

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Міністерство освіти і науки України

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Міністерство освіти і науки України

Кваліфікаційна наукова
праця на правах рукопису

АНДРОСОВ ДМИТРО ВАСИЛЬОВИЧ

УДК 004.942:519.216.3

ДИСЕРТАЦІЯ
РЕКОМЕНДАЦІЙНА СИСТЕМА НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ ТА
ДИНАМІЧНИХ ВКЛАДЕНЬ ГРАФІВ

122 – Комп'ютерні науки
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

_____ Д. В. Андросов

Науковий керівник: Недашківська Надія Іванівна, д.т.н., доцент

АНОТАЦІЯ

Андросов Д. В. Рекомендаційна система на основі нейронних мереж та динамічних вкладень графів. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 122 Комп'ютерні науки (12 – Інформаційні технології). – Національний Технічний Університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, 2026.

Розуміння поведінки користувачів на різноманітних платформах-маркетплейсах є одним з ключових завдань для компаній, які прагнуть зберегти лояльність клієнтів та тим самим забезпечити собі стабільний дохід. Очевидно, що поведінкові шаблони користувачів формуються як на основі довгострокових індивідуальних уподобань, так і під впливом нещодавніх взаємодій користувачів з платформою. З метою моделювання та подальшого дослідження цих складних патернів використовуються рекомендаційні системи – спеціалізований клас алгоритмів статистичного навчання. Головна суть цих систем полягає в генерації персоналізованих рекомендацій – найбільш релевантних пропозицій – детально аналізуючи минулі дії, уподобання та історію взаємодій користувачів.

На практиці релевантність рекомендацій може оцінюватись за допомогою явного зворотного зв'язку, як-от рейтингів, які користувачі надають взаємодіям з контентом. Однак, коли отримання таких даних є складним або непрактичним, застосовуються неявні джерела зворотного зв'язку. До неявних сигналів належать показники активності користувачів, такі як кліки, перегляди, історія покупок, час перебування на сторінці та інші взаємодії з платформою, які дозволяють визначити приховані уподобання та наміри користувачів. Таким чином, рекомендаційні системи на основі неявного зворотного зв'язку відіграють ключову роль у прогнозуванні майбутніх дій та покращенні користувацького досвіду.

Попри суттєві досягнення у сфері глибоких нейронних мереж, ці підходи часто мають обмеження, включаючи скінченність розміру послідовностей, що подаються на вхід до мережі, та, як наслідок, схильність до рекомендації пропозицій, що є популярними на загал. Такі результати роботи рекомендаційної системи зменшують різноманітність рекомендацій, негативно впливаючи на залученість користувачів.

Щоб вирішити ці проблеми, у роботі запропоновано новий методологічний підхід, що інтегрує зовнішню реляційну інформацію у вигляді графів знань, побудованих на основі історичних даних взаємодій. Завдяки включенню графів знань у механізм багатоголової уваги, запропонована модель ефективно враховує складні взаємозв'язки та контекстну інформацію.

Метою дисертаційної роботи є підвищення точності прогнозування наступної найкращої дії користувача, одночасно забезпечуючи швидкість обчислень, необхідну для видачі рекомендацій у режимі реального часу. Заради досягнення даної мети застосовано методологію побудови рекомендаційних систем на основі моделей глибинного навчання для прогнозування послідовностей, а також методів інтелектуального аналізу даних для роботи зі зваженими графами.

У дисертації отримано такі результати, представлені нижче.

1. **Вперше запропоновано методологію** побудови гібридної рекомендаційної системи, що включає метод автоматичної генерації графів знань, метод одночасного аналізу цих графів з обробкою послідовностей поточних взаємодій користувача та платформи на основі механізмів уваги, дистиляції знань для оптимізації складності системи та метод прогнозування попиту на елементи за допомогою моделей простору станів.

2. **Вперше запропоновано метод** прогнозування попиту за допомогою спеціалізованих архітектур моделей простору станів, що ґрунтуються на операторах поліноміальної проекції високого порядку (HiPRO-операаторах) та моделюванні неоднорідних пуасонівських процесів з метою неявного моделювання графу знань, що керується динамікою цього процесу.

3. **Запропоновано метод** побудови графу знань у вигляді причинно-наслідкових зв'язків на основі статистичних метрик, що відрізняється від існуючих мінімізацією втручання особи, яка приймає рішення у процес з метою автоматизації процедури конструювання.

4. **Запропоновано модифікацію** моделі видачі послідовних рекомендацій на основі механізму уваги, що відрізняється від інших подібних архітектур можливістю одночасного аналізу послідовностей та графу знань, що робить можливим врахувати поточний контекст та довгострокові залежності, збільшуючи точність результатів у межах вирішення видачі послідовних рекомендацій.

5. **Набув подальшого розвитку метод** дистиляції знань для задачі видачі послідовних та сесійних рекомендацій з метою покращення швидкодії системи.

Практична цінність дисертаційної роботи полягає в тому, що:

1) **розроблено методологію** побудови великомасштабних рекомендаційних систем, що здатні працювати з гетерогенними даними та вирішувати задачу видачі найкращої наступної пропозиції.

2) **у межах методології** вперше запропоновані швидкодуючі гібридні рекомендаційні системи на основі дистиляції знань, а також моделей простору станів, що виконують неявне моделювання графу знань на основі даних візитів користувачів, що демонструють значний приріст у точності прогнозування наступної найкращої дії користувача, порівняно з іншими дослідженнями.

У дисертаційному дослідженні шляхом проведення низки експериментів над набором даних REES46 OpenCDP показано, що в порівнянні з бейзлайн-рішенням, що використовує моделі глибинних мереж на основі моделей LSTM з метою прогнозування послідовності взаємодії користувача з елементами, отриманими як результат застосування представленої методології моделями було досягнуто значного приросту метрик ранжування та скорочення часу видачі моделлю топ-100 найкращих пропозицій. Так, дистильована гібридна рекомендаційна система на основі модифікованого механізму уваги демонструє

приріст метрики $MAP@1$ на 169%, $MAP@10$ – на 117%, $NDCG@1$ – на 167%, а $NDCG@10$ – на 105%, при цьому час видачі топ-100 рекомендацій $ATI@100$ покращено на 10%.

У свою чергу побудована у межах дослідження гібридна рекомендаційна система на основі HiPPO-операторів демонструє покращення відносно метрики $MAP@1$ на 169%, $MAP@10$ – на 341%, $NDCG@1$ – на 217%, $NDCG@10$ – на 301%, $NDCG@10$ – 192%, при цьому час видачі топ-100 рекомендацій $ATI@100$ гірше на 14%.

В результаті проведення експериментів над популярним набором даних RetailRocket в дисертаційній роботі показано, що синтезована у результаті дослідження рекомендаційна система на основі модифікації моделей механізму уваги та моделі на основі HiPPO-операторів мають перевагу в порівнянні з моделями, що є найпопулярнішими рішеннями в індустрії – GRU4Rec, SASRec та BERT4Rec.

Зокрема, гібридна рекомендаційна система на основі модифікованого механізму уваги продемонструвала сталий приріст значень метрики $MAP@20$ – на 70% відносно GRU4Rec, і 40% відносно SASRec, та метрики $NDCG@10$ – на 72% відносно GRU4Rec, 13% – SASRec та 38% – BERT4Rec. У свою чергу, метод на основі HiPPO-операторів надає покращення $MAP@20$ – на 480% в порівнянні з GRU4Rec та 406% відносно SAS4Rec, а відносно метрики $NDCG@10$ – на 90% відносно GRU4Rec, 20% відносно SAS4Rec та 47% – відносно BERT4Rec.

Результати роботи впроваджено у навчальний процес навчально-наукового інституту прикладного системного аналізу КПІ ім. Ігоря Сікорського.

Всі теоретичні і практичні результати дисертаційної роботи у повній мірі опубліковано у фахових вітчизняних та закордонних наукових виданнях, що входять до відповідного встановленого переліку, а також виконано їх належну апробацію на міжнародних наукових конференціях і семінарах.

Дисертаційна робота виконана у відповідності до планів науково-дослідних робіт:

– НДР «Розробка моделей та методів розв’язання задач передбачення на основі великих обсягів слабкоструктурованої інформації в умовах невизначеності», №2305-ф, Державний реєстраційний номер: 0122U000671, дата реєстрації: 26-03-2025;

– НДР «Розв’язання задач передбачення з використанням методів і моделей штучного інтелекту», №2314-ф, Державний реєстраційний номер:0125U000244, дата реєстрації: 13-01-2025.

У дисертаційному дослідженні розв’язана задача підвищення точності статистичних моделей в контексті видачі послідовних рекомендацій, тобто рекомендацій найбільш релевантних для користувача наступних пропозицій, шляхом застосування сучасних методів інтелектуального аналізу даних, глибинного навчання та теорії графів до статистичних даних, представлених у вигляді гетерогенних часових рядів.

У роботі вибрано, розглянуто і описано класи рекомендаційних систем, що використані для моделювання задачі видачі найбільш релевантної пропозиції, а також виконано опис математичних моделей та підходів, що на базі статистичних даних у вигляді часових рядів використовуються для моделювання спеціального класу графів, що називаються графами знань, а також для дослідження динаміки даних часових рядів. Задля останньої задачі були досліджені нестационарні пуасонівські процеси.

У роботі досліджені часові ряди візитів користувачів до онлайн-ритейлерів, сформульовано задачу прогнозування наступної найкращої пропозиції. Однак розроблену методику можливо поширити і на інші домени, де необхідним є вирішення задачі ранжування зліченого набору предметів (товарів, послідовностей дій) за умов попередньо виконаних дій особи, що приймає рішення у межах поставленої задачі. Для прогнозування розглянуто підхід предиктивної аналітики з використанням методів інтелектуального аналізу даних та глибинного навчання.

Для первинної обробки даних і створення графів знань запропоновано підходи статистичного аналізу, а саме алгоритми асоціативних правил на основі

дерев FP-росту, TF-IDF та статистику взаємної попарної інформації (pointwise mutual information).

У роботі запропоновано методи побудови рекомендаційних систем, що полягають в автоматичному створенні графів знань, тренуванні дворівневої моделі колаборативної фільтрації та вирішенні задачі оптимізації швидкості відповіді системи за допомогою підходу дистиляції знань та моделей простору станів. Така методологія забезпечує підвищення адекватності моделі і точності прогнозів найкращої наступної пропозиції. Для оцінок якості ранжування прогнозів побудовано критеріальну базу на основі метрик MAP та NDCG. Додатково, враховуючи важливість не лише точності системи, а й швидкодії обробки нею вихідних вибірок рекомендацій, з метою швидшого надання даних рекомендацій, критеріальну базу доповнено метрикою ATI (англ. average time of inference, середній час відповіді системи).

Для конструювання нелінійної контент-орієнтованої моделі колаборативної фільтрації розглянуті підходи на базі рекурентних (RNN) нейронних мереж та механізму уваги (attention mechanism), проаналізовано їх переваги та недоліки. Для адресування проблеми вузької спеціалізації рекомендацій на основі контенту та можливості використання інформації зі сконструйованого графу знань запропоновано модифікацію механізму уваги та застосування алгоритмів поліноміальної проекції для апроксимації функцій в моделях простору станів (HiPRO-оператори).

Проведено порівняльний аналіз отриманих прогнозів з результатами використання відомих підходів та методів конструювання рекомендаційних систем колаборативної фільтрації на основі RNN. За результатами всіх практичних експериментів було показано, що застосування методології робить можливим отримувати гібридні рекомендаційні системи, що надають швидкі та точні прогнози при ненабагато більших обчислювальних витратах.

За матеріалами дисертації опубліковано 8 робіт, з яких 5 – це статті у журналах, що входять до переліку фахових видань затверджених МОН України за спеціальністю дисертації (2 з них включені до міжнародної наукометричної

бази Scopus та 2 - до Web of Science, 1 – до фахового видання категорії «Б»), 1 стаття конференції, що увійшла до іноземного періодичного видання, що індексується в наукометричній базі Scopus та 2 публікації у матеріалах конференцій (у тому числі 1 міжнародній).

Ключові слова: рекомендаційні системи, гібридні системи, графи знань, дистиляція знань, нестаціонарні процеси, часові ряди, нейронні мережі, рекурентні нейронні мережі, механізм уваги, моделі у просторі станів, апроксимаційні моделі, автокодувальники, соціальні мережі, електронна комерція.

SUMMARY

Androsov D. V. Hybrid recommendation system based on neural networks and dynamic graph embeddings.

Dissertation for the degree of Doctor of Philosophy in the field of knowledge 12 Information Technology, specialty 122 Computer Science. – National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, 2026.

Understanding user behavior on various marketplace platforms is one of the key tasks for companies seeking to maintain customer loyalty and thereby ensure stable revenue. It is evident that user behavioral patterns are formed both on the basis of long-term individual preferences and under the influence of users' recent interactions with the platform. To model and further study these complex patterns, recommendation systems are used – a specialized class of statistical learning algorithms. The main essence of these systems lies in generating personalized recommendations – that is, the most relevant offers – by thoroughly analyzing past actions, preferences, and interaction history of users.

In practice, the relevance of recommendations can be assessed through explicit feedback, such as ratings that users provide for interactions with content. However, when obtaining such data is difficult or impractical, implicit feedback sources are applied. Implicit signals include user activity indicators such as clicks, views, purchase history, time spent on page, and other platform interactions that allow determining hidden preferences and user intentions. Thus, recommendation systems based on implicit feedback play a key role in predicting future actions and improving user experience.

Despite significant achievements in the field of deep neural networks, these approaches often have limitations, including the finite size of sequences fed as input to the network and, as a consequence, a tendency to recommend offers that are generally popular. Such recommendation system results reduce the diversity of recommendations, negatively affecting user engagement.

To address these problems, this work proposes a new methodological approach that integrates external relational information in the form of knowledge graphs built based on historical interaction data. By incorporating these graphs into the multi-head attention mechanism, the proposed model effectively accounts for complex relationships and contextual information.

The goal of this dissertation work is to improve the accuracy of predicting the user's next best action while ensuring the computational speed necessary for delivering recommendations in real-time. To achieve this goal, a methodology for building recommendation systems based on deep learning models for sequence prediction is applied, as well as data mining methods for working with weighted graphs.

The dissertation obtained the following results.

1. **For the first time**, a methodology for building hybrid recommender systems is proposed, which includes a method for automatic generation of knowledge graphs, a method for joint analysis of these graphs with processing of sequences of current user–platform interactions based on attention mechanisms, a knowledge distillation method for optimizing system complexity, and a method for demand forecasting for items using state-space models.
2. **For the first time**, a method is proposed for demand forecasting using specialized state-space model architectures based on high-order polynomial projection operators (HiPPO operators) and modeling of inhomogeneous Poisson processes with the aim of implicitly modeling the knowledge graph, which is dependent on the dynamic of the given process.
3. **A method is proposed** for constructing a knowledge graph in the form of causal relationships based on statistical metrics, which differs from existing approaches by minimizing decision-maker intervention in the process in order to automate the construction procedure.
4. **A modification of a model** for generating sequential recommendations based on the attention mechanism is proposed, which differs from other similar architectures by its ability to simultaneously analyze sequences and the knowledge graph, thereby taking into account the current context and long-term dependencies and

increasing the accuracy of results in the task of generating sequential recommendations.

5. **The method of knowledge distillation** for the task of generating sequential and session-based recommendations has been further developed to improve the system's performance.

The practical value of the dissertation work lies in the fact that:

1) **a methodology has been developed** for building large-scale recommender systems capable of working with heterogeneous data and solving the task of providing the best next offer.

2) within this methodology, high-performance recommender systems based on a knowledge distillation framework **were proposed for the first time**, as well as state-space models that perform implicit knowledge graph modeling from user visit data, which demonstrate a significant increase in recommendation accuracy compared to other studies.

In the dissertation study, a series of experiments on the REES46 OpenCDP dataset showed that, compared to a baseline solution using deep neural networks based on LSTM models for predicting sequences of user–item interactions, the models developed with the proposed methodology achieved a substantial increase in ranking metrics and a reduction in the time required to generate the top-100 recommendations. Specifically, the distilled hybrid recommender system based on the modified attention mechanism yields an improvement in MAP@1 of 169%, MAP@10 of 117%, NDCG@1 of 167%, and NDCG@10 of 105%, while the time to produce the top-100 recommendations (ATI@100) is reduced by 10%. In turn, the hybrid recommender system built in the study on the basis of HiPPO operators achieves improvements of 169% in MAP@1, 341% in MAP@10, 217% in NDCG@1, 301% in NDCG@10 and 192% in NDCG@10, although ATI@100 becomes 14% worse.

As a result of experiments on the popular RetailRocket dataset, the dissertation demonstrates that the synthesized recommender system based on the modified attention-mechanism models and the model based on HiPPO operators outperforms the most widely used industry baselines – GRU4Rec, SASRec, and BERT4Rec. In

particular, the hybrid attention-based recommender system shows a consistent increase in MAP@20 of 70% relative to GRU4Rec and 40% relative to SASRec, and an improvement in NDCG@10 of 72% over GRU4Rec, 13% over SASRec, and 38% over BERT4Rec. The HiPPO-based method, in turn, provides a 480% improvement in MAP@20 compared to GRU4Rec and 406% compared to SASRec, and NDCG@10 gains of 90% over GRU4Rec, 20% over SASRec, and 47% over BERT4Rec.

The results of this work have been incorporated into the teaching process of the Educational and Research Institute for Applied System Analysis at Igor Sikorsky Kyiv Polytechnic Institute.

All theoretical and practical results of the dissertation work are fully published in professional domestic and foreign scientific publications included in the corresponding established list, and their proper validation has been performed at international scientific conferences and seminars.

The dissertation was carried out in accordance with the plans of research projects:

- Research project "Development of models and methods for solving prediction problems based on large volumes of weakly structured information under uncertainty," No. 2305-f, State registration number: 0122U000671, registration date: 26-03-2025;

- Research project "Solving prediction problems using artificial intelligence methods and models," No. 2314-f, State registration number: 0125U000244, registration date: 13-01-2025.

The dissertation research solved the problem of improving the adequacy of statistical models in the context of delivering sequential recommendations, i.e., recommendations of the most relevant next offers for the user, by applying modern methods of intelligent data analysis, deep learning, and graph theory to statistical data represented as heterogeneous time series.

The work selected, considered, and described classes of recommendation systems used for modeling the task of delivering the most relevant offer, and performed a description of mathematical models and approaches that, based on statistical data in

the form of time series, are used for modeling a special class of graphs called knowledge graphs, as well as for studying the dynamics of time series data. For the latter task, non-stationary Poisson processes were investigated.

The work investigated time series of user visits to online retailers and formulated the task of predicting the next best offer. However, the developed methodology can be extended to other domains where it is necessary to solve the problem of ranking a countable set of objects (goods, action sequences) under conditions of previously performed actions by the decision-maker within the framework of the posed task. For forecasting, a predictive analytics approach using data mining methods and deep learning is considered.

For primary data processing and knowledge graph creation, statistical analysis approaches are proposed, namely association rule algorithms, TF-IDF, and mutual pairwise information statistics.

The work proposes methods for building recommendation systems that consist of automatic knowledge graph creation, training a two-level collaborative filtering model, and solving the problem of optimizing system response speed using knowledge distillation approach and state-space models. This methodology ensures improved model adequacy and accuracy of next best offer predictions. For evaluating ranking quality of predictions, a criteria base was built based on MAP and NDCG metrics.

For constructing a nonlinear content-oriented collaborative filtering model, approaches based on recurrent (RNN) neural networks and attention mechanism are considered, and their advantages and disadvantages are analyzed. To address the problem of narrow specialization of content-based recommendations and the possibility of using information from the constructed knowledge graph, a modification of the attention mechanism and application of polynomial projection algorithms for function approximation in state-space models (HiPPO operators) is proposed.

A comparative analysis of the obtained predictions with results from using known approaches and methods for constructing collaborative filtering recommendation systems based on RNN was conducted. Based on the results of all practical experiments, it was shown that using the developed approach allows obtaining

adequate models and accurate predictions with only slightly higher computational costs.

Based on the dissertation materials, 8 works have been published, including 5 articles in journals and collections of scientific papers included in the list of professional publications approved by the Ministry of Education and Science of Ukraine for the dissertation specialty (2 of which are indexed in the Scopus international scientometric database and 2 in Web of Science, 1 – to a professional publication of category “B”), 1 conference paper published in a foreign periodical indexed in the Scopus scientometric database, and 2 publications in conference proceedings (including international ones).

Keywords: recommender systems, hybrid systems, knowledge graphs, knowledge distillation, non-stationary processes, time series, neural networks, recurrent neural networks, attention mechanism, state-space models, approximation models, autoencoders, social networks, e-commerce.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Публікації, в яких опубліковані основні наукові результати дисертації:

1. Nedashkovskaya, N. I., & Androsov, D. V. (2022). Generative time series model based on encoder-decoder architecture. *System Research and Information Technologies*, 1, 97-109. (внесок: Недашківська Н.І. – наукове керівництво та формулювання концепції; Андросов Д.В. – розробка архітектури автокодувальника, проведення експериментів, аналіз результатів)
2. Androsov, D. V. (2023). Neural ordinary differential equations for time series reconstruction. *Radio Electronics, Computer Science, Control*, 4, 69-75. (внесок: Недашківська Н.І. – наукове керівництво та формулювання концепції; Андросов Д.В. – розробка архітектури мережі на основі нейронних ЗДР, проведення експериментів, аналіз результатів)
3. Androsov, D. V., & Nedashkovskaya, N. I. (2025). The hybrid sequential recommender system synthesis method based on attention mechanism with automatic knowledge graph construction. *System Research and Information Technologies*, 1, 7-18. (внесок: Недашківська Н.І. – наукове керівництво та редагування; Андросов Д.В. – розробка методу, проведення експериментів, аналіз результатів)
4. Androsov, D. V., & Nedashkovskaya, N. I. (2025). Simple, fast and scalable recommendation systems via external knowledge distillation. *Radio Electronics, Computer Science, Control*, 3, 126-137. (внесок: Недашківська Н.І. – наукове керівництво та редагування; Андросов Д.В. – розробка методу, проведення експериментів, аналіз результатів)
5. Androsov, D. V., & Nedashkovskaya, N. I. (2025). System approach to multicriteria evaluation of session-based and sequential recommendation systems. *KPI Science News*, 4, 7-15. (внесок: Недашківська Н.І. – наукове керівництво та редагування; Андросов Д.В. – аналіз літератури, порівняння ефективності рекомендаційних систем, аналіз результатів точності рекомендаційних систем)
6. Androsov, D. (2025). Fast and scalable recommendation retrieval model with mixed attention and knowledge distillation. In V. Julian et al. (Eds.), *Intelligent*

Data Engineering and Automated Learning – IDEAL 2024 (Lecture Notes in Computer Science, Vol. 15346, pp. 244–253). Springer. (внесок: Андросов Д.В. – аналіз літератури, створення методу побудови гібридної рекомендаційної системи, проведення експериментів, аналіз результатів)

Публікації, які засвідчують апробацію матеріалів дисертації:

7. Андросов, Д. В., & Недашківська, Н. І. (2022, листопад 22–29). Порівняльний аналіз моделей генерації рекомендацій на основі методів машинного навчання на графах. У *Системні науки та інформатика: збірник доповідей I Всеукраїнської науково-практичної конференції «Системні науки та інформатика»* (с. 9–13). НН ІПСА КПІ ім. Ігоря Сікорського [Тези конференції]. (внесок: Андросов Д.В. – аналіз існуючих архітектур графових нейронних мереж в контексті рекомендаційних систем, проведення експериментів, аналіз результатів; Недашківська Н.І. – наукова редакція)

8. Androsov, D. (2023, January 24–25). Recommendation retrieval on dynamic graphs. In *Proceedings of the 2nd International Conference on Emerging Technology Trends on the Smart Industry and the Internet of Things (TTSIIT-2023)*. [Conference paper]. (внесок: Андросов Д.В. – набуття подальшого розвитку методу дистиляції знань в контексті побудови гібридних рекомендаційних систем, проведення експериментів, аналіз результатів)

ЗМІСТ

ВСТУП	18
РОЗДІЛ 1 АКТУАЛЬНІСТЬ ЗАДАЧІ ПОБУДОВИ ГІБРИДНОЇ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ ТА ОГЛЯД МЕТОДІВ ЇХНЬОГО СИНТЕЗУ	
1.1 Актуальність задачі побудови рекомендаційних систем	27
1.2 Різновиди методологій побудови рекомендаційних систем	31
1.2.1 Рекомендаційні системи на основі правил	31
1.2.2 Фільтрування на основі контенту	34
1.2.3 Рекомендаційні системи на основі знань	36
1.2.4 Рекомендаційні системи на основі контексту	39
1.3 Сучасні методології в дослідженнях поведінки користувачів.....	40
1.4 Існуючі програмні засоби для створення рекомендаційних систем ...	43
1.4.1 TensorFlow Recommenders.....	44
1.4.2 PyTorch TorchRec	45
Висновки до розділу 1	47
РОЗДІЛ 2 АНАЛІЗ МЕТОДІВ РОЗВ’ЯЗАННЯ ЗАДАЧ КОЛАБОРАТИВНОЇ ФІЛЬТРАЦІЇ ТА ВИДАЧІ ПОСЛІДОВНИХ РЕКОМЕНДАЦІЙ.....	
2.1 Постановка задачі колаборативної фільтрації.....	51
2.1.1 Метод змінних найменших квадратів	52
2.1.2 Байєсівське персоналізоване ранжування.....	54
2.2 Вирішення задачі колаборативної фільтрації за допомогою глибинних нейронних мереж.....	57
2.2.1 Постановка задачі нейронної колаборативної фільтрації	57
2.2.2 Найпопулярніші моделі глибинного навчання, що використовуються в задачах колаборативної фільтрації.....	59
2.3 Постановка задачі видачі послідовних рекомендацій	64
2.4 Рекурентні нейронні мережі в контексті задачі видачі послідовних рекомендацій.....	65

2.5 Системи на основі механізму уваги.....	68
Висновки до розділу 2.....	70
РОЗДІЛ 3 МЕТОДОЛОГІЯ ПОБУДОВИ ГІБРИДНОЇ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ ВИДАЧІ ПЕРСОНАЛІЗОВАНИХ РЕКОМЕНДАЦІЙ	76
3.1 Метод побудови графу знань для представлення сесій користувачів	78
3.2 Алгоритми навчання представлень вершин графів	84
3.3 Методи побудови нейронних мереж для обробки графових даних	87
3.3.1 Метод на основі модифікації моделей згорткових мереж	87
3.3.2 Метод на основі модифікацій моделей механізму уваги	88
3.4 Подальший розвиток методу дистиляції знань для рекомендаційних систем	93
3.5 Аналіз методів на основі моделей нейронних диференціальних рівнянь	98
3.6 Метод на основі моделей простору станів.....	101
3.7 Метрики оцінювання якості прогнозів рекомендаційної системи	110
Висновки до розділу 3.....	113
РОЗДІЛ 4 ЕКСПЕРИМЕНТАЛЬНИЙ АНАЛІЗ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ ТА ДИНАМІЧНИХ ВКЛАДЕНЬ ГРАФІВ	118
4.1 Дворівнева (two-tower) архітектура рекомендаційних систем	118
4.2 Набори даних (датасети) для експериментів	119
4.2.1 Набір даних REES46 Open CDP.....	119
4.2.2 Набір даних RetailRocket	120
4.3 Конструкція графа знань на основі даних взаємодій (сесій).....	121
4.4 Дослідження методу дистиляції знань для покращення швидкості відповіді моделей	124
4.5 Аналіз моделей простору станів для задачі видачі послідовних рекомендацій.....	130
Висновки до розділу 4.....	133

ВИСНОВКИ	135
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	142
ДОДАТОК А СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ	152
ДОДАТОК Б АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНОГО ДОСЛІДЖЕННЯ.....	154

ВСТУП

Актуальність роботи. Розуміння поведінки користувачів на різноманітних платформах-маркетплейсах є одним з ключових завдань для компаній, які прагнуть зберегти лояльність клієнтів та тим самим забезпечити собі стабільний дохід. Рекомендаційні системи – спеціалізований клас алгоритмів статистичного навчання, що моделюють складні патерни поведінки користувачів, сформовані як на основі довгострокових індивідуальних уподобань, так і під впливом нещодавніх взаємодій з платформою. Головна суть цих систем полягає в генерації персоналізованих рекомендацій – тобто найбільш релевантних пропозицій – детально аналізуючи минулі дії, уподобання та історію взаємодій користувачів.

На практиці релевантність рекомендацій може оцінюватись за допомогою явного зворотного зв'язку, як-от рейтингів, які користувачі надають взаємодіям з контентом. Однак, коли отримання таких даних є складним або непрактичним, застосовуються неявні джерела зворотного зв'язку. До неявних сигналів належать показники активності користувачів, такі як кліки, перегляди, історія покупок, час перебування на сторінці та інші взаємодії з платформою, які дозволяють визначити приховані уподобання та наміри користувачів.

Попри суттєві досягнення у сфері глибоких нейронних мереж, ці підходи часто мають обмеження, включаючи скінченність розміру послідовностей, що подаються на вхід до мережі, та, як наслідок, схильність до рекомендації пропозицій, що є популярними на загал. Такі результати роботи рекомендаційної системи зменшують різноманітність рекомендацій, негативно впливаючи на залученість користувачів. На даному етапі підходи до розв'язання поставленої задачі методами інтелектуального аналізу даних знаходяться в стані активного дослідження, причому загальноприйнятої методики не існує.

Мета і задачі наукового дослідження. Метою дисертаційної роботи є підвищення точності прогнозування наступної найкращої дії користувача, за рахунок поєднання моделей глибинного навчання для аналізу послідовностей та

методів інтелектуального аналізу зважених динамічних графів, одночасно забезпечуючи швидкість обчислень, необхідну для видачі рекомендацій у режимі, наближеному до реального часу. Важливість задоволення обидвох складових мети прямо впливає з прикладного характеру задачі синтезу та впровадження рекомендаційних систем. Заради досягнення даної мети запропонована методологія побудови рекомендаційних систем на основі моделей глибинного навчання для прогнозування послідовностей, а також методів інтелектуального аналізу даних для роботи з зваженими графами, що мають темпоральну природу генезу.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

- виконати аналіз існуючих методів побудови рекомендаційних систем, навести їх недоліки в контексті задачі видачі персоналізованих рекомендацій;
- сформулювати задачу прогнозування найбільш релевантних пропозицій для користувача, вибрати типи взаємодій для дослідження, зібрати статистичні дані про поведінку користувачів;
- сформулювати і описати методологію та її основні етапи для побудови гібридної рекомендаційної системи на основі генерації графу знань та аналізу цього графу з обробкою послідовностей поточних взаємодій користувача та платформи;
- запропонувати шляхи подальшого удосконалення існуючих моделей та методів за допомогою їх комбінування з графовими нейронними мережами, механізмами уваги, моделями у просторі станів;
- запропонувати та використати нові методи інтелектуального аналізу даних стосовно розв'язання поставлених задач, включаючи динамічні графові вкладення та дистиляцію знань;
- застосувати методику на реальних даних користувацьких взаємодій шляхом реалізації програмного продукту, виконати експериментальне дослідження та провести порівняльний аналіз отриманих прогнозів з результатами відомих моделей РС.

Об'єкт дослідження: послідовності взаємодій користувачів з асортиментами офлайн- та онлайн-платформ, що надають товари та послуги, а також їхні властивості.

Предмет дослідження: методи побудови рекомендаційних систем на основі динамічних графових вкладень, механізмів уваги, нейронних диференціальних рівнянь та моделей простору станів.

Методи дослідження: математична статистика, чисельні методи, методи оптимізації та дослідження операцій, методи інтелектуального аналізу даних, теорія графів, методи глибинного навчання з використанням архітектур на основі механізму уваги, методи дистиляції знань.

У ході дослідження використовувалась мова програмування Python, фреймворк для чисельних методів оптимізації та нейронних мереж TensorFlow, бібліотека для роботи з графовими структурами даних NetworkX.

Наукова новизна роботи. Наукова новизна даної дисертаційної роботи доведена такими інноваціями:

- вперше запропоновано методологію побудови гібридної рекомендаційної системи, що включає метод автоматичної генерації графу знань, метод одночасного аналізу цих графів з обробкою послідовностей поточних взаємодій користувача та платформи на основі механізмів уваги, дистиляцію знань для оптимізації складності системи та метод прогнозування попиту на елементи за допомогою моделей простору станів, за допомогою яких досягнуто суттєвого підвищення точності виданих рекомендацій наступної найкращої пропозиції при цьому мінімізуючи час роботи отриманих рекомендаційних систем;

- запропоновано метод побудови графу знань у вигляді причинно-наслідкових зв'язків на основі статистичних метрик, що відрізняється від існуючих мінімізацією втручання особи, що приймає рішення у процес з метою автоматизації процедури конструювання;

- вперше запропоновано метод прогнозування попиту за допомогою спеціалізованих архітектур моделей простору станів, що ґрунтуються на

операторах поліноміальної проекції високого порядку (HiPPO-операторах) та моделюванні неоднорідних пуасонівських процесів з метою неявного моделювання графу знань, що диктується даними процесами;

- запропоновано модифікацію моделі видачі послідовних рекомендацій на основі механізму уваги, що відрізняється від інших подібних архітектур можливістю одночасного аналізу послідовностей та графу знань, що надає можливість враховувати поточний контекст та довгострокові залежності, збільшуючи точність результатів у межах вирішення видачі послідовних рекомендацій;

- набув подальшого розвитку метод дистиляції знань для задачі видачі послідовних та сесійних рекомендацій, де був застосований з метою покращення швидкодії системи та спрощення її архітектури.

Обґрунтованість і достовірність наукових результатів забезпечується коректним застосуванням математичного апарату теорії графів, механізмів уваги та нейронних диференціальних рівнянь, а також відповідних обчислювальних експериментів з використанням сучасного програмного забезпечення. Забезпечується також використанням методів інтелектуального аналізу даних, математичної статистики, методів аналізу послідовностей взаємодій, а також процедурами контролю точності проміжних та остаточних результатів, що оцінювались на незалежних тестових вибірках реальних даних електронної комерції.

Практичне значення отриманих результатів. В результаті дослідження здобувачем було запропоновано методологію побудови рекомендаційних систем, що дозволяють мінімізувати вплив людського фактора та автоматизувати процес моделювання допоміжних структур, що називаються графами знань з метою покращення точності вирішення задачі інформованого пошуку, до якої зокрема відносяться рекомендаційні системи. Здобувачем створено швидкісну та масштабовану модель для вирішення задач рекомендації наступної найкращої пропозиції, в основу якої покладено запропоновані здобувачем механізми модифікації моделей механізму ваги, що набули назви

змішаної уваги, та дистиляції знань. Практична значущість моделі полягає у забезпеченні високої швидкості генерації кандидатів на наступну найкращу пропозицію при збереженні високої точності. Так шляхом проведення низки експериментів над набором даних REES46 OpenCDP показано, що в порівнянні з бейзлайн-рішенням, що використовує моделі глибинних мереж на основі моделей LSTM з метою прогнозування послідовності взаємодії користувача з елементами, отриманими як результат застосування представленої методології моделями було досягнуто значного приросту метрик ранжування та скорочення часу видачі моделлю топ-100 найкращих пропозицій. Так, дистильована гібридна рекомендаційна система на основі модифікованого механізму уваги демонструє приріст метрики $MAP@1$ на 169%, $MAP@10$ – на 117%, $NDCG@1$ – на 167%, а $NDCG@10$ – на 105%, при цьому час видачі топ-100 рекомендацій $ATI@100$ покращено на 10%. У свою чергу побудована у межах дослідження гібридна рекомендаційна система на основі HiPRO-операторів демонструє покращення відносно метрики $MAP@1$ на 169%, $MAP@10$ – на 341%, $NDCG@1$ – на 217%, $NDCG@10$ – на 301%, $NDCG@10$ – 192%, при цьому час видачі топ-100 рекомендацій $ATI@100$ гірше на 14%.

Результати роботи впроваджено у навчальний процес наавчально-наукового інституту прикладного системного аналізу КПІ ім. Ігоря Сікорського.

Дисертаційна робота виконана у відповідності до планів науково-дослідних робіт:

- НДР «Розробка моделей та методів розв’язання задач передбачення на основі великих обсягів слабкоструктурованої інформації в умовах невизначеності», №2305-ф, Державний реєстраційний номер: 0122U000671, дата реєстрації: 26-03-2025;

- НДР «Розв’язання задач передбачення з використанням методів і моделей штучного інтелекту», №2314-ф, Державний реєстраційний номер: 0125U000244, дата реєстрації: 13-01-2025.

Всі результати дисертаційного дослідження доведено до високого практичного інженерного рівня та можуть бути впроваджені у компаніях

електронної комерції з метою покращення процесів генерації персоналізованих рекомендацій та підвищення залученості користувачів. Створено систему підтримки прийняття рішень для видачі рекомендацій в режимі реального часу, що побудована на основі розробленої методології з використанням динамічних графових вкладень та механізмів багатоголової уваги. За допомогою розробленої системи і методики спеціалісти можуть будувати моделі і прогнози для рекомендаційних систем на основі аналізу поведінкових даних користувачів.

Всі теоретичні і практичні результати дисертаційної роботи у повній мірі опубліковано у фахових вітчизняних та закордонних наукових виданнях, що входять до відповідного встановленого переліку, а також виконано їх належну апробацію на міжнародних наукових конференціях і семінарах.

Особистий внесок здобувача. Усі основні результати дисертаційної роботи отримані автором особисто. Усі шість фахових публікацій написано у співавторстві, а здобувачеві належать такі результати, представлені нижче.

1. У роботі [1] здобувачем запропоновано генеративну модель часових рядів на основі архітектури кодер-декодер з використанням варіаційних автокодерів та рекурентних нейронних мереж. Розроблена модель поєднує переваги варіаційного автокодера, мереж GRU та елементи звичайних диференціальних рівнянь для ефективного моделювання нелінійних нестационарних процесів. Запропоновано систему, що мінімізує вплив людського фактора при аналізі часових рядів та забезпечує високорівневий інтерфейс для швидкого конструювання глибоких моделей.

2. У роботі [2] здобувачем запропоновано метод синтезу глибокої нейронної мережі для розв'язання задач реконструкції вхідного сигналу та екстраполяції часових рядів з використанням нейронних звичайних диференціальних рівнянь. Розроблена модель Latent Ordinary Differential Equations Variational Autoencoder (LODE-VAE) демонструє видатну продуктивність при вирішенні задач реконструкції часових рядів за наявності незначного відсотка зашумлених та нерівномірно розподілених вхідних сигналів. Проведені експерименти підтвердили ефективність запропонованого

методу для розв'язання практичних задач, що потребують відновлення динаміки нестационарних процесів.

3. У роботі [3] здобувачем запропоновано новий метод синтезу гібридної послідовної рекомендаційної системи, що базується на механізмах уваги з автоматичним конструюванням графів знань. Розроблений підхід поєднує переваги глибокого навчання з представленням знань у вигляді графових структур для підвищення якості персоналізованих рекомендацій. Сформульована методологія для інтеграції зовнішньої реляційної інформації через механізм багатоголової уваги, що ефективно враховує складні взаємозв'язки та контекстну інформацію.

4. У роботі [4] здобувачем запропоновано швидкісну та масштабовану модель для задач рекомендаційного пошуку, що базується на комбінованих механізмах уваги та дистиляції знань. Розроблена модель поєднує переваги декодерних архітектур з навчанням на графах знань для підвищення якості прогнозування через явне збагачення пулу кандидатів рекомендацій. Запропоновано двобаштову архітектуру глибокої нейронної мережі, що забезпечує високу швидкість виведення та виняткову точність при великомасштабній обробці користувацьких запитів.

5. У роботі [5] здобувачем розроблено метод створення простих, швидких та масштабованих рекомендаційних систем через зовнішню дистиляцію знань. Запропонований підхід використовує комбінацію архітектур глибокого навчання на основі механізмів уваги з застосуванням графів знань для підвищення якості прогнозування. Побудовано систему для прогнозування найбільш релевантних пропозицій для користувача, що включає запропоновану модель та метод її навчання з використанням метрик MAP@k та NDCG@k для оцінки якості рекомендацій. Проведені експерименти на великих обсягах реальних даних від відвідувань онлайн-магазину підтвердили, що запропонований метод гарантує високу релевантність видаваних рекомендацій, є швидким та невибагливим до обчислювальних ресурсів.

6. У роботі [6] здобувачем застосовано системний підхід до багатокритеріального оцінювання різних моделей послідовних рекомендаційних систем, зокрема систем, що базуються на сесійних даних – проведений структурний аналіз літератури щодо побудови таких моделей, присвячено окрему увагу до задачі вибору найкращої моделі та окреслені перспективи майбутніх напрямків досліджень у рекомендаційних системах. Здобувачем виконано систематичний огляд і всебічний аналіз сесійних і послідовних рекомендаційних систем, їх теоретичних основ, еволюції, емпіричних характеристик продуктивності та аспектів практичного розгортання.

Апробація результатів дисертації. Результати та основні положення роботи подавалися та обговорювалися у межах заходів, представлених нижче.

1. І Всеукраїнська науково-практична конференція «Системні науки та інформатика», 22–29 листопада 2022 року, Київ [7].

На даній конференції здобувачем розглянуті основні проблеми імплементації рекомендаційних систем та основні обмеження існуючих підходів. Здобувач виконав порівняння існуючих моделей рекомендаційних систем колаборативної фільтрації, систем на основі нейронних мереж для обробки послідовностей та графових нейронних мереж.

2. The 2nd International Conference on Emerging Technology Trends on the Smart Industry and the Internet of Things “TTSIIT-2023”, (online), January 24-25th 2023 [8].

Здобувачем шляхом проведення експериментальних досліджень була встановлена ефективність застосування динамічних графів у якості основної структури даних, що використовується для навчання рекомендаційних систем. Експериментальним шляхом доведено, що використання графових структур даних для пошуку найбільш пов'язаних елементів з поточними в сесії користувача, значно підвищують точність рекомендаційної системи. Дослідження набуло подальшого розвитку в роботі [3].

3. IDEAL 2024: 25th International Conference, Valencia, Spain, November 20-22nd 2024 [4].

Здобувач вперше продемонстрував ефективність застосування методів дистиляції знань в контексті побудови рекомендаційних систем для вирішення задачі видачі послідовних персоналізованих рекомендацій. Експериментальні дослідження показали незначні втрати точності дистильованої моделі в порівнянні з недистильованими моделями, що використовують графи знань та запропонований здобувачем у роботах [3, 4] методом побудови РС на основі механізму змішаної уваги, при значному прирості швидкодії та спрощенні моделі. Окрім того, отримана дистильована модель має значно кращу точність за бейзлайн-рішення, що використовує мережі довгої короткострокової пам'яті, тим самим доводячи перевагу методу дистиляції знань відносно навчання моделі «з нуля».

Публікації. За матеріалами дисертації опубліковано вісім робіт, п'ять з яких – статті у журналах і збірниках наукових праць, що входять до переліку фахових видань затверджених МОН України за спеціальністю дисертації або у періодичних іноземних виданнях (дві з них включені до міжнародної наукометричної бази Scopus та дві – до Web of Science, одна – до фахового видання категорії «Б»), одна стаття конференції, що увійшла до іноземного періодичного видання, що індексується в наукометричній базі Scopus та дві публікації у матеріалах конференцій (у тому числі, одній міжнародній).

РОЗДІЛ 1 АКТУАЛЬНІСТЬ ЗАДАЧІ ПОБУДОВИ ГІБРИДНОЇ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ ТА ОГЛЯД МЕТОДІВ ЇХНЬОГО СИНТЕЗУ

1.1 Актуальність задачі побудови рекомендаційних систем

Світові процеси діджиталізації створили безпрецедентні виклики в галузі управління інформацією та прийняття рішень користувачами. Як підкреслюється в останніх дослідженнях [10-12] сучасні користувачі стикаються з величезним асортиментом товарів та послуг на різних платформах, від електронних торгових майданчиків (e-commerce) до стрімінгових сервісів. Рекомендаційні системи вирішують цю фундаментальну проблему, виступаючи в ролі інтелектуальних механізмів фільтрації, що зменшують когнітивне навантаження та покращують якість рішень.

Найперші рекомендаційні системи використовували базові підходи на основі колабортивної фільтрації та рекомендацій на основі контенту, що з плином часу заслужили репутацію надійних рішень, та досі широко використовуються в прототипуванні рішень на основі даних [10-12].

Сучасні системи у свою чергу пройшли еволюцію від базових статистичних алгоритмів до застосування передових методів машинного навчання, а саме глибинного навчання, графових нейронних мереж та дедалі більшого впровадження гібридних методів, що закликані вирішити недоліки кожного вищезгаданого методу [4-6, 10-12].

Еволюція алгоритмів рекомендаційних систем насамперед продиктована зростаючою кількістю бізнес-вимог. Сучасні підприємства потребують систем, що здатні обробляти гетерогенні типи даних, а також видавати результати в режимі реального часу, при цьому надаючи обґрунтовані рекомендації та адаптуючись до дедалі зростаючих потреб користувачів.

Саме вищезгадані критерії і слугують каталізатором переходу від систем, що базуються на правилах, заданих експертами галузі, чи добуті статистичними

методами, до сучасних систем, що здатні працювати з мультимодальними даними та забезпечувати пошук неочевидних патернів [10, 12].

Практично встановлено, що бізнеси, впроваджуючи рекомендаційні системи, значно поліпшують ключові показники ефективності, такі як коефіцієнт кліків (click-through rate, CTR), коефіцієнт конверсії (conversion rate, CVR) та валову вартість товарів (gross merchandise value, GMV) [11]. Ці показники залучення клієнтів довгостроково впливають на конкурентоздатність бізнесу, а тому сам факт впровадження рекомендаційних систем є стратегічною інвестицією.

Платформи електронної комерції (e-commerce) є найбільш розповсюдженою, та історично найпершою, сферою застосування рекомендаційних систем. Такі платформи, як Amazon, Alibaba та eBay впродовж десятиліть демонструють зростаючі доходи та завдячують великій базі постійних користувачів саме рекомендаційним системам [10, 12].

Системи, що впроваджують вказані вище компанії використовують гібридні підходи, що поєднують колаборативну фільтрацію, контент-аналіз та контекстну інформацію для надання персоналізованих релевантних пропозицій. Серед основних застосувань рекомендаційних систем в електронній комерції можна виокремити такі:

- крос-продажі – пошук товарів, покупка яких користувачем є найбільш ймовірною при умові, що користувач купує цільовий продукт, тим самим збільшуючи середній чек покупця, при цьому утримуючи їх на платформі покупок;
- персоналізований пошук – платформи впорядковують елементи (товари) по спаданню їхньої релевантності, утримуючи увагу користувачів;
- динамічне ціноутворення – дані поточного попиту на товари та послуги використовуються для корекції ціни.

Індустрія відео- та аудіостримінгу також широко використовує рекомендаційні системи для вирішення проблем видачі релевантного контенту. Такі платформи, як Netflix [11, 12], Spotify [12, 13] та YouTube [10-12, 14],

інвестували значні кошти у складні алгоритми, що аналізують поведінку користувачів, особливості шуканого контенту та контекстні фактори (наприклад, зовнішньо- та внутрішньополітичні чинники) для надання персоналізованих рекомендацій щодо контенту [10-14].

Netflix використовує дані щодо локації для видачі адаптованого пулу рекомендацій одразу на головній сторінці, щоб моментально залучити користувача до перегляду (рис. 1.1).

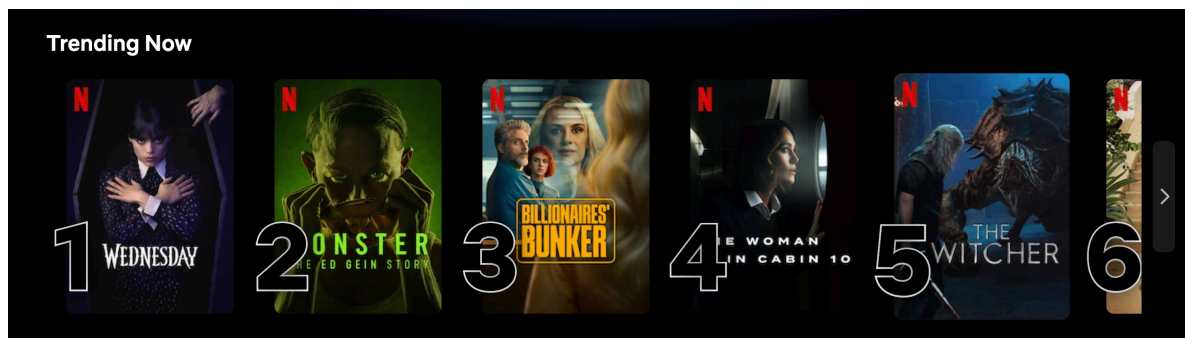


Рисунок 1.1 – Секція «головна сторінка Netflix»

Основні сфери застосування рекомендаційних систем для даного домену включають у себе курацію контенту, оптимізацію залучення користувачів та аналіз привабливості інвестицій в контент [11].

Для вирішення задачі управління контенту (content curation) рекомендаційні системи ранжують контент для автоматичного створення плейлистів та упорядкування контенту на основі уподобань користувачів (релевантності).

Оптимізацію залучення користувачів рекомендаційні системи вирішують завдяки постійній видачі релевантних пропозицій, досягаючи максимізації ключових показників залучення користувачів, таких як час перегляду, тривалість сеансу та частота повторних візитів.

Деякі відомі платформи, такі як YouTube [14] та TikTok [10, 11, 15] використовують результати роботи рекомендаційної системи для прийняття

рішень щодо монетизації контенту, аналізуючи уподобання користувачів (рис. 1.2).

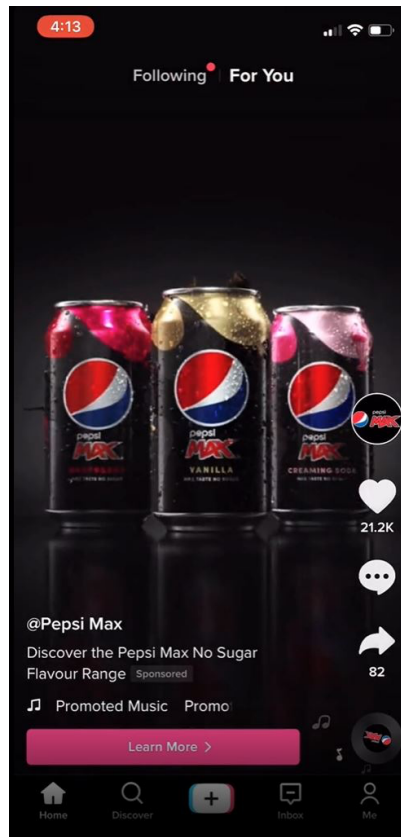


Рисунок 1.2 – Приклад реклами в TikTok [15]

Сектор фінансових послуг також все частіше застосовує рекомендаційні системи для різних цілей, від інвестиційних порад до рекомендацій щодо кредитування. Останні дослідження [10-12, 16] демонструють застосування рекомендаційних систем у фінансовому домені, використовуючи дані транзакцій за кредитними картками для прогнозування наступних покупок та пропозицій відповідних фінансових послуг.

Рекомендаційні системи все частіше застосовуються для створення ботів-консультантів, що пропонують інвестиційні портфелі на основі толерантності користувача до ризику, його фінансових цілей, стану ринку та інших зовнішніх чинників. Сучасні банки впроваджують системи, які рекомендують відповідні

фінансові інструменти на основі профілів клієнтів, історії транзакцій та кредитоспроможності [11].

Хоча в наявній науковій літературі про рекомендаційні системи серед прикладів впровадження переважають онлайн-додатки, «фізичні» мережі роздрібної торгівлі також використовують дані системи, наприклад розміщуючи продукти на полицях таким чином, щоб максимізувати середній чек покупця.

Сучасна роздрібна торгівля все більше поєднує онлайн- та офлайн-методи, надаючи через додатки персоналізовані пропозиції на основі поточного місцезнаходження, попередніх покупок та наявності товарів на складі магазинів поруч в режимі реального часу [11].

1.2 Різновиди методологій побудови рекомендаційних систем

Історично формувалися різні підходи для розв’язання задач видачі персоналізованих рекомендацій. Серед усталеної практики виокремлюють підходи на основі правил [11, 17], контент-фільтрацію [10-12, 18, 19], підходи на основі знань [11, 19, 26] та контекстні системи [11, 21].

1.2.1 Рекомендаційні системи на основі правил

Рекомендаційні системи на основі правил є найдавнішим систематичним підходом до автоматизованого формування рекомендацій. Ці системи кодують доменні знання та бізнес-логіку у вигляді чітких правил, які визначають пропозиції товарів на основі спостережуваних характеристик або поведінки користувачів. Останні дослідження демонструють, що підходи на основі правил продовжують відігравати важливу роль у сучасних системах, особливо для очищення рекомендацій від «шуму», тобто похибок моделі, та застосувань принципів обґрунтованого штучного інтелекту (explainable AI) [11, 17].

Рекомендаційні системи на основі правил досі являють собою один з найбільш найрозповсюджених підходів в індустрії, завдяки своїй надійності та

використанню експертно-заданих правил та евристик. На відміну від колаборативної фільтрації або систем, базованих на контенті, що розглянуті в наступних пунктах цього розділу, рекомендаційні системи на основі правил функціонують на основі прозорих, зрозумілих процесів прийняття рішень, які можуть бути безпосередньо перевірені та модифіковані експертами предметної області, а не спираються на неявні патерни, вилучені з історичних даних.

Основні компоненти системи на основі правил охоплюють модуль збору та підготовки даних, модуль добування правил, сховище правил та метаданих, модуль відбору та агрегації правил під контекст користувача, а також, наостанок, модуль генерації рекомендацій [11].

Першим кроком є збір історичних взаємодій користувачів з системою: перегляди, покупки, кліки, послідовності відвіданих сторінок, рейтинги тощо. На цьому етапі формуються транзакції (кошики покупок, сесії навігації, послідовності подій), застосовується очищення, відсів аномалій в даних, наприклад, неправильно записаних чи пошкоджених транзакцій, а також вилучення з вибірки даних,. Важливість цього етапу полягає у тому, наскільки репрезентативними та інтерпретованими будуть подальші правила [11, 12].

Далі запускається модуль видобування закономірностей, де застосовуються алгоритми пошуку частих патернів та асоціативних правил. Асоціативні правила зазвичай відповідають такому шаблону: «Користувачі, які взаємодіють з елементами A і B, ймовірно, будуть зацікавлені в елементі C», що формалізується як:

$$A \wedge B \rightarrow C_{supp=s, conf=c}.$$

де підтримка (supp) вимірює частоту правила в даних, а впевненість (conf) вимірює вірогідність імплікації, тобто відносну частоту даного причинно-наслідкового зв'язку при умові істинності імпліканти правила. Серед найпоширеніших алгоритмів асоціативних правил виділяють Apriori, дерева FP-

росту та Eclat [3, 4, 10-12, 17]. Цей підхід виявляється особливо ефективним для задачі крос-продажів в електронній комерції [11, 12].

Сучасні реалізації даної методології еволюціонували від статичних правил, записаних в певну базу даних до їхньої динамічної адаптації. Серед таких прикладів можна привести RuleAgent [17], що використовують великі мовні моделі виявлення правил, імітуючи поведінку людських експертів та адаптуючись до зміни поведінки інших користувачів.

Здобуті правила зберігаються у репозиторії правил, який, крім самих патернів, містить значення метрик support, confidence, lift, інформацію про часову валідність – тобто коли правило було визначене – а також можливі бізнес-обмеження.

У момент формування рекомендації система повинна обрати релевантні правила під поточний контекст (історія користувача, активна сесія, пристрій, локація тощо). Після відбору та агрегації система формує список рекомендацій. На даному, заключному, етапі кожному можливому рекомендованому елементу присвоюється підсумковий бал (score), що враховує силу й кількість активних правил. Надалі, після присвоєння даної числової оцінки релевантності елементу, вони сортуються за цим балом, додаються механізми відсікання вже придбаних товарів, застосовуються додаткові бізнес-обмеження.

Найпростіший варіант реалізації таких рекомендаційних систем складається з вручну відібраних правил, де експерти предметної області явно визначають умови, за яких слід рекомендувати предмети [10, 11]. Наприклад, правило може визначати, що «якщо користувач придбав товари в категорії А та переглянув товари в категорії Б, рекомендуйте найбільш оцінені товари з категорії Б».

Особливо успішним є підхід на основі інтеграції асоціативних правил з сучасними факторизаційними машинами для створення гібридних систем (FMAR) [16], які зберігають інтерпретованість, одночасно досягаючи підвищеної точності.

Незважаючи на появу підходів на основі нейронних мереж та методів матричної факторизації, рекомендаційні системи на основі правил вражають своєю обчислювальною ефективністю, оскільки оцінка правил зазвичай вимагає логарифмічної або лінійної часової складності відносно кількості правил та товарів, різко контрастує з квадратичною або вищою складністю багатьох нейромережових підходів.

1.2.2 Фільтрування на основі контенту

Основною перевагою рекомендаційних систем фільтрації на основі контенту є їх здатність функціонувати в умовах, де статистичні методи навчання виявляються неефективними або застосування їхнє є неможливим.

У контексті холодного старту (cold start problem), коли новий користувач щойно приєднався до системи або новий товар щойно був доданий до каталогу, контент-базовані методи можуть генерувати рекомендації на основі доступних метаданих та описів [10, 11, 18], тоді як статистичні підходи (наприклад, колаборативна фільтрація) потребують накопичення мінімальної кількості взаємодій [11, 12]. Крім того, такі рекомендаційні системи не потребують збору великих матриць користувачів-товарів з історичних даних, що є їхньою безумовною перевагою в умовах роботи з розрідженими даними або обмеженими ресурсами для збереження таких даних.

Проте РС на основі фільтрації контенту мають суттєві обмеження, які обмежують їхню застосованість у бізнесі. Основною проблемою є критична залежність від повноти метаданих та описів контенту. У випадках, коли опис об'єкта недостатньо інформативний або неточний, система не може сформувати адекватне представлення об'єкта, що призводить до неточних рекомендацій.

Фільтрування на основі контенту рекомендує елементи (товари, послуги), порівнюючи характеристики елементів із профілями уподобань користувачів. Цей підхід вирішує проблему холодного старту для нових елементів, при цьому

забезпечуючи обґрунтованість рекомендацій за допомогою аналізу схожості ознак [18].

Методологія фільтрування на основі контенту вимагає реалізації таких етапів:

- кодування елементів – інтерпретація описових атрибутів елементів у якості числових векторів ознак [12];
- моделювання профілю користувача – створення моделей даних уподобань користувача на основі історичних взаємодій та явного зворотного зв'язку, що також представляються числовим вектором [10-12];
- обчислення метрики дистанції між атрибутами профілів користувачів та характеристиками елементів [11];
- генерація рекомендацій шляхом ранжування елементів на основі отриманих значень [11, 12].

Як видно з опису методології, рекомендаційні системи на основі контенту вимагають попередньої обробки ознак з метою синтезу векторів-представлень елементів та користувачів. Найпоширеніші підходи до моделювання таких векторів зазвичай охоплюють обробку текстових ознак методом TF-IDF, використання one-hot кодування для дискретних (скінченних та злічених) множин атрибутів, а також стандартизацію неперервних атрибутів [12].

Профілі користувачів у системах на основі контенту змінюються з плином часу, «еволюціонуючи» завдяки зворотному зв'язку від взаємодії з попередніми рекомендаціями та асортиментом в загальному. Для моделювання профілів як векторів використовують дві техніки:

- зважена агрегація ознак – поєднання векторів елементів, з якими у користувача була взаємодія;
- використання окремого класифікатора взаємодії користувача з елементами методами статистичного навчання.

Рекомендаційні системи на основі контенту страждають від надмірної спеціалізації (обмеженості прогнозів), оскільки профіль користувача формується на основі нещодавно відвіданих ним елементів [12].

Цей ефект особливо негативний для задач видачі рекомендацій для особливих користувачів, для яких важливими є нішеві, менш популярні об'єкти, які мають високу вартість для певних користувачів.

Проте, незважаючи на вищезазначене, важко переоцінити важливість фільтрації на основі контенту. Особливо враховуючи поширення мультимодальних систем, що обробляють текст, зображення, аудіо та метаданих одночасно [3, 4, 9-12, 14], все частіше для обробки частини цієї інформації використовуються дані підходи.

Крім того, у галузях, де цілісність та приватність даних є беззаперечною вимогою (наприклад, у медичних системах) [11], такі методи набувають переваги завдяки відсутності потреби у аналізі даних про поведінку інших користувачів.

1.2.3 Рекомендаційні системи на основі знань

Рекомендаційні системи на основі знань є результатом еволюції РС на основі правил, що використовують відомості про домен (галузь застосування) для генерації рекомендацій, що є особливо цінним в тих ситуаціях, де уподобання користувачів важко визначити на основі даних про їхню поведінку в силу малих вибірок [12, 19]. Такі РС, на відміну від попередніх систем, які оперують безпосередньо з описовими атрибутами об'єктів, використовують структуровані представлення знань, такі як онтології, графи знань та семантичні мережі, для кодування складних взаємозв'язків між сутностями, концепціями та властивостями. Ці системи побудовані на припущенні, що явне моделювання семантичного змісту та причинних зв'язків надає можливість краще зрозуміти поведінку користувачів та виокремити найбільш значущі властивості товарів [12].

Беззаперечною перевагою систем на основі знань є їхня інтерпретованість. Ця властивість гарантується явно закодованими семантичними структурами, що надає можливість чітко встановити причини, по яким конкретний товар був

рекомендований, посиляючись на специфічні властивості товару та явні правила виведення.

Підхід на основі знань характеризується трьома ключовими компонентами [12]:

- знання про товари – дескриптивні характеристики, атрибути та взаємозв'язки продуктів або послуг;
- знання про уподобання користувачів – явне представлення вимог, обмежень та уподобань користувачів;
- галузеві знання про те, які товари відповідають конкретним вимогам користувачів.

Одним з основних класів рекомендаційних систем на основі знань є системи на основі обмежень [11, 12]. Дані системи моделюють рекомендації як задачу оптимізації з обмеженнями, де останніми виступають формалізовані вимоги користувачів. Цей підхід виявляється особливо ефективним для областей, таких як фінансові послуги, нерухомість та конфігурація технічних продуктів.

Процес видачі рекомендацій на основі обмежень зазвичай складається з таких кроків:

- уточнення вимог – проведення аналізу уподобань користувачів, наприклад шляхом опитування;
- формалізація обмежень – задання функцій на домені дійсних чисел, що визначають чутливість користувача до вимоги, впливаючи на рекомендацію послуги або елемента; наприклад, для обачного інвестора функція обмежень може бути пропорційною квадрату ризиковості потенційного портфеля, а для необачного – логарифму цієї ризиковості;
- пошук простору рішень шляхом вирішення задачі оптимізації за умов обмежень;
- ранжування отриманих результатів, що належать простору рішень, відповідно до метрики вподобань користувачів.

Основною перешкодою для широкого впровадження рекомендаційних систем на основі обмежень є процес здобуття знань (knowledge acquisition) [20, 21].

Побудова нових та підтримка існуючих семантичних структур знань є трудомістким та дорогим процесом, що вимагає залучення експертів предметної області для явного задання онтології, побудови графу знань, тощо. У галузях, що динамічно розвиваються, або при роботі з великими каталогами елементів (товарів, послуг), ручна побудова та оновлення графу знань стає неймовірно складною. Крім того, системи на основі знань часто демонструють меншу точність порівняно з колаборативною фільтрацією у випадках, де є наявність великих обсягів статистичних даних про взаємодії користувачів, оскільки згадані вище підходи на рівні правил не можуть встановити всі неявні закономірності та нюанси, які можна вилучити з поведінкових даних.

Для вирішення труднощів, пов'язаних із здобуттям знань, запропоновано кілька підходів, наприклад використання ймовірнісних баз даних, що здатні «вбудовувати» логіку обмежень в сам запит, видаючи результати, що належать простору рішень [22]. Фреймворки м'яких обмежень (soft constraint) вирішують проблему задоволення обмежень з оптимізацією уподобань [23, 24], наприклад зменшуючи кількість обмежень шляхом усування недосяжних з них, або введення нечітких обмежень, де кожному з обмежень присвоюється функція належності [25].

Іншим поширеним класом систем на основі знань є рекомендаційні системи на основі випадків (case-based recommendations) [12]. Дані системи послуговуються методологією прийняття рішень на основі схожості, рекомендуючи елементи, схожі на ті, які раніше сподобалися користувачеві, або елементи, схожі на явно задані експертом приклади. Цей підхід виявляється особливо ефективним, коли користувачі можуть навести конкретні приклади своїх вподобань [12, 26].

Загалом, РС на основі знань є перспективним напрямком. Поширення корпусів знань, таких як DBpedia, Wikidata та Freebase, які кодують

структуровану інформацію про мільйони сутностей та мільярди фактів, відкриває нові можливості для побудови масштабних семантичних структур даних. Крім того, розвиток мультимодальних РС – поєднання текстових описів, зображень, аудіо та структурованих атрибутів – демонструє, як і у випадку з системами фільтрації контенту, що системи на основі знань можуть застосовуватися до кожного з типів атрибутів окремо, ефективно звужуючи простір пошуку найкращих пропозицій.

1.2.4 Рекомендаційні системи на основі контексту

Рекомендаційні системи з урахуванням контексту ґрунтуються на гіпотезі, що уподобання користувачів значно варіюються в залежності від зовнішніх факторів, таких як поточний час, місце перебування користувача, характеристики пристрою, тощо [11, 12]. Як встановлено в межах проекту CARS [27], контекстна інформація істотно покращує якість рекомендацій, дозволяючи персоналізацію з урахуванням конкретної ситуації.

Інформація щодо контексту вбудовується в логіку видачі рекомендацій трьома способами:

- попереднє фільтрування, де контекстуальні обмеження, тобто інформація про стан середовища, в якому знаходиться користувач, звужують множину елементів перед застосуванням власне рекомендаційної системи;
- пост-фільтрування, де знання про контекст впливає на ранжування або рекомендацій після етапу їх генерації системою;
- контекстуальне моделювання, тобто використання інформації про контекст як одну зі змінних моделі, що обирає елементи-кандидати для рекомендацій.

Часово-орієнтовані, або темпоральні фактори, є найбільш розповсюдженою формою контексту, оскільки уподобання користувачів інтерпретують як певний часовий ряд, що демонструє одночасно короткострокову (щоденні закономірності, сезонні коливання, ефекти реклами

найпопулярніших товарів), та довгострокову динаміки (поступова зміна інтересів користувачів) [12, 25, 26].

Іншим поширеним підвидом контексту є просторовий, що став активно використовуватися з появою геолокаційних сервісів. Просторовий контекст охоплює:

- семантичну класифікацію місцезнаходження, де система присвоює певний характеристичний клас до місцезнаходжень (наприклад, дім, робота, розважальні заклади);
- розрахунок географічної близькості – система рекомендує товари та послуги, що знаходяться в прийнятній відстані від поточного місцезнаходження користувача;
- дослідження патернів переміщення, де рекомендаційна система використовує спостереження за переміщеннями користувачів для прогнозування майбутніх місцезнаходжень та застосування цієї інформації для видачі рекомендацій.

Соціально-демографічний контекст у свою чергу охоплює міжособистісні та демографічні фактори, що впливають на уподобання користувачів, на кшталт соціальних зв'язків, належності користувача до груп користувачів із потенційно схожими уподобаннями, а також стратифікацію користувачів за когортами віку, культурного походження та інших демографічних факторів [11, 12].

1.3 Сучасні методології в дослідженнях поведінки користувачів

Сучасні рекомендаційні системи все частіше послуговуються підходами глибинного навчання, особливо в контексті задачі видачі послідовних рекомендацій [28].

Для мети аналізу послідовностей рекомендаційні системи використовують рекурентні нейронні мережі, в особливості архітектури LSTM та GRU, що моделюють еволюцію уподобань користувачів у часі [3, 4, 10-12]; механізми уваги, що фіксують як короткострокові, так і довгострокові залежності в

послідовностях взаємодії користувачів та сервісу [4-6, 10-12, 29]; згорткові нейронні мережі, що зазвичай використовуються для вирішення задач комп'ютерного зору, однак демонструють свою ефективність при обробці послідовностей, особливо такі модифікації як Conv-LSTM [10-12], що використовують згорткові мережі для локального згладжування послідовностей безпосередньо перед обробкою рекурентною мережею.

Підходи глибинного навчання також використовуються для виявлення патернів у взаємодіях користувачів з конкретними елементами та у представленнях характеристик користувачів та елементів. Для останньої задачі активно використовуються архітектури автокодувальників – клас мереж, що реалізують стратегію навчання без вчителя для вирішення задачі зменшення розмірності та отримання числових представлень ознак у контексті видачі рекомендацій. Також, в РС є непоодиноким використання «класичних» архітектур багат шарових перцептронів [3, 4, 11, 12, 30-32], задля моделювання складних взаємодій між користувачами та елементами, які становлять альтернативу підходам простої матричної факторизації, що будуть розглянуті детальніше в наступних розділах.

Сучасні рекомендаційні системи дедалі частіше використовують гібридні підходи, що поєднують кілька методів для досягнення задовільної точності отриманої системи. Поширені стратегії створення гібридних систем охоплюють:

- лінійну або нелінійну зважені комбінації виходів моделей (прогнозів) з подальшим ранжуванням [3-6, 12, 27-29];
- перемикання між моделями: динамічний вибір між різними методологіями, що приймається на основі доступності даних, характеристик користувачів або зовнішнього контексту.

Отже, в цій роботі надалі розглянуто саме методологія побудови гібридної рекомендаційної системи в контексті задачі видачі послідовних рекомендацій.

Задачі видачі послідовних рекомендацій, зокрема задача генерації наступної найкращої пропозиції (next best offer, NBO) [3-5, 9, 27] та

прогнозування наступного кошика (next basket prediction, NBP) [29], постали як критично важливі компоненти в сучасних бізнес системах.

На відміну від традиційних підходів до створення рекомендаційних систем, що фокусуються на сталих уподобаннях користувачів, системи NBO явно моделюють часову динаміку та послідовні залежності в поведінці користувачів для прогнозування майбутніх дій та намірів щодо покупок [16, 28, 29].

Актуальність задачі NBO підтверджується емпірично впливом релевантних рекомендацій на ключові показники ефективності бізнесу, включаючи коефіцієнти конверсії, середню вартість замовлення та загальну кількість виручки з клієнта за увесь час (lifetime value, LTV) [10-12]. Як задокументовано в провідних дослідженнях, системи видачі послідовних рекомендацій демонструють значно кращі значення метрики середньої точності прогнозу (mean average precision, MAP) при порівнянні з традиційними підходами [10-12, 16, 23, 27-31].

Сучасні рекомендаційні системи NBO та NBP все більше покладаються на методології глибокого навчання для моделювання послідовностей. Серед останніх напрацювань можна відзначити фреймворк SEQNBT, що поєднує архітектуру автокодувальників та використовує рекурентні нейронні мережі GRU для вирішення задачі прогнозування наповнення наступного кошика користувача [16].

Впровадження таких систем в продакшн-середовище вимагає вирішення таких задач:

- досягнення мінімізації затримки. РС мають взаємодіяти з серверною частиною онлайн сервісів та передавати дані про рекомендації у реальному часі клієнту, тобто надавати відповідь на запити протягом мілісекунд; серед прикладів подолання цієї труднощі можна навести систему iPrescribe, що поєднує в собі ефективну інфраструктуру комунікації сервісів та технології потокової обробки даних [29];

- підтримання актуальності моделі: РС, що вирішують задачі NBO/NBP є системами на основі контенту, а тому є необхідним постійне оновлення та навчання моделей на свіжих даних з метою охоплення нового асортименту та підлаштування до нових патернів поведінки користувачів [27-29].

- забезпечення масштабування: з плином часу змінюється не лише поведінка існуючих користувачів, а й їх кількість, а також розширюється асортимент товарів та послуг, що платформа може надавати своїй аудиторії [29-31].

Сучасні рекомендаційні системи успішно вирішують задачі NBO та NBP, гарантуючи масштабованість, своєчасну обробку інформації про взаємодію користувача з асортиментом платформи та розв'язання задачі холодного старту завдяки поєднанню різних методів [3-5, 10-12, 27-29] .

Серед останніх надбань можна привести інтеграцію в інструментарій побудов послідовних рекомендаційних систем архітектур трансформерних моделей, рекурентних нейронних мереж, методів на основі аналізу частоти, алгоритмів синтезу асоціативних правил та можливостей навчання на основі експертних знань, що покращує процес обґрунтування відповідей цих систем.

1.4 Існуючі програмні засоби для створення рекомендаційних систем

Сучасні рекомендаційні системи здатні обробляти мільярди взаємодій користувачів і надавати персоналізовані рекомендації в режимі, наближеному до реального часу. Традиційні фреймворки машинного навчання, такі як Scikit-Learn, часто не мають спеціалізованих компонентів, необхідних для ефективного вирішення задач видачі персоналізованих рекомендацій.

До таких засобів належать, зокрема, модулі та програмні компоненти для ефективного зберігання та оновлення ознак товарів і користувачів, розподілених

операцій на великих розріджених наборах даних взаємодій, а також вбудованої підтримки специфічних для рекомендацій метрик якості.

Крім того, вони зазвичай не враховують вимоги до інтеграції з продакшн-сервісами, що є критично важливими для промислових систем рекомендацій. Це призвело до розробки спеціально створених фреймворків та програмних рішень, які орієнтовані на виклики великомасштабних рекомендаційних систем [30-32].

1.4.1 TensorFlow Recommenders

TensorFlow Recommenders (TFRS) – це відкрита бібліотека Google, побудована на TensorFlow 2.x для створення комплексних рекомендаційних систем [30, 33]. TFRS надає високорівневі абстракції, які об'єднують завдання пошуку та ранжування в межах єдиного фреймворку.

TFRS імплементує інструменти створення РС, надаючи розробнику доступ до таких абстракцій:

- програмні класи для створення моделей користувачів та товарів. Фреймворк прискорює розробку компонентів, що оброблюють дані про користувачів та товари/елементи з навчальних наборів даних, шляхом інкапсуляції логіки ініціалізації вагів моделей та алгоритму зворотного поширення похибок на основі бібліотеки Keras;
- вбудована логіка «завдань». TFRS надає змогу користувачеві швидко прототипізувати логіку, за якою працюватиме рекомендаційна система, задаючи завдання. Розробники фреймворку послуговуються парадигмою дворівневих рекомендаційних систем, що вирішують дві задачі – ефективний пошук кандидатів (retrieval) та їхнє ранжування (ranking);
- імплементовані метрики ранжування. Моделі TFRS сумісні зі всіма метриками, що доступні в екосистемі TensorFlow 2.x, зокрема TFRanking, що спрощує прототипування моделей;

- нативна інтеграція в продакшн-середовища. Оскільки моделі TFRS працюють на графах обчислень TensorFlow, вони легко суміщаються з спеціальними компонентами екосистеми TensorFlow для оцінки та моніторингу моделей TFX, бібліотекою автоматизованого створення API моделей TensorFlow Serving та фреймворком портування моделей на пристрої з обмеженими ресурсами, наприклад, телефони - TensorFlow Lite;
- інструменти MLOps інтеграції. TensorFlow нативно підтримується Google Cloud Platform, що надає змогу без зайвих зусиль масштабувати ML-системи, використовуючи такі сервіси як VertexAI та Kubeflow (рис. 1.3).

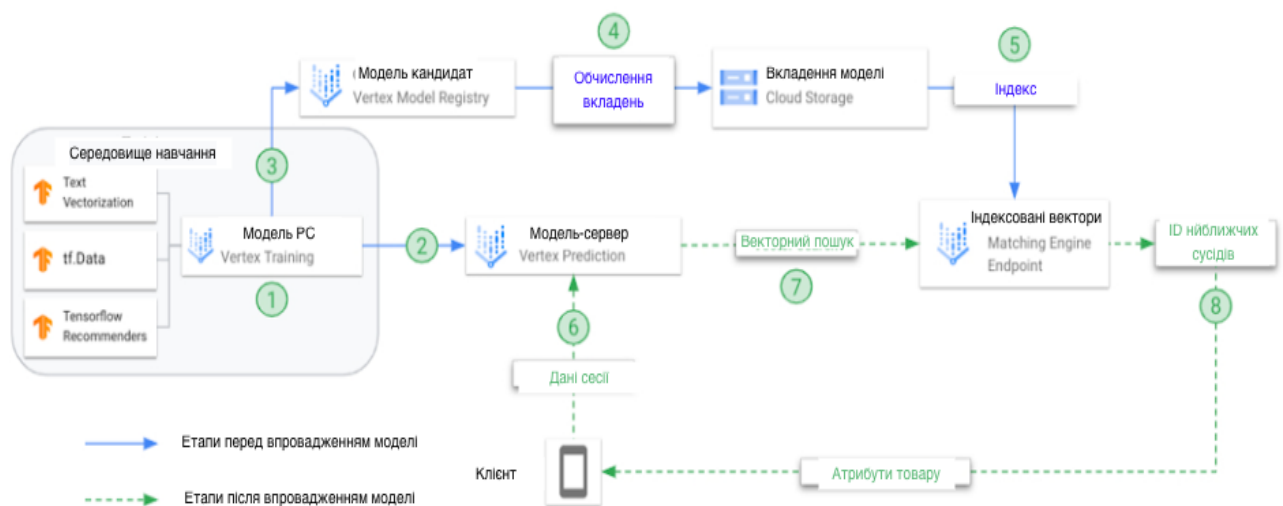


Рисунок 1.3 – Приклад MLOps інтеграції PC на основі TFRS в середовищі Google Cloud [33]

1.4.2 PyTorch TorchRec

TorchRec – це бібліотека Meta на основі PyTorch, спеціально розроблена для рекомендаційних продакшн-систем [31, 35]. На відміну від фреймворків загального призначення, на кшталт Torch Lightning, TorchRec фокусується виключно на прототипуванні рекомендаційних систем у продакшн-середовищах.

TorchRec вирішує кілька ключових задач проектування і підтримки РС в високонавантажених середовищах:

- забезпечення можливості розподіленого навчання. Оскільки TorchRec заснований на фреймворку PyTorch, то в ньому реалізована можливість розподіленої навчання параметрів на кількох графічних процесорах (GPU);
- оптимізація пошуку векторів-вкладень. TorchRec реалізує розподілення операцій пошуку елементів по таблицях вкладень (ембедингів, embeddings) по кільком GPU (рис. 1.4);
- нативна підтримка квантизованих обчислень. TorchRec використовує бібліотеку FBGEMM (Facebook general matrix multiplication), що робить можливим виконувати швидке множення матриць за рахунок зниження точності чисельних представлень елементів (квантизації);
- інтеграція з Hugging Face, що надає можливість використовувати попередньо навчені моделі для вбудовування в рекомендаційні системи;
- підтримка динамічних обчислювальних графів. PyTorch використовує динамічний обчислювальний граф (eager execution), що робить можливим змінювати архітектуру мережі під час викликів моделей. Це робить код більш інтуїтивним та схожим на звичайний ООП-код на Python.

Обидва інструменти широко використовуються в академічних дослідженнях та продакшн-системах та є майже взаємозамінними. В дослідженні для проведення експериментів використано саме фреймворк TensorFlow Recommenders, оскільки тренування моделі відбувалося в середовищі Google Cloud Platform.

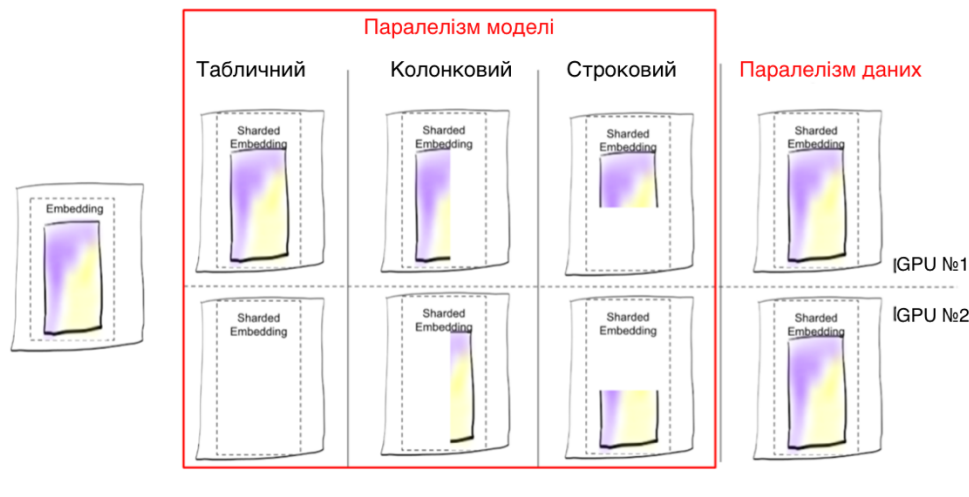


Рисунок 1.4 – Стратегії розподілених обчислень векторів-вкладень в TorchRec [35]

Висновки до розділу 1

Галузь рекомендаційних систем за останні десятиліття перейшла від переважно теоретичних досліджень до повноцінної інтеграції в бізнес-процеси компаній. Докази з нещодавніх досліджень та впроваджень свідчать, що рекомендаційні системи є особливо цінними, оскільки збільшують середній чек покупця та створюють базу лояльних користувачів, що мінімізує довгострокові ризики зменшення прибутку. Системи видачі персоналізованих рекомендацій демонструють позитивний вплив на показники утримання користувачів та зниження коефіцієнта відтоку.

Також варто підкреслити, що на практиці впровадження рекомендаційних систем призводить до збільшення часу, проведеного користувачем на платформі, та зростання частоти повторних відвідувань, що прямо корелює з приростом виручки та покращенням фінансових показників компанії.

Цифрові та офлайн бізнеси по-різному впроваджують рекомендаційні технології відповідно до наявних викликів та ресурсів. Інтернет-компанії мають доступ до детальної аналітики користувачів, включаючи дані взаємодій, часові мітки, геолокаційну інформацію та контекстуальні дані, та можуть реагувати на

зміни в поведінці користувачів у реальному часі, адаптуючи алгоритми. Цей постійний процес адаптації надає змогу платформам перебувати в стані рівноваги між максимізацією релевантності рекомендацій та виявленням нових цікавих товарів для користувачів.

Офлайн-бізнеси у свою чергу стикаються з суттєвими обмеженнями щодо наявності та якості інформації про клієнтів, що передбачає розробку спеціалізованих методів, спроможних виділяти поведінкові патерни при розріджених даних та обмежених можливостях збору інформації.

У контексті традиційної роздрібно́ї торгівлі рекомендаційні системи часто реалізуються через програми лояльності та картки клієнта, що забезпечують необхідну видимість у поведінку покупців.

Методології побудови рекомендаційних систем еволюціонували від базових алгоритмів до складних систем, що використовують алгоритми статистичного та глибинного навчання. Розвиток рекомендаційних систем від простих підходів на основі правил до складних сучасних мультимодальних систем є прямим наслідком прогресу обчислювальної техніки, що робить можливим досліджувати складні поведінкові патерни користувачів та виокремлювати з даних мотивацію того, чи іншого рішення цільового клієнта.

Серед множин розглянутих рекомендаційних систем, особливе місце посідають системи на основі правил, що забезпечують високий рівень інтерпретованості, дозволяючи експертам предметної області явно кодувати бізнес-логіку. Цей клас систем залишається критично важливим у галузях, таких як фінанси та медицина, де є строга вимога дотримання приватності даних, а тому унеможливорює використання підходів на основі статистичного навчання.

Далі були розглянуті моделі на основі фільтрації контенту. У межах дослідження було наведено, що дані підходи вирішують проблему холодного старту та демонструють ефективність у ситуаціях нестачі історичних даних про взаємодії користувачів, що є особливо цінним при запуску нових платформ або введенні «свіжих» товарів у каталог.

У свою чергу досліджено як системи на основі знань кодують семантичні взаємозв'язки через структуровані представлення, такі як графи знань та ймовірнісні бази даних, дозволяючи системам розуміти не лише явно задані характеристики товарів, але й глибші семантичні та категоріальні зв'язки.

Гібридні архітектури, що поєднують множину методів, виникли насамперед з метою подолання обмежень окремих підходів та максимізації отриманих переваг кожного з методів побудови рекомендаційних систем. Такі системи, наприклад, можуть використовувати моделі фільтрацій на основі контенту як компоненти для вирішення проблеми холодного старту, РС на основі правил та випадків для забезпечення інтерпретованості отриманих рекомендацій, а також використовувати рекомендаційні системи на основі знань як додаткові модулі для забезпечення якнайширшого семантичного розуміння домену, доповнюючи все вищезазначене нейромережевими архітектурами з метою виокремлення складних поведінкових патернів з великомасштабних наборів даних. Комбінування цих підходів надає змогу системам балансувати між точністю прогнозування, інтерпретованістю результатів та досягнення компромісу щодо обчислювальної складності системи.

Дослідження, проаналізовані в межах цієї дисертації, показують, що гібридні системи часто демонструють вищу якість рекомендацій порівняно з розрізненними підходами, особливо у тих випадках, де дані користувачів та товарів є різноманітними (мультимодальними) та мають множину вимірів для опису.

Це дослідження зосереджується на розв'язанні основних недоліків наявних рекомендаційних систем та пропонує нову методологію, що поєднує переваги методів на основі знань, контенту та правил в єдиній моделі. Запропонований підхід передбачає структурування представлень знань з автоматичним видобуванням заданої структури даних, забезпечує спроможність системи працювати з гетерогенними типами даних та забезпечувати інтерпретовані рекомендації, які користувачі та експерти можуть легко розуміти та пояснювати.

Експериментальна валідація запропонованого підходу проводиться на реальних наборах даних з домену електронної комерції для демонстрування точності та практичної обґрунтованості методології.

Для експериментів використано мову програмування Python та фреймворк TensorFlow Recommenders – один із найпоширеніших інструментів побудови моделей глибинного навчання для видачі персоналізованих рекомендацій.

РОЗДІЛ 2 АНАЛІЗ МЕТОДІВ РОЗВ'ЯЗАННЯ ЗАДАЧ КОЛАБОРАТИВНОЇ ФІЛЬТРАЦІЇ ТА ВИДАЧІ ПОСЛІДОВНИХ РЕКОМЕНДАЦІЙ

2.1 Постановка задачі колаборативної фільтрації

Колаборативна фільтрація (КФ) являє собою фундаментальну методологію в рекомендаційних системах, оскільки використовує припущення, що елементи, котрі користуються попитом певного користувача, є доцільним рекомендувати іншим користувачам, схожим на вказаного [3-12, 36, 37].

В основі алгоритмів КФ стоїть аналіз певної матриці спостережень (рейтингів, уподобань, тощо) $R \in \mathbb{R}^{m \times n}$, де m позначає кількість користувачів, а n позначає кількість елементів, з елементами r_{ui} , що представляють оцінку або взаємодію користувача u з елементом i [36, 37]. Дана матриця завжди є некоректно визначеною, оскільки за відсутності взаємодії користувача та елемента елементу r_{ui} присвоюється довільне значення, що не ґрунтується на спостереженнях.

Основне припущення, що лежить в основі усіх алгоритмів КФ, полягає в тому, що ця матриця взаємодій є розрідженою та має низький ранг, що надає можливість виражати метрику вподобання користувачами елемента на основі комбінацій вподобань інших користувачів, а також виділяти латентні змінні-предиктори [10-12, 36].

Для заданої матриці R з множиною спостережень $\Omega \subseteq \{1, \dots, m\} \times \{1, \dots, n\}$, задача колаборативної фільтрації полягає у знаходженні «повної» (в сенсі наявності рейтингів товарів, з якими користувачі ще не взаємодіяли) матриці $\hat{R}$ такої, що:

$$\min_{\hat{R}} \sum_{(u,i) \in \Omega} \mathcal{L}(r_{ui}, \hat{r}_{ui}) + \lambda \cdot \Psi(\hat{R}), \quad (2.1)$$

де $\mathcal{L}$ є функцією втрат, Ψ є регуляризатором, а λ контрольє силу регуляризації [38].

Припущення про низький ранг постулює, що $rk(R) \ll \min(m, n)$, що робить можливим факторизацію $R \approx UV^T$, де $U \in R^{m \times k}$ та $V \in R^{n \times k}$ з умовою $k \ll \min(m, n)$ [39].

Класичний підхід матричної факторизації апроксимує рейтингову матрицю через декомпозицію латентних факторів:

$$\widehat{r_{ui}} = u_u^T v_i + b_u + b_i + \mu, \quad (2.2)$$

де $u_u \in R^k$ та $v_i \in R^k$ є латентними векторами користувача та елемента відповідно, b_u, b_i є зміщеннями, а μ є ковзним середнім [40].

Стандартна оптимізаційна задача (2.1) для матричної факторизації з регуляризацією за нормою Фробеніуса набуває вигляду:

$$\min_{U, V} \sum_{(u,i) \in \Omega} (r_{ui} - u_u^T v_i)^2 + \lambda_u |U|_F^2 + \lambda_v |V|_F^2, \quad (2.3)$$

де $|\cdot|_F$ позначає норму Фробеніуса [41].

Існує множина алгоритмів, що вирішують дану задачу, серед найпопулярніших – розклад матриці на власні (сингулярні) значення (SVD) [3-6, 10-12, 36, 37], метод змінних найменших квадратів (ALS) [3-6, 10-12, 38-40, 42-46], байєсівське персоналізоване ранжування (BPR) [3-6, 10-12, 37, 43, 44, 46-49] та градієнтні методи [3-6, 10-12, 37, 40-42, 50-52].

2.1.1 Метод змінних найменших квадратів

Метод змінних найменших квадратів став одним із найбільш широко застосовуваних алгоритмів оптимізації для матричної факторизації в системах колаборативної фільтрації [42, 43, 46]. Популярність алгоритму впливає з його

простоти, можливості паралелізації та численних експериментальних підтверджень точності та надійності.

Фундаментальна ідея ЗНК полягає в перетворенні неопуклої задачі матричної факторизації на послідовність задач опуклої оптимізації, що ітеративно вирішуються методом найменших квадратів [42, 43, 46].

Нехай для частково спостережуваної матриці рейтингів користувач-елемент $R \in R^{m \times n}$, постає необхідним знайти матриці $U \in R^{m \times k}$ та $V \in R^{n \times k}$ такі, що $R \approx UV^T$, де $k \ll \min(m, n)$ представляє розмірність «прихованого» простору ознак (latent space) [44].

ЗНК вирішує задачу (2.3) ітеративно методами опуклої оптимізації за допомогою двох підзадач:

$$\begin{aligned} U^{(t+1)} &= \operatorname{argmin}_U \mathcal{L}(U, V^{(t)}), \\ V^{(t+1)} &= \operatorname{argmin}_V \mathcal{L}(U^{(t+1)}, V). \end{aligned}$$

Для фіксованої V , оптимальний вектор користувача u_i визначається як:

$$u_i = (V_i^T V_i + \lambda_u I)^{-1} V_i^T r_i,$$

де V_i містить рядки V , що відповідають елементам, оціненим користувачем i , а r_i є вектором рейтингів користувача i . Аналогічно, для фіксованої U :

$$v_j = (U_j^T U_j + \lambda_v I)^{-1} U_j^T c_j,$$

де U_j містить рядки U , що відповідають користувачам, які оцінили елемент j .

Оскільки кожен етап алгоритму ЗНК виконується для визначення однієї матриці, його можна прискорити за допомогою паралельних обчислень для кожного з векторів-стовпців, що, наприклад, реалізовано фреймворком паралельних обчислень Apache Spark [42, 45].

Належна регуляризація є критично важливою для алгоритму ЗНК, оскільки на кожному етапі задача матричної декомпозиції, що вирішується алгоритмом найменших квадратів, є погано-зумовленою, з чого випливає існування сценаріїв, коли рішення не існує, або навпаки кількість розв'язків є нескінченною, особливо в умовах розріджених даних. Окрім стандартної регуляризації L_2 автори адаптації алгоритму ЗНК до домену задач КФ з неявним відгуком – тобто таким, що не надає інформації про ступінь вподобання користувачем елемента, а власне лише «повідомляє» систему, що факт взаємодії був – вводять спеціальну матрицю C , що моделює «впевненість» в тому, що взаємодія корелює з вподобаннями [44]:

$$c_{ui} = 1 + \alpha r_{ui},$$

де α – певний зважуючий гіперпараметр. Отже, таким чином задача (2.3) перетворюється на

$$\min_{U,V} \sum_{(u,i) \in \Omega} c_{ui} (1[r_{ui} > 1] - u_u^T v_i)^2 + \lambda_u |U|_F^2 + \lambda_v |V|_F^2.$$

2.1.2 Байєсівське персоналізоване ранжування

Байєсівське персоналізоване ранжування (БПР) є новішим алгоритмом, що опирається на іншу методологію ранжування елементів, не шляхом точкової оцінки рейтингів елементів, якою є значення, здобуті за МНК чи ЗНК, до моделювання ймовірнісного розподілу власне відношень між елементами [3, 4, 10-12, 46-49]. Цей підхід виявився особливо ефективним для вирішення задачі ранжування елементів за умови наявності лише «неявного» рейтингу, де уподобання користувача задані виключно через поведінкові сигнали, такі як кліки, покупки або перегляди [10, 12].

Основна ідея БПР, як видно з описаного вище, полягає в моделюванні відносних уподобань між спостережуваними та неспостережуваними

взаємодіями користувача з елементом. Дана парадигма стала основою значної кількості сучасних алгоритмів РС [47, 48].

Нехай для множини користувачів $\mathcal{U}$ та елементів $\mathcal{I}$, «неявний» зворотний зв'язок (рейтинг) представляється як множина спостережуваних взаємодій $\mathcal{S} \subseteq \mathcal{U} \times \mathcal{I}$. Для кожного користувача u визначаємо як множину елементів, з якими користувач u взаємодіяв:

$$\mathcal{I}^+(u) := \{i: (u, i) \in \mathcal{S}\}.$$

Для кожного користувача u , спостережувані взаємодії передбачають введення відношення строгого порядку $>_u$ таке, що $i >_u j$ для всіх $i \in \mathcal{I}^+(u)$ та $j \in \mathcal{I} \setminus \mathcal{I}^+(u)$. Тобто, в основі алгоритму БПР існує припущення, що елементи, з якими взаємодіяв користувач, є більш популярними, аніж ті, досвід взаємодії з якими у користувача відсутній з невідомих причин [46-49].

Критерій оптимізації БПР впливає наслідком застосування байєсівського статистичного виводу задля вирішення проблеми персоналізованого ранжування. Нехай Θ представляє параметри функції рейтингів елементів $f(\cdot | \Theta): \mathcal{I} \times \mathcal{U} \rightarrow \mathcal{R}$.

Таким чином, вводиться апостеріорна ймовірність параметрів за умови уподобань користувача:

$$p(\Theta | >_u) \propto p(>_u | \Theta)p(\Theta). \quad (2.4)$$

Нехай є істинним припущення незалежності попарної взаємодії користувачів та елементів. Тоді є можливою побудова функції правдоподібності набуває вигляду:

$$p(>_u | \Theta) = \prod_{(u,i,j) \in \mathcal{D}_S} p(i >_u j | \Theta),$$

де $\mathcal{D}_S = \{(u, i, j): i \in \mathcal{I}^+(u), j \in \mathcal{I} \setminus \mathcal{I}^+(u)\}$.

Критерій оптимізації БПР максимізує логарифм апостеріорної ймовірності (2.4) [47]:

$$\ln p(\Theta | \mathcal{D}_S) = \sum_{(u,i,j) \in \mathcal{D}_S} \ln \sigma(\widehat{x}_{uij}) - \lambda_\Theta |\Theta|^2, \quad (2.5)$$

де $\sigma(x) = \frac{1}{1+e^{-x}}$ є сигмоїдною функцією, $\widehat{x}_{uij} = \widehat{x}_{ui} - \widehat{x}_{uj}$ представляє (попарну) різницю уподобань, а λ_Θ є параметром регуляризації.

Підсумовуючи, відповідна функція втрат для мінімізації (2.5):

$$\mathcal{L}_{BPR} = - \sum_{(u,i,j) \in \mathcal{D}_S} \ln \sigma(\widehat{x}_{uij}) + \lambda_\Theta |\Theta|^2.$$

Функція втрат оптимізуються за допомогою стохастичного градієнтного спуску з бутстреп-вибіркою навчальних трійок (u, i, j) . Градієнт відносно параметрів моделі:

$$\frac{\partial \mathcal{L}_{BPR}}{\partial \Theta} = - \sum_{(u,i,j) \in \mathcal{D}_S} (1 - \sigma(\widehat{x}_{uij})) \frac{\partial \widehat{x}_{uij}}{\partial \Theta} + \lambda_\Theta \Theta.$$

Для моделей матричної факторизації, $\widehat{x}_{ui} = u_u^T v_i$, де $u_u \in R^k$ та $v_i \in R^k$ є векторами латентних факторів. Оновлення параметрів набувають вигляду:

$$\begin{aligned} \frac{\partial \widehat{x}_{uij}}{\partial w_{uf}} &= h_{if} - h_{jf}, \\ \frac{\partial \widehat{x}_{uij}}{\partial h_{if}} &= w_{uf}, \\ \frac{\partial \widehat{x}_{uij}}{\partial h_{jf}} &= -w_{uf}. \end{aligned}$$

де w_{uf} та h_{if} позначають f -ті компоненти векторів користувача та елемента відповідно [47].

2.2 Вирішення задачі колаборативної фільтрації за допомогою глибинних нейронних мереж

У зв'язку з розвитком обчислювальної техніки та можливістю використання складніших алгоритмів статистичного навчання на відносно великих вибірках все більшої популярності набуває впровадження методології глибинного навчання до вирішення задачі видачі рекомендацій.

Серед наявних підходів та алгоритмів варто зазначити алгоритми нейронної колаборативної фільтрації (НКФ) [49]. Фундаментальна відмінність НКФ полягає в тому, що задачу (2.1) замінюють задачею «навчання» матриць користувачів та товарів, що надає змогу моделювати відгуки користувачів на спожиті послуги як нелінійні оператори [5-9, 38, 50]. Цей перехід від лінійного до нелінійного моделювання породив численні архітектури РС, від простих багат шарових перцептронів для отримання оцінки ймовірності взаємодії [50, 51] до складних алгоритмів на основі механізмів уваги [52-54], варіаційних автокодувальників [55, 56], тощо.

2.2.1 Постановка задачі нейронної колаборативної фільтрації

Нейронна колаборативна фільтрація (НКФ) представляє парадигмальний зсув у рекомендаційних системах, замінюючи традиційну лінійну матричну факторизацію глибокими нейронними архітектурами, здатними навчатися складних нелінійних взаємодій користувач-елемент [49].

Основна особливість рекомендаційних систем, що імплементують методи НКФ полягає в заміні операції внутрішнього добутку матричної факторизації навчальними нейронними функціями, що дає змогу моделювати складні та приховані патерни колаборативної фільтрації, які лінійні методи не можуть охопити.

Метод НКФ усуває ключові обмеження класичних підходів, використовуючи здатність глибоких нейронних мереж апроксимувати довільні функції взаємодії користувача з елементом [38, 50].

Постановка задачі НКФ схожа з задачею колаборативної фільтрації. Нехай задано матрицю взаємодій користувача з елементами $Y \in \{0,1\}^{M \times N}$, де M та N позначають кількість користувачів та елементів відповідно. Таким чином, з точки зору задачі КФ мета полягає в прогнозуванні ймовірності того, що користувач u взаємодіятиме з елементом i :

$$\hat{y}_{ui} = f(u, i | \Theta), \quad (2.6)$$

де f представляє параметризовану вектором Θ (лінійну) функцію, що апроксимує кількісну характеристику взаємодії.

НКФ узагальнює задачу (2.6), замінюючи матричний добуток, який лежить в основі лінійних методів апроксимації деякою нелінійною трансформацією:

$$\widehat{y}_{ui} = f(P^T v_u^U, Q^T v_i^I | P, Q, \Theta_f),$$

де v_u^U та v_i^I є векторами one-hot кодування ідентифікаторів користувачів та елементів відповідно, $P \in \mathbb{R}^{M \times K}$ та $Q \in \mathbb{R}^{N \times K}$ є матрицями вкладень, а Θ_f представляє параметри нейронної мережі [50].

Варто зазначити, що НКФ моделює в задачі матричної факторизації (2.1) як окремий випадок узагальненої факторизації:

$$y_{ui} = a_{out}(h^T(p_u \odot q_i)),$$

де $\odot$ позначає поелементний добуток, h є вектором ваг, а a_{out} є функцією активації виходу [49].

2.2.2 Найпопулярніші моделі глибинного навчання, що використовуються в задачах колаборативної фільтрації

Для прогнозування факту взаємодії користувача з елементом з логістичною регресією $a_{out} = \sigma(x) = \frac{1}{1+e^{-x}}$, функція втрат (2.1) набуває вигляду перехресної ентропії:

$$\mathcal{L} = -\sum_{y_{ui} \in Y} y_{ui} \log \widehat{y}_{ui} + (1 - y_{ui}) \log(1 - \widehat{y}_{ui}).$$

Основна архітектура НКФ об'єднує вкладення користувачів та елементів і обробляє їх через багат шаровий персептрон [49, 50]:

$$z_l = a_l(W_l^T z_{l-1} + b_l),$$

де W_l , b_l та a_l представляють матрицю ваг, вектор зміщення та функцію активації для шару l відповідно.

За достатньої ширини та глибини багат шарового персептрона НКФ може апроксимувати будь-яку неперервну функцію, що відображає пари користувач-елемент у ймовірності взаємодії з довільною точністю, що є прямим наслідком теореми про універсальну апроксимацію [49].

Окрім багат шарового персептрону популярним рішенням є використання моделей на основі механізму уваги, що здатні обробляти послідовності взаємодій з елементами, при цьому динамічно визначаючи вплив різних латентних факторів на результат прогнозування [51, 52]:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V.$$

Для колаборативної фільтрації матриці запитів, ключів та значень виводяться з векторів-вкладень користувачів та елементів:

$$\begin{aligned} Q &= p_u W^Q, \\ K &= q_i W^K, \\ V &= q_i W^V. \end{aligned}$$

Механізм уваги зазвичай застосовується по відношенню до вхідної послідовності кілька разів (Multi-head attention) [53, 54]:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h) W^O,$$

де $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$.

Серед популярних архітектур моделей глибинного навчання, альтернативних попереднім двом, варто зазначити модель варіаційних автокодувальників. На відміну від вищезгаданих механізмів уваги та «класичних» перцептронів, основною задачею варіаційних автокодувальників є не прогнозування, а «стиснення» інформації про набір ознак шляхом апроксимації ймовірнісного розподілу, з якого породжені вхідні дані [55, 56].

Роботу автокодувальників можна розділити на два етапи – кодування, тобто ін'єктивне відображення з кожного елементу з вхідної вибірки x_u до певної реалізації випадкової величини

$$z \sim q_\phi(z|x_u),$$

та декодування, тобто «відновлення» елементу вхідної вибірки відповідно до реалізації величини, отриманої на етапі кодування:

$$\hat{x}_u \sim p_\theta(x_u|z).$$

Таким чином, функціонал втрат варіаційних автокодувальників поєднує реконструкцію та регуляризацию:

$$\mathcal{L}_{\mathcal{VAE}} = E_{q_{\phi}(z|x_u)}[\log p_{\theta}(x_u|z)] - \beta \cdot \text{KL}[q_{\phi}(z|x_u)|p(z)].$$

У контексті колаборативної фільтрації для вираження факту взаємодії користувача з елементом (неявного зворотного зв'язку), широко використовується поліноміальний розподіл [55]:

$$p_{\theta}(x_u|z) = \text{Multinomial}\left(N_u, \text{softmax}(f_{\theta}(z))\right),$$

де $N_u = \sum_i x_{ui}$ є загальною кількістю взаємодій.

Гіперпараметр β , що регулює вагу дивергенції Кульбака-Лейблера, контролює «компроміс» між точністю реконструкції та варіативністю роботи автокодувальника, що корисно для різноманіття рекомендацій [55, 56].

Незважаючи на можливість моделювання складних перетворень, НКФ має суттєвий недолік в порівнянні зі «звичайними» моделями матричної факторизації – апроксимація скалярного (матричного) добутку потребує великої розмірності латентного простору, причому залежність між точністю апроксимації та розмірністю латентного простору є логарифмічною.

Вкупі з розрідженими даними, якими зазвичай є матриці взаємодій користувача з елементами, занадто велика розмірність латентного стану та глибина мережі породжує ризик пере-навчання (overfitting) [50].

Для подолання даних труднощів використовуються техніки дропауту (dropout), нормалізацію за вхідною порцією даних (batch normalization) та негативного семплінгу (negative sampling) – «зашумлення» навчальної вибірки даними взаємодій, які не відбулися.

Дропаут є однією з основних технік регуляризації, що запобігає перенавчанню нейронних мереж шляхом випадкової деактивації нейронів під час тренування [56].

Під час тренування дропаут застосовує бінарну маску до прихованих активацій:

$$h_{\text{dropout}} = h \odot m,$$

де $m \sim \text{Bernoulli}(p)$ – бінарна маска з ймовірністю «викидання» ваг $1 - p$.

Оптимальна ймовірність дропауту залежить від даних – вищі рівні шуму вимагають підвищення значення. Для задачі колаборативної фільтрації ймовірності дропауту зазвичай коливаються від 0.2 до 0.8, залежно від глибини моделі та розрідженості даних [57, 58, 59].

Нормалізація за навчальним набором даних усуває «внутрішнє зміщення ознак» (internal covariate shift) шляхом нормалізації вибірки, що в даний момент обробляється мережею. Окрім вищезгаданої особливості, даний алгоритм забезпечує прискорене тренування, оскільки за рахунок нормалізації можна використовувати більші значення кроків градієнтних методів при оптимізації функціоналу втрат.

Таким чином, застосовуючи дропаут та нормалізації до НКФ, рівняння перетворюється на:

$$z_{l+1} = \text{Dropout} \left(\text{ReLU}(\text{BatchNorm}(W_l z_l + b_l)) \right).$$

Проте, регуляризація моделей НКФ забезпечується не лише дропаутом та нормалізацією даних, а й негативним семплінгом даних, тобто їхньою аугментацією.

Стратегії негативного семплінгу є різноманітними, найпоширеніші серед них:

- рівномірне семплювання з навчальної вибірки товарів та користувачів;
- пошук негативних елементів (hard negative mining) – вибір прикладів на основі прогнозів моделі;

- змагальне семплювання (adversarial sampling) – використання генеративно-змагальних мереж для покращення влучності рекомендацій.

Рівномірне семплювання становить найпростіший підхід до негативного семплювання, випадково обираючи негативні елементи з однаковою ймовірністю:

$$P(j|u) = \frac{1}{|J \setminus J^+(u)|},$$

де $J^+(u)$ позначає елементи, з якими взаємодіяв користувач u [60, 61]. Як можна побачити, саме рівномірне семплювання є «стратегією за замовчуванням» в алгоритмі байєсівського персоналізованого ранжування.

Попри обчислювальну ефективність, рівномірне семплювання характеризується повільною збіжністю, оскільки більшість випадково вибраних негативів впродовж навчання дають моделі мінімальну інформацію, тобто градієнти похибок рівні нулю за рахунок того, що жоден користувач не взаємодіє з більшістю асортименту елементів [61].

Пошук негативних елементів обирає "складні" негативні зразки, які модель оцінила як найбільш ймовірні кандидати до рекомендацій. Найпопулярніша реалізація, динамічний негративний семплінг (DNS) вибирає негативи як:

$$j^* = \operatorname{argmax}_{j \in J^-(u)} \widehat{r}_{uj},$$

де $\widehat{r}_{uj}$ представляє прогнозований «рейтинг» [48].

Альтернативна реалізація даного алгоритму, що зветься адаптивним пошуком негативних елементів (AHNS) вирішує проблему false negatives шляхом варіювання рівнів впевненості вибору негативного елемента-кандидата:

$$P_{\text{AHNS}}(j|u) \propto \exp\left(\beta \cdot \operatorname{rank}(\widehat{r}_{uj})\right),$$

де β контролює «жорсткість» розподілу, тобто чутливість до прогнозів моделі [58, 59].

Змагальний семплінг використовує генеративно-змагальні мережі (GAN) для створення негативних зразків через $\min_{\max}$ оптимізацію:

$$\min_G \max_D E_{x \sim p_{data}} [\log D(x)] + E_{z \sim p_z} [\log (1 - D(G(z)))],$$

де генератор G продукує негативні зразки, а дискримінатор D оцінює їхню належність до негативних зразків [60].

Змагальне персоналізоване ранжування синтезує вектор збурень, використовуючи градієнт функціоналу втрат самої моделі БПР, відносно до змінної збурення, тим самим позбавляючи необхідності навчання додаткової моделі [61]:

$$\Delta_{adv} = \epsilon \cdot \frac{\nabla_{\Delta} \mathcal{L}_{BPR}(\Theta + \Delta)}{|\nabla_{\Delta} \mathcal{L}_{BPR}(\Theta + \Delta)|_2}.$$

Такий вектор збурення викликаний максимізувати $\mathcal{L}_{BPR}$, тому й даний підхід і називають змагальним персоналізованим ранжуванням.

2.3 Постановка задачі видачі послідовних рекомендацій

Послідовні рекомендаційні системи спрямовані на прогнозування наступних взаємодій користувачів на основі їхніх історій взаємодій, впорядкованих хронологічно, з метою вивчення часозалежних патернів та дослідження зміни уподобань користувачів.

На відміну від традиційних методів, що розглядають взаємодії користувач-елемент як незалежні події, послідовні рекомендаційні системи моделюють часові залежності між послідовними взаємодіями для забезпечення точних та своєчасних рекомендацій [62, 63].

Нехай $\mathcal{I} = \{i_1, i_2, \dots, i_{|\mathcal{I}|}\}$ – множина елементів, а $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ – множина користувачів. Для кожного користувача $u \in \mathcal{U}$ нехай $S^u = \{i_1^u, i_2^u, \dots, i_{|S^u|}^u\}$ – хронологічно впорядкована послідовність елементів, з якими взаємодіяв користувач u (сесія), де $i_t^u \in \mathcal{I}$ представляє елемент, з яким відбулась взаємодія на часовому кроці t .

Задача послідовної рекомендації полягає в навчанні функції $f: S^u \rightarrow \mathcal{I}$, що прогнозує наступний елемент $i_{|S^u|+1}^u$, з яким користувач u найімовірніше взаємодіятиме, враховуючи його історичну послідовність взаємодій S^u [29, 30, 62, 63].

Формально, прогноз задається як:

$$\hat{i}_{|S^u|+1}^u = \arg \max_{i \in \mathcal{I}} P(i|S^u), \quad (2.7)$$

де $P(i|S^u)$ представляє ймовірність того, що користувач u взаємодіятиме з елементом i , враховуючи його послідовність (сесію) S^u .

Ранні підходи до послідовної рекомендації спиралися на моделі ланцюгів Маркова для захоплення ймовірностей переходу між послідовними елементами [46, 64]. Хоча ці ймовірнісні методи забезпечували інтерпретовані фреймворки для моделювання послідовних залежностей, вони страждали від серйозних проблем розрідженості даних та обмеженої здатності моделювання складних поведінкових патернів. Розширення матричної факторизації, такі як FPMC [46], намагалися вирішити ці обмеження шляхом поєднання колаборативної фільтрації з ланцюгами Маркова, але залишалися обмеженими припущенням про залежності першого порядку.

2.4 Рекурентні нейронні мережі в контексті задачі видачі послідовних рекомендацій

Впровадження глибокого навчання означило парадигмальний зсув у задачі видачі послідовних рекомендацій. GRU4Rec [9, 11, 12, 65] започаткував застосування Gated Recurrent Units до послідовних рекомендацій, демонструючи, що RNN здатні ефективно моделювати складні послідовні патерни в поведінці користувачів. Ця робота заклала основу для нейронних підходів у послідовній рекомендації та показала значні покращення порівняно з традиційними методами колаборативної фільтрації.

Основа GRU4Rec базується на Gated Recurrent Unit, який використовує два механізми керування для контролю потоку інформації:

$$\begin{aligned} r_t &= \sigma(W_r x_t + U_r h_{t-1}), \\ z_t &= \sigma(W_z x_t + U_z h_{t-1}), \\ \tilde{h}_t &= \tanh(W_h x_t + U_h (r_t \odot h_{t-1})), \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t, \end{aligned}$$

де x_t – вхідний вектор на часовому кроці t , h_t – прихований стан на часовому кроці t , W_r, W_z, W_h – матриці ваг для вхідної послідовності, а U_r, U_z, U_h – рекурентні матриці ваг.

GRU4Rec використовує one-hot кодування для представлення елементів, де довжина вхідного вектора дорівнює кількості елементів, і лише координата, що відповідає активному елементу, встановлюється в одиницю [65]. Вхід на часовому кроці t :

$$x_t = \text{one-hot}(\text{item}_t) \in \{0,1\}^{|J|},$$

де $|J|$ – загальна кількість елементів.

Фінальний вихідний шар продукує score елементів через лінійну трансформацію:

$$y_t = W_y h_t + b_y,$$

де $y_t \in \mathbb{R}^{|I|}$ містить «рейтинг» всіх елементів.

Спираючись на GRU4Rec, з'явилися декілька модифікацій даної архітектури, серед яких можна виділити модель NARM [66]. NARM є encoder-decoder архітектурою, що вирішує завдання session-based рекомендацій шляхом включення механізму attention для моделювання як послідовної поведінки користувача, так і його основної мети в межах поточної сесії. Ключова інновація полягає у використанні гібридного encoder, який поєднує глобальну послідовну інформацію з локальним вибором елементів на основі attention.

Локальний encoder використовує механізм attention для ідентифікації основної мети користувача шляхом фокусування на релевантних елементах у сесії, в той час як глобальний – останній вихід прихованого шару GRU [9, 11, 12, 66]:

$$\begin{aligned} c_t^g &= h_t, \\ c_t^l &= \sum_{j=1}^t \alpha_{tj} h_j, \\ \alpha_{tj} &= q(h_t, h_j), \\ q(h_t, h_j) &= v^T \sigma(A_1 h_t + A_2 h_j), \end{aligned}$$

де σ – сигмоїдна функція активації, A_1 та A_2 – матриці трансформації, а v – навчуваний параметричний вектор.

NARM поєднує обидва encoder через конкатенацію:

$$c_t = [c_t^g; c_t^l] = [h_t; \sum_{j=1}^t \alpha_{tj} h_j].$$

У якості декодувальника NARM застосовує bi-linear функцію подібності:

$$S_i = \text{emb}_i^T B c_t,$$

де $B \in R^{|D| \times |H|}$ – матриця трансформації, $|D|$ – розмірність item embedding, а $|H|$ – розмірність представлення сесії.

Фінальні «рейтинги» рекомендацій отримуються через softmax нормалізацію:

$$P(i|c_t) = \frac{\exp(S_i)}{\sum_{j=1}^{|I|} \exp(S_j)}.$$

Функціоналом втрат NARM є функція перехресної ентропії.

Не дивлячись на ефективність застосування рекурентних нейронних мереж для рекомендаційних систем, самі RNN стикаються з фундаментальними обмеженнями:

- Зникаючі градієнти обмежують їхню здатність моделювати довгострокові залежності.
- Послідовна обробка градієнтів (алгоритми зворотного поширення похибок в часі) перешкоджає ефективній паралелізації.
- Фіксовані розмірності прихованих станів обмежують можливість системи вивчати довгострокові залежності.

2.5 Системи на основі механізму уваги

Архітектура мереж типу «трансформер» (transformer) також широко використовується для послідовних рекомендацій, наприклад в мережі SASRec [9-12, 63]. SASRec пропонує використовувати механізм само-уваги (self-attention), об'єднуючи кілька таких шарів в блоки. Дана модель «навчається» шляхом мінімізації перехресної ентропії з негативним семплюванням:

$$\mathcal{L} = -\sum_{u \in \mathcal{U}} \sum_{t \in \mathcal{T}_u} \log \sigma(r_{s_t^u, t}) + \sum_{j \in \mathcal{N}_t} \log(1 - \sigma(r_{j, t})),$$

де $\mathcal{T}_u$ позначає часові кроки для користувача u , $\mathcal{N}_t$ представляє негативні зразки, а σ – сигмоїдна функція.

Як альтернатива, BERT4Rec [9, 67] адаптує двосторонню архітектуру трансформер, що полягає у вивченні послідовності у обох напрямках. Для послідовності $v = [v_1, v_2, \dots, v_n]$ представлення входу поєднує вектори вкладень елементів та їхніх позицій у хронологічній послідовності:

$$h_i^{(0)} = v_i + p_i,$$

де $v_i \in \mathbb{R}^d$ – вектор вкладень елементів, а $p_i \in \mathbb{R}^d$ – вектор вкладень позицій для v_i в поточній сесії.

На відміну від «односторонньої» (каузальної) уваги у SASRec, BERT4Rec обробляє послідовність в обох напрямках:

$$\begin{aligned} a^{(l-1)} &= \text{BatchNorm} \left(h^{(l-1)} + \text{Dropout} \left(\text{MH}(h^{(l-1)}) \right) \right), \\ h^{(l)} &= \text{BatchNorm} \left(a^{(l-1)} + \text{Dropout} \left(\text{PFFN}(a^{(l-1)}) \right) \right), \end{aligned}$$

де $\text{PFFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2$, BatchNorm – функція пакетної нормалізації:

$$\begin{aligned} \hat{x} &= \frac{X - \mu}{\sqrt{(\sigma)^2 + \epsilon}}, \\ \text{BatchNorm}(\hat{x}) &= \gamma \hat{x} + \beta, \end{aligned}$$

де μ – вектор середніх значень кожної ознаки вхідної матриці X , σ – вектор стандартних відхилень, а ϵ – нескінченно мала константа, що забезпечує ненульові значення знаменника. Також γ та β є векторами масштабу та зміщення, що є навченими параметрами моделі. Суть функції пакетної нормалізації –

апроксимація середнього та стандартного відхилення генеральної сукупності даних, що продукують вибірки X [67].

BERT4Rec випадково маскує частку ρ вхідних токенів і навчає модель прогнозувати ці замасковані елементи, використовуючи наступну стратегію:

- 80% обраних позицій замінюються на «[MASK]» токен;
- 20% обраних позицій замінюються на випадковий елемент.

Модель максимізує функцію максимальної правдоподібності, що еквівалентно мінімізації від'ємного логарифму правдоподібності замаскованих елементів:

$$\mathcal{L} = - \sum_{v_m \in v_{\text{masked}}} \log P(v_m | \tilde{v}), \quad (2.8)$$

де v_{masked} позначає множину замаскованих позицій, а $\tilde{v}$ – «зашумлена» вхідна послідовність.

Проте систематичні дослідження моделі виявили, що покращення показників якості рекомендацій, продукованих BERT4Rec часто вимагають інтенсивного налаштування гіперпараметрів даної моделі і можуть не завжди домогтися кращих результатів, аніж більш прості алгоритми [68].

Загалом, незважаючи на свою ефективність, моделі-трансформери мають квадратичну обчислювальну складність $O(L^2)$ відносно довжини послідовності L , що суттєво обмежує їхню масштабованість для довгих історій користувачів та сценаріїв розгортання в реальному часі [9].

Висновки до розділу 2

У цьому розділі розглянуто основні алгоритми колаборативної фільтрації (КФ), що продовжують залишатися однією з найбільш широкоживаних та універсальних парадигм, що використовуються для побудови сучасних рекомендаційних систем. Колаборативна фільтрація базується на припущенні, що користувачі з подібними історіями взаємодій матимуть подібні уподобання

щодо подібних товарів, що надає змогу системам генерувати рекомендації без необхідності в детальних описах товарів або явних характеристиках користувачів. Цей принцип довів свою ефективність у практичних додатках, де накопичується достатня кількість історичних даних про взаємодії користувачів, забезпечуючи високу якість рекомендацій у сценаріях з великомасштабними каталогами товарів.

У дослідженні сформовано загальну математичну постановку задачі колаборативної фільтрації та пов'язана з нею задача матричної факторизації для одержання латентних ознак векторів користувачів та елементів.

Продемонстровано, що матрична факторизація являє собою потужний математичний апарат, що робить можливим представлення матриці взаємодій користувачів та товарів як добуток двох матриць меншої розмірності, кожна з яких кодує приховані характеристики користувачів та товарів відповідно. Цей підхід значно скорочує обчислювальні витрати порівняно з роботою з повною матрицею, особливо у випадках, коли вхідні дані є розрідженими, причому більшість елементів матриці залишаються такими, з якими користувач не взаємодівав.

Серед найпопулярніших алгоритмів матричної факторизації було детально розглянуто метод змінних найменших квадратів (ЗНК), який являє собою ітеративний підхід до навчання латентних факторів. Алгоритм ЗНК добив свою значущість завдяки простоті реалізації, обчислювальній ефективності та здатності масштабуватися на великі матриці.

У дослідженні наведено оптимальні рішення у закритій формі (аналітичному вигляді) для кожного кроку оптимізації, що дозволило продемонструвати теоретичні основи алгоритму та обґрунтувати його конвергенцію. Наявність аналітичних рішень для підзадач ЗНК робить алгоритм особливо привабливим для практичної імплементації, оскільки надає змогу уникнути числової нестабільності, притаманної деяким ітеративним методам оптимізації.

У контексті задачі матричної факторизації в рекомендаційних системах було досліджено байєсівське персоналізоване ранжування (BPR), що представляє собою альтернативний підхід до задачі матричної факторизації, і є спеціалізованим алгоритмом навчання рекомендаційних систем, що працює з неявними рейтингами – тобто відсутністю заміряної метрики уподобання елемента користувачем. На відміну від стандартних методів найменших квадратів, BPR формулює задачу як проблему апроксимації відношень порядку, що виражається у попарному ранжуванні товарів. Мета такого відношення полягає в тому, щоб упорядкувати розглянуті товари вище від невиконаних у рейтингах користувачів. Цей підхід більш інтуїтивно моделює реальну природу проблеми видачі релевантних, зміщуючи акцент з відновлення рейтингу товару до видачі ранжованого списку власне рекомендацій товарів чи послуг.

Альтернативні методи матричної факторизації на основі нейронних мереж (НКФ) були розглянуті як узагальнення класичних підходів, метою яких є моделювання нелінійних взаємозв'язків між користувачами та товарами.

Алгоритми глибокого навчання у контексті колаборативної фільтрації забезпечують більшу гнучкість у моделюванні складних патернів взаємодій користувача з платформою (бізнесом) та дозволяють системам автоматично більш точно виконувати моделювання функцій, що покликані відобразити уподобання користувачів. У даній роботі досліджено, що багат шарові перцептрони можуть бути використані для кодування взаємодій між векторами-вкладеннями користувачів та товарів через послідовність нелінійних перетворень, що надає змогу моделювати глибші семантичні зв'язки порівняно з лінійною матричною факторизацією.

Механізми уваги (attention mechanism) були досліджені в межах цього розділу як спосіб неявного задання профілю користувача, інтерпретуючи останнього як набір останніх елементів, з яким у нього була взаємодія. Механізми уваги дозволяють моделям динамічно зважувати різні компоненти вхідного представлення на основі їхньої релевантності в межах поточного контексту, що забезпечує більшу гнучкість порівняно з «фіксованими»

алгоритмами на кшталт МНК чи багат шарового персептрону. При застосуванні до колаборативної фільтрації, механізми уваги дозволяють системам визначити, які історичні взаємодії користувача є найбільш релевантними при апроксимації міри його зацікавлення в новому товарі.

Варіаційні автокодувальники (VAE) були розглянуті як альтернативний ймовірнісний підхід до вивчення латентних векторів-представлень користувачів та елементів. VAE перш за все надають спосіб моделювання атрибутів користувачів та елементів у латентному просторі як ймовірнісні розподіли, що додає різноманіття при етапі генерації рекомендацій. Це, на відміну від застосування «детерміністичних» методів матричної факторизації, надає змогу рекомендаційним системам на основі VAE досліджувати невизначеність та варіативність в уподобаннях користувачів, що особливо потрібно в контексті задачі вивчення зміни уподобань користувачів з плином часу.

У цьому дослідженні також показано, що нейромережеві підходи до матричної факторизації демонструють високу якість при видачі рекомендацій на гетерогенних наборах, особливо при наявності достатньої кількості обчислювальних ресурсів для навчання складних моделей.

Проте, вибір між класичними методами, такими як ЗНК та BPR, та нейромережевими підходами повинен ґрунтуватися на конкретних вимогах до системи, включаючи обмеження на обчислювальні ресурси, та обсяг доступних даних. У середовищах з обмеженими обчислювальними ресурсами або де інтерпретованість є ключовою вимогою, класичні методи матричної факторизації залишаються кращим вибором.

У межах цього дослідження також можна зробити висновок, що залежність колаборативної фільтрації від якості та обсягу історичних даних є одним з найбільших обмежень даного сімейства алгоритмів. Зокрема, задача холодного старту, коли система повинна генерувати рекомендації для нових користувачів або товарів без попередніх взаємодій, залишається актуальною та вимагає розробки та впровадження додаткових обчислювальних модулів, таких як системи на основі фільтрації контенту, або системи на основі знань. Саме тому,

на практиці часто використовують гібридні підходи, де основною моделлю є саме система на основі колаборативної фільтрації, що доповнюється окремими модулями, що імплементують інші підходами для охоплення повного спектру задач видачі персоналізованих рекомендацій.

Також у цьому розділі досліджено еволюцію послідовних РС від ранніх методів на основі ланцюгів Маркова до сучасних глибоких архітектур. Показано, що класичні підходи, такі як FPMC, хоча і забезпечували високий рівень інтерпретованості результатів у межах задачі моделювання послідовних залежностей, все ж вони страждали від серйозних проблем розрідженості даних та обмеженої здатності моделювання складних поведінкових патернів.

Детально проаналізовано GRU4Rec як один з перших підходів у застосуванні рекурентних нейронних мереж до рекомендацій на основі сесій. Розглянуто ключові компоненти архітектури, та показано, що GRU4Rec продемонстрував значні покращення порівняно з традиційними методами колаборативної фільтрації.

Вивчено розвиток гібридних підходів, зокрема NARM, що поєднує інформацію про глобальний контекст з локальним застосуванням механізму уваги. Проаналізовано, як такі архітектури дозволяють одночасно захоплювати як послідовні залежності, так і важливість окремих елементів, забезпечуючи більш комплексне розуміння сесійної поведінки користувачів.

У розділі також розглянуто вплив архітектур типу трансформер на розвиток послідовних рекомендаційних систем. Найпершим чином проаналізовано SASRec з його механізмами само-уваги та здатністю моделювати довільні залежності, долаючи обмеження RNN на довжину вхідних послідовностей.

У якості альтернативи до попередньої моделі було досліджено BERT4Rec як приклад застосування двостороннього механізму уваги з навчанням відновлення замаскованих елементів послідовності. Розглянуто переваги та обмеження такого підходу, включаючи потребу в інтенсивному налаштуванні

гіперпараметрів та неконсистентні покращення за умов справедливого порівняння даного підходу з традиційними.

Виявлено ключові обмеження існуючих підходів. Методи на основі рекурентних нейронних мереж страждають від затухаючих градієнтів, що обмежує їхню здатність моделювати довгострокові залежності, а послідовна обробка перешкоджає ефективній паралелізації. Методи на основі механізму уваги, хоча і подолали деякі з цих обмежень, мають квадратичну обчислювальну складність відносно довжини послідовності, що суттєво обмежує їхню масштабованість для дослідження дуже довгих послідовностей.

У розділі окреслено перспективи розвитку даних підходів, що полягають у розробці більш ефективних архітектур, здатних обробляти довгі послідовності з лінійною або сублінійною (логарифмічно-лінійною) складністю.

Особливу увагу варто приділити розробці методів, що балансують між модельною складністю та практичними вимогами до розгортання в реальному часі. Саме компромісне рішення забезпечить задовільну точність, достатнє охоплення елементів-кандидатів у рекомендації користувачу, при цьому не сповільнюючи процес надання таких рекомендацій.

Виконаний у межах цього розділу аналіз методів вирішення задач колаборативної фільтрації та видачі послідовних рекомендацій продемонстрував, що постає проблема у створенні гібридного методу, або низки гібридних методів, кожен з яких закликаний подолати недоліки існуючих рішень, забезпечити високий ступінь узагальнення вподобань користувачів, при цьому модифікуючи проаналізовані патерни шляхом надання більшої переваги саме останнім діям користувача.

РОЗДІЛ 3 МЕТОДОЛОГІЯ ПОБУДОВИ ГІБРИДНОЇ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ ВИДАЧІ ПЕРСОНАЛІЗОВАНИХ РЕКОМЕНДАЦІЙ

Попередній розділ було присвячено аналізу сучасних алгоритмів рекомендаційних систем, що вирішували проблему видачі персоналізованих рекомендацій (2.7). Такі системи ґрунтуються насамперед на архітектурах моделей глибинного навчання для обробки послідовностей, та на припущенні, що послідовність взаємодії користувачів та елементів є часовим рядом, що є природнім. Проте, вирішення задачі (2.7) не обмежується лише аналізом вхідної послідовