пошуків і покупок. Однак поточна реалізація не виглядає повністю адаптованою в режимі реального часу, оскільки навіть досвідчені користувачі отримують загальні результати пошуку, подібні до нових користувачів. UdeMy також надсилає рекомендації електронною поштою, які здаються більш персоналізованими та ефективними, ніж пропозиції на сайті в режимі реального часу. Ці листи враховують історію реєстрації користувача та його пошукову активність.

Враховуючи раніше проаналізовану інформацію про сервіс UdeMy, можемо виділити кілька проблемних місць:

- Для нових користувачів рекомендуються високорейтингові курси, не враховуючи дані про користувача.
- Під час рекомендацій у реальному часі система неефективно враховує історичні дані.
- Хоча рекомендації, що приходять на електронну пошту, є кращими порівняно з рекомендаціями в режимі реального часу, проте це висвітлює проблему можливості швидко опрацьовувати великі обсяги даних і надавати якісні рекомендації в режимі реального часу.

Отже, підсумувавши наведену вище інформацію, можемо зробити висновок, що зараз – пік розвитку рекомендаційних систем, адже об'єми даних, якими оперують компанії невпинно зростають і стає все важче знаходити релевантну інформацію самотужки без використання спеціалізованих систем. Особливо це помітно у сфері освіти, де проблема не у відсутності матеріалів та курсів для навчання, а у занадто великому виборі. При тому системи не завжди мають достатньо інформації про користувачів та їхні вподобання, тому рекомендаційна система, яка зможе ефективно адаптуватись під різні об'єми інформації та генерувати якісні рекомендації, є дуже затребуваною. Після здійснення аналізу системи-аналогу – UdeMy можна дійти висновку, що система має проблемні місця, які запропонований гібридний алгоритм зміг би повністю або частково вирішити, а саме: проблеми холодного старту та ефективних рекомендацій у режимі реального часу.

**Виклад основного матеріалу.** Основою будь-якої моделі рекомендацій є її алгоритм, тому, перш за все, сфокусуємось на описі запропонованого алгоритму та опису особливостей його реалізації. Раніше було розглянуто кілька основних підходів, які застосовуються при побудові рекомендаційних моделей, а саме: колаборативна фільтрація, фільтрування, що базується на змісті, і гібридні методи. Оскільки нашою метою є створення моделі, що повинна добре справлятися з різними об'ємами даних, то було прийнято рішення використати гібридний підхід, поєднавши колаборативну фільтрацію та змістове фільтрування. Схожий підхід описаний у статті Р. Бурка [9], де описана ефективність такого підходу у веб-сервісах. Зокрема тут зазначається, що колаборативну та змістову фільтрацію можна об'єднати так, щоб компонент, заснований на знаннях, міг вирішити проблему "холодного старту" та надати рекомендації новим користувачам, чий профілі занадто обмежені для ефективної колаборативної фільтрації. Тим часом колаборативна фільтрація може

використовувати свої статистичні можливості для виявлення користувачів зі схожими вподобаннями, які займають унікальні ніші в просторі інтересів, які важко передбачити навіть експертам.

Розглянемо детальніше один із основних підходів, який буде застосовано у пропонуваній гібридній моделі, – колаборативну фільтрацію. Цей метод спрямований на виявлення схожостей між користувачами, що дозволяє створювати релевантні рекомендації навіть у випадках, коли інформація про користувача обмежена. Основний принцип колаборативної фільтрації полягає в припущенні, що користувачі з подібними оцінками або поведінкою на платформі, ймовірно, матимуть схожі вподобання та інтереси. У запропонованій моделі колаборативна фільтрація має кілька ключових відмінностей від традиційних підходів.

Найважливіша з них – це використання спеціально розробленого коефіцієнта відмінності, що дозволяє адаптувати модель під специфіку навчальної платформи. Цей коефіцієнт враховує різні характеристики користувачів, такі як вік, професія та середні оцінки по категоріях курсів. Налаштовувані ваги для кожного параметра додають моделі гнучкості, що дозволяє краще адаптувати систему до потреб навчальних платформ і специфіки освітнього домену. Завдяки цьому модель може враховувати відмінності між користувачами більш точно, надаючи рекомендації, які стають не лише персоналізованими, але й максимально релевантними для освітнього контексту та індивідуальних потреб користувачів. Коефіцієнт відмінності змінюється в діапазоні від 0 до 1, де значення 0 означає, що користувачі є максимально схожими, а 1 – що вони кардинально різняться. Обчислюється цей коефіцієнт за формулою:

$$D(u, v) = \frac{\sum_{i=1}^N \omega_i * d_i(u, v)}{\sum_{i=1}^N \omega_i},$$

де  $D(u, v)$  – коефіцієнт відмінності між користувачем  $u$  та  $v$ ;

$N$  – загальна кількість змінних (атрибутів);

$\omega_i$  – вага  $i$ -тої змінної;

$d_i(u, v)$  – різниця в  $i$ -тій змінній;

Підрахунок  $d_i(u, v)$  для дійсних атрибутів здійснюють так:

$$d_i(u, v) = \frac{|x_i(u) - x_i(v)|}{x_i^{max} - x_i^{min}}.$$

Підрахунок  $d_i(u, v)$  для категоріальних атрибутів виконують за формулою:

$$d_i(u, v) = \begin{cases} 0, & \text{якщо } x_i(u) = x_i(v) \\ 1, & \text{якщо } x_i(u) \neq x_i(v) \end{cases}$$

Іншою важливою особливістю, яка відрізняє колаборативну фільтрацію в даній роботі від інших, є наявність початкових даних для нових користувачів. Багато імплементацій даного алгоритму стикаються з проблемою «холодного пуску», але оскільки специфіка навчальних платформ вимагає від користувача реєстрації для доступу до курсів, то це частково вирішує проблему з відсутністю даних для нових користувачів і в результаті покращує ефективність застосування підходу. Колаборативна фільтрація, як окремий алгоритм, має низку переваг, таких як ефективність при обмеженій кількості даних, адаптивність до освітнього домену завдяки запропонованому коефіцієнту відмінності та гнучкість. Однак алгоритм має суттєвий недолік – він показує не найкращу якість рекомендацій при достатньо великій кількості даних, адже на відміну від, наприклад, нейронних мереж, гірше знаходить складні зв'язки між даними. Саме тому колаборативна фільтрація є лише одним з компонентів гібридної моделі, де її недоліки компенсуються, що забезпечує оптимальну ефективність моделі.

Другою основною складовою гібридної моделі є змістова фільтрація, що базується на використанні глибокої нейронної мережі. Такий підхід дозволяє прогнозувати вподобання користувачів на основі характеристик курсів та історії їхніх взаємодій із платформою. Цей підхід є особливо ефективним при наявності великої кількості даних, оскільки він здатний враховувати складні зв'язки між атрибутами курсів і вподобаннями користувачів. Архітектура мережі була ретельно підібрана на основі експериментів, що дозволило визначити її як найбільш оптимальну для роботи з конкретним набором навчальних даних, використаних у процесі тренування моделі. Завдяки такому підходу вдалося досягти високої точності прогнозів і максимально адаптувати мережу до особливостей платформи, дані якої було використано під час тренування моделі.

- Вхідний шар: вхідний шар представляє собою повнозв'язний шар (Dense) з кількістю нейронів, рівною кількості стовпців вхідних даних. Активаційна функція 'relu' використовувалась для забезпечення нелінійності моделі, що дозволяє мережі вчитися складним залежностям у даних.
- Приховані шари: конфігурація включає три приховані шари, кожен з яких має 50 нейронів. Активаційна функція 'relu' використовується у кожному шарі. Конфігурація мережі була підібрана експериментально, що забезпечило максимальну ефективність цієї структури.

Вихідний шар представлений одним нейроном, який генерує числове значення – прогнозований рейтинг курсу для конкретного користувача. Цей показник визначає, наскільки ймовірно, що курс відповідатиме вподобанням користувача. На рис. 1 зображена топологія нейронної мережі, яка використана в гібридній моделі.

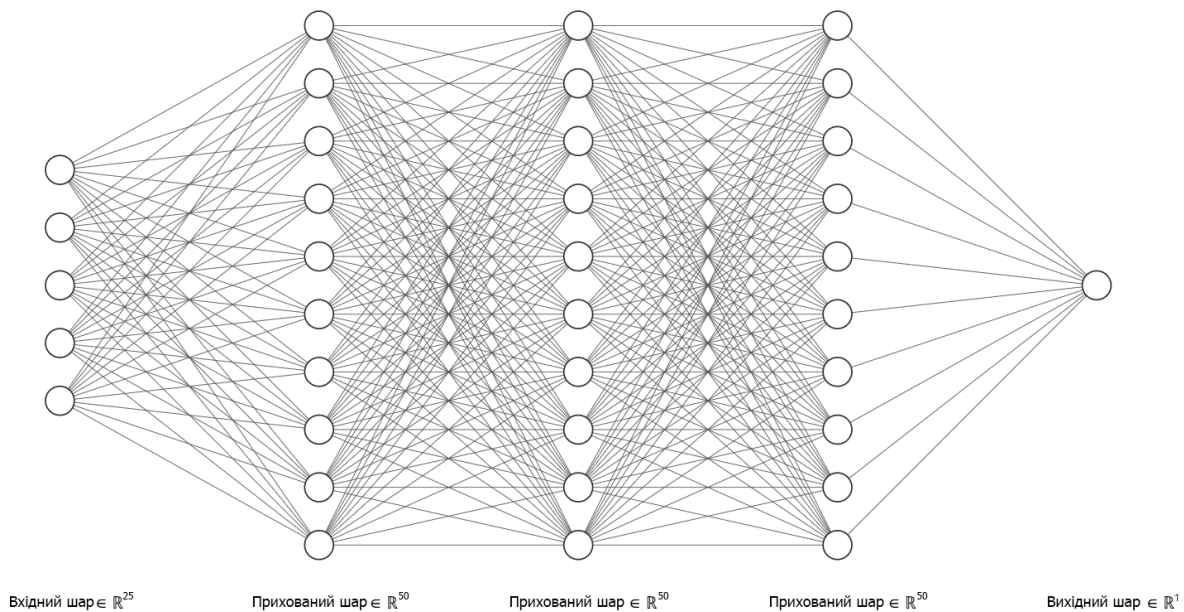


Рис. 1. Топологія нейронної мережі, використана в гібридній моделі

Ефективність нейронної мережі у рекомендаційній системі добре проілюстровано в статті [10], де зазначається, що моделі глибокого навчання вміють вивчати представлення ознак з необроблених даних, а це може бути особливо корисним для фільтрації на основі змісту, при якій ознаки елементів відіграють важливу роль.

Розглянемо детальніше використані функції втрат, оскільки вони не лише задіяні у процесі навчання моделі, а й служать мірилом її ефективності.

Mean Squared Error (MSE) – одна з найпопулярніших функцій втрат, MSE знаходить середнє значення квадратів різниць між цільовими та прогнозованими виходами:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y^{(i)} - \hat{y}^{(i)})^2.$$

Ця функція має численні властивості, що роблять її особливо придатною для розрахунку втрат. Різниця піднесена до квадрату дозволяє надавати більшої ваги значенням з великою похибкою. MSE також є опуклою функцією з чітко визначеним глобальним мінімумом – це дозволяє легше використовувати оптимізацію методом градієнтного спуску для встановлення вагових значень. Окрім цього, MSE має просту похідну, що робить її обчислювально ефективною для алгоритмів градієнтної оптимізації [11].

Mean Absolute Error (MAE) знаходить середнє значення абсолютних відмінностей між цільовими та прогнозованими результатами:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y^{(i)} - \hat{y}^{(i)}|.$$

Ця функція втрат у деяких випадках використовується як альтернатива MSE. MSE дуже чутливий до викидів, які можуть різко вплинути на втрати, оскільки відстань є квадратом. MAE використовується у випадках, коли навчальні дані мають велику кількість викидів, щоб пом'якшити це. Окрім цього MAE накопичує похибку лінійно, що може бути корисним, коли метрика продуктивності має бути краще інтерпретованою та прямо пропорційною похибці, зробленій моделлю [12].

Гібридна модель поєднує в собі два алгоритми, які були розглянуті вище, а саме колаборативної та змістової фільтрації. Основною ідеєю гібридного підходу є динамічне перемикання між двома методами рекомендацій залежно від доступного обсягу даних. Коли обсяг даних обмежений, алгоритм надає пріоритет колаборативній фільтрації, яка дозволяє знаходити схожих користувачів і надавати рекомендації на основі їхніх вподобань. З іншого боку, коли даних вдосталь, алгоритм переходить до змістового фільтрування, що використовує нейронну мережу для прогнозування рейтингу курсів. Для наочності наведемо діаграму, що ілюструє процес функціонування гібридного алгоритму (рис. 2). На діаграмі активності зображено, як гібридна модель вирішує, який з алгоритмів використовувати. Після вибору алгоритму виконується прогнозування рейтингу курсів. Після прогнозування рейтингів відбувається генерування рекомендацій. Список рекомендацій залежить від згенерованих рейтингів. Стандартна поведінка: в рекомендацію потрапляють ті курси, які мають найбільший прогнозований рейтинг.

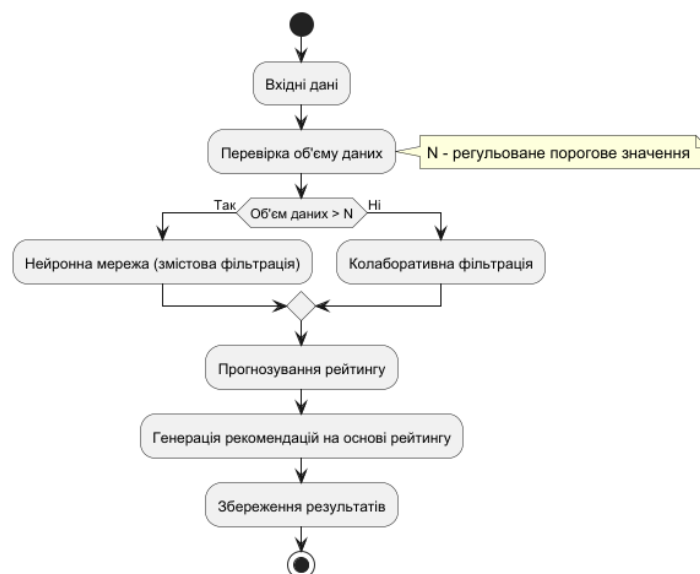


Рис. 2. Діаграма гібридного алгоритму рекомендацій

Гібридний підхід має кілька ключових переваг, які роблять його ефективним у навчальних платформах, а саме:

- Гнучкість і адаптивність: алгоритм автоматично підлаштовується під доступний обсяг даних, що робить його придатним для роботи з новими та маловідомими користувачами, а також з тими, хто активно взаємодіє із системою. Завдяки цьому забезпечується постійна висока точність рекомендацій
- Оптимальна точність: завдяки поєднанню двох методів, гібридний алгоритм дозволяє компенсувати слабкі місця кожного з підходів, підсилюючи їхні переваги. Колаборативна фільтрація особливо ефективна при малому об'ємі даних, забезпечуючи релевантні рекомендації навіть за обмеженої інформації про користувачів, тоді як нейронна мережа в змістовому фільтруванні здатна проводити глибокий аналіз при великих обсягах даних, виявляючи складні зв'язки та підвищуючи загальну точність рекомендацій.
- Зменшення проблеми холодного старту: поєднання методів частково вирішує проблему "холодного старту", адже як було вказано раніше, завдяки специфіці доменної області, колаборативна фільтрація завжди матиме на вході мінімальний набір даних, таким чином, навіть для нових користувачів, система може надавати релевантні рекомендації.

На рис. 3 зображена частина вибірки даних з інформацією про початкові курси.

| course_id | course_title   | url                              | price | num_subscribers | num_reviews | num_lectures | level             | rating | content_duration | published_timestamp  | subject             | date       | free/paid |
|-----------|--|----------------------------------|-------|-----------------|-------------|--------------|-------------------|--------|------------------|----------------------|---------------------|------------|-----------|
| 41295     | Learn HTML5 Programming From Scratch                         | https://www.udemy.com/learn-htr  | 0     | 268923          | 8629        | 45           | Beginner Level    | 0.82   | 10.Tpa           | 2013-02-14T07:03:41Z | Web Development     | 2013-02-14 | Free      |
| 59014     | Coding for Entrepreneurs Basic                               | https://www.udemy.com/coding-fo  | 0     | 161029          | 279         | 27           | Expert Level      | 0.69   | 03.Tpa           | 2013-06-09T15:51:52Z | Web Development     | 2013-06-09 | Free      |
| 625204    | The Web Developer Bootcamp                                   | https://www.udemy.com/the-web-b  | 200   | 121584          | 27445       | 342          | Beginner Level    | 0.89   | 43               | 2015-11-02T21:13:27Z | Web Development     | 2015-11-02 | Paid      |
| 173548    | Build Your First Website in 1 Week with HTML5 and CSS3       | https://www.udemy.com/build-you  | 0     | 120291          | 5524        | 30           | All Levels        | 0.78   | 3                | 2014-04-08T16:21:30Z | Web Development     | 2014-04-08 | Free      |
| 764164    | The Complete Web Developer Course 2.0                        | https://www.udemy.com/the-comp   | 200   | 114512          | 22412       | 304          | Beginner Level    | 0.55   | 30.Tpa           | 2016-03-08T22:28:36Z | Web Development     | 2016-03-08 | Paid      |
| 19421     | Free Beginner Electric Guitar Lessons                        | https://www.udemy.com/beginner-  | 0     | 101354          | 1042        | 95           | All Levels        | 0.88   | 04.Tpa           | 2012-06-15T17:00:33Z | Musical Instruments | 2012-06-15 | Free      |
| 473160    | Web Design for Web Developers: Build Beautiful Websites!     | https://www.udemy.com/web-desi   | 0     | 98867           | 6512        | 20           | All Levels        | 0.82   | 3                | 2015-04-13T18:29:47Z | Web Development     | 2015-04-13 | Free      |
| 94430     | Learn Javascript & JQuery From Scratch                       | https://www.udemy.com/learn-jav  | 30    | 84897           | 2685        | 10           | All Levels        | 0.79   | 2                | 2013-10-10T15:19:29Z | Web Development     | 2013-10-10 | Paid      |
| 130064    | Practical PHP: Master the Basics and Code Dynamic Websites   | https://www.udemy.com/code-dyn   | 0     | 83737           | 4598        | 45           | Intermediate Leve | 0.85   | 06.Tpa           | 2014-07-19T03:02:05Z | Web Development     | 2014-07-19 | Free      |
| 364426    | JavaScript: Understanding the Weird Parts                    | https://www.udemy.com/understar  | 175   | 79612           | 16976       | 85           | All Levels        | 0.69   | 11.Tpa           | 2015-03-12T23:19:54Z | Web Development     | 2015-03-12 | Paid      |
| 238934    | PianoForAll - Incredible New Way To Learn Piano & Keyboard   | https://www.udemy.com/pianofora  | 200   | 75499           | 7676        | 362          | Beginner Level    | 0.96   | 30               | 2014-08-07T06:27:51Z | Musical Instruments | 2014-08-07 | Paid      |
| 76150     | Angular 4 (formerly Angular 2): The Complete Guide           | https://www.udemy.com/the-comp   | 190   | 73783           | 19649       | 329          | Beginner Level    | 0.9    | 22               | 2016-02-11T07:29:29Z | Web Development     | 2016-02-11 | Paid      |
| 21386     | Beginner Photoshop to HTML5 and CSS3                         | https://www.udemy.com/psd-html   | 0     | 73110           | 1716        | 22           | All Levels        | 0.94   | 2                | 2012-07-27T12:54:57Z | Web Development     | 2012-07-27 | Free      |
| 65330     | Web Development By Doing: HTML / CSS From Scratch            | https://www.udemy.com/web-dev    | 0     | 72932           | 2575        | 21           | All Levels        | 0.39   | 1                | 2013-09-25T00:42:09Z | Web Development     | 2013-09-25 | Free      |
| 405926    | HTML and CSS for Beginners - Build a Website & Launch ONLINE | https://www.udemy.com/html-and-  | 0     | 70773           | 5660        | 50           | All Levels        | 0.57   | 6                | 2015-03-19T19:07:12Z | Web Development     | 2015-03-19 | Free      |
| 11174     | Become a Web Developer from Scratch                          | https://www.udemy.com/how-to-bi  | 120   | 69186           | 2408        | 197          | All Levels        | 0.61   | 27.Tpa           | 2011-11-19T21:51:20Z | Web Development     | 2011-11-19 | Paid      |
| 49798     | Bitcoin or How I Learned to Stop Worrying and Love Crypto    | https://www.udemy.com/bitcoin-or | 0     | 65576           | 936         | 24           | All Levels        | 0.56   | 8                | 2013-04-20T02:25:22Z | Business Finance    | 2013-04-20 | Free      |
| 314662    | Quickstart AngularJS   | https://www.udemy.com/quickstart | 0     | 64128           | 4047        | 17           | Beginner Level    | 0.96   | 01.Tpa           | 2014-11-27T12:42:22Z | Web Development     | 2014-11-27 | Free      |
| 128946    | Learn Responsive Web Development from Scratch                | https://www.udemy.com/learn-res  | 0     | 59029           | 2692        | 24           | All Levels        | 0.55   | 04.Tpa           | 2013-12-09T12:16:40Z | Web Development     | 2013-12-09 | Free      |
| 289230    | Learn and Understand AngularJS                               | https://www.udemy.com/learn-ang  | 175   | 55961           | 11550       | 55           | Beginner Level    | 0.87   | 7                | 2014-09-24T00:57:25Z | Web Development     | 2014-09-24 | Paid      |
| 461160    | Learn and Understand NodeJS                                  | https://www.udemy.com/understar  | 195   | 58208           | 11123       | 98           | Beginner Level    | 0.73   | 13               | 2015-08-31T21:07:48Z | Web Development     | 2015-08-31 | Paid      |

Рис. 3. Вибірка даних з інформацією про навчальні курси

Загальний обсяг вибірки складає 23 тис. курсів та близько 1 тис. користувачів [13]. Для забезпечення належної якості даних було виконано кілька етапів попередньої обробки, включно з фільтрацією, нормалізацією та кодуванням категоріальних даних. Також було виконано заповнення відсутніх значень, щоб уникнути пропусків, які могли б вплинути на точність прогнозів. Дані були розділені на навчальну та тестову вибірки, що дозволило оцінити здатність моделі узагальнювати результати та перевірити її точність у реальних умовах.

Для системи рекомендацій ключовими показниками є два параметри: час, який витрачається на генерацію рекомендацій, та їхня точність. Обидва ці показники відіграють критичну роль і потребують оптимізації, адже швидкість формування рекомендацій впливає на зручність користування системою, а точність визначає релевантність запропонованого контенту. Навчання/тренування моделі проводилось на сервері з процесором Intel Core i7, 16 ГБ оперативної пам'яті та графічною картою NVIDIA GeForce GTX 1650. Для оцінки точності рекомендацій використовували метрики MSE та MAE, а також методи перехресної валідації з різними розмірами тестових наборів даних. На наступному графіку (рис. 4) представлено середній час, необхідний для тренування моделі, що використовує змістове фільтрування. Оскільки модель функціонує на базі нейронної мережі, особливу увагу приділено залежності часу тренування від кількості навчальних епох, що дозволяє оцінити, як швидко модель досягає оптимальної точності при різній кількості ітерацій.

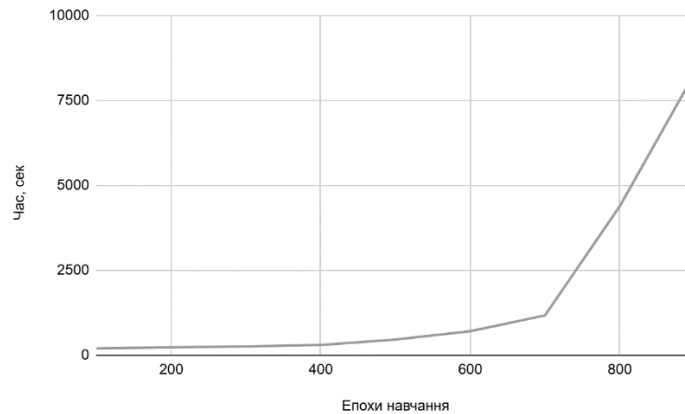


Рис. 4. Залежність часу навчання моделі від кількості епох

Експериментально доведено, що оптимальна кількість епох для обраної конфігурації мережі становить 500, що займає приблизно 710 секунд. Абсолютна похибка становить 19%, а при більшій кількості навчальних епох можна досягнути 17%, що є хорошим результатом. Щодо часу, що витрачається на генерацію рекомендацій, то він є прийнятним, адже це лише початкове тренування моделі і додаткове навчання вимагатиме менше часу та ресурсів при умові, що буде виконуватись періодично. У таблиці 2 наведено метрики, які використовувались для вимірювання ефективності рекомендаційної моделі:

Таблиця 2. Основні метрики оцінки моделі

| МЕТРИКА                           | ЗНАЧЕННЯ |
|-----------------------------------|----------|
| Середньоквадратична похибка (MSE) | 0.056    |



|                                      |        |
|--------------------------------------|--------|
| Середня абсолютна похибка (MAE)      | 0.1719 |
| Час навчання нейронної мережі        | 8300с  |
| Середня абсолютна відсоткова похибка | ~17%   |

Важливо зазначити, що дані, наведені в таблиці, отримано при тренуванні моделі впродовж 800 епох. Як уже було зазначено вище, можна отримати дещо гіршу точність, проте зі значно меншими затратами часу. Порівняно з використанням окремих алгоритмів, гібридна модель показала на 20% вищу точність, що підтверджує високу ефективність моделі в задачах персоналізованих рекомендацій. Для оцінки точності рекомендацій використовували метрики MSE та MAE, а також методи перехресної валідації з різними розмірами тестових наборів даних – від 5% (імітуючи достатню наявність даних) до 95% (імітуючи нестачу даних).

**Висновки та перспективи подальшого дослідження.** Розроблено ефективний гібридний алгоритм рекомендаційної системи для освітніх платформ, який завдяки поєднанню колаборативної та змістової фільтрації на основі нейронних мереж забезпечує високу точність рекомендацій навіть при обмеженій кількості даних. Основні результати дослідження показали, що алгоритм здатен генерувати релевантні та персоналізовані рекомендації, покращуючи користувацький досвід на освітніх платформах. Експериментальне дослідження на даних з платформи UdeMy підтвердило, що підібрана конфігурація нейронної мережі забезпечує високу точність прогнозів. Використання методів перехресної валідації на різних розмірах тестових наборів даних продемонструвало стабільність алгоритму і його здатність адаптуватися до різних умов. Окрім того, отримані результати вказують на можливість впровадження розробленого алгоритму в існуючі освітні платформи, що сприятиме підвищенню точності та персоналізації рекомендацій, а також підвищенню ефективності навчального процесу.

Запропонований гібридний алгоритм відкриває можливості для подальших досліджень і вдосконалення рекомендаційних систем. Перш за все, перспективним є дослідження ефективності алгоритму в інших доменах, де також важлива точність рекомендацій при обмеженій кількості даних, наприклад, в електронній комерції або сфері охорони здоров'я. Подальші дослідження можуть бути спрямовані на розробку більш адаптивних моделей, які враховуватимуть змінні потреби користувачів з часом та зможуть швидко адаптуватися до нових даних. Також актуальним є вивчення можливості інтеграції додаткових методів обробки природної мови (NLP) для аналізу текстового контенту курсів, що може ще більше підвищити точність рекомендацій.

#### Список бібліографічного опису

1. Doh, R. F., Zhou, C., Arthur, J. K., Tawiah, I., & Doh, B. (2022). A systematic review of deep knowledge graph-based recommender systems, with focus on explainable embeddings. *Data*, 7(7), 94. <https://doi.org/10.3390/data7070094>.
2. Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1), 5–53. <https://doi.org/10.1145/963770.963772>.
3. Chen, R., Hua, Q., Chang, Y.-S., Wang, B., Zhang, L., & Kong, X. (2018). A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access: Practical Innovations, Open Solutions*, 6, 64301–64320. <https://doi.org/10.1109/access.2018.2877208>.
4. Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In *The Adaptive Web* (pp. 325–341). Springer Berlin Heidelberg.
5. Messaoudi, F., & Loukili, M. (2024). E-commerce personalized recommendations: A deep neural collaborative filtering approach. *Operations Research Forum*, 5(1). <https://doi.org/10.1007/s43069-023-00286-5>.
6. Burke, R. (2002). User Modeling and User-Adapted Interaction, 12(4), 331–370. <https://doi.org/10.1023/a:1021240730564>.
7. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2020). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, 52(1), 1–38. <https://doi.org/10.1145/3285029>.
8. Rpubs - Recommender Systems Analysis - UdeMy's Recommender Engine. (n.d.). Rpubs.com. Retrieved November 6, 2024, from <https://rpubs.com/christianthieme/599308>.
9. Burke, R. (2007). Hybrid Web Recommender Systems. In *The Adaptive Web* (pp. 377–408). Springer Berlin Heidelberg.
10. Wang, X., Li, Q., Yu, D., Huang, W., & Xu, G. (2023). Neural Causal Graph Collaborative Filtering. In arXiv [cs.IR]. <https://doi.org/10.48550/ARXIV.2307.04384>.
11. Ruder, S. (2016). An overview of gradient descent optimization algorithms. In arXiv [cs.LG]. <https://doi.org/10.48550/ARXIV.1609.04747>.
12. Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? *Geoscientific Model Development Discussions*, 7(1), 1525–1534. <https://doi.org/10.5194/gmdd-7-1525-2014>.
13. UdeMy courses analysis. (2020, June 4). Kaggle.com; Kaggle. <http://www.kaggle.com/code/sreelathar/udemy-courses-analysis>.

**References**

1. Doh, R. F., Zhou, C., Arthur, J. K., Tawiah, I., & Doh, B. (2022). A systematic review of deep knowledge graph-based recommender systems, with focus on explainable embeddings. *Data*, 7(7), 94. <https://doi.org/10.3390/data7070094>.
2. Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1), 5–53. <https://doi.org/10.1145/963770.963772>.
3. Chen, R., Hua, Q., Chang, Y.-S., Wang, B., Zhang, L., & Kong, X. (2018). A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access: Practical Innovations, Open Solutions*, 6, 64301–64320. <https://doi.org/10.1109/access.2018.2877208>.
4. Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In *The Adaptive Web* (pp. 325–341). Springer Berlin Heidelberg.
5. Messaoudi, F., & Loukili, M. (2024). E-commerce personalized recommendations: A deep neural collaborative filtering approach. *Operations Research Forum*, 5(1). <https://doi.org/10.1007/s43069-023-00286-5>.
6. Burke, R. (2002). User Modeling and User-Adapted Interaction, 12(4), 331–370. <https://doi.org/10.1023/a:1021240730564>.
7. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2020). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, 52(1), 1–38. <https://doi.org/10.1145/3285029>.
8. Rpubs - Recommender Systems Analysis - Udemy's Recommender Engine. (n.d.). Rpubs.com. Retrieved November 6, 2024, from <https://rpubs.com/christianthieme/599308>.
9. Burke, R. (2007). Hybrid Web Recommender Systems. In *The Adaptive Web* (pp. 377–408). Springer Berlin Heidelberg.
10. Wang, X., Li, Q., Yu, D., Huang, W., & Xu, G. (2023). Neural Causal Graph Collaborative Filtering. In arXiv [cs.IR]. <https://doi.org/10.48550/ARXIV.2307.04384>.
11. Ruder, S. (2016). An overview of gradient descent optimization algorithms. In arXiv [cs.LG]. <https://doi.org/10.48550/ARXIV.1609.04747>.
12. Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? *Geoscientific Model Development Discussions*, 7(1), 1525–1534. <https://doi.org/10.5194/gmdd-7-1525-2014>.
13. Udemy courses analysis. (2020, June 4). Kaggle.com; Kaggle. <http://www.kaggle.com/code/sreelathar/udemy-courses-analysis>.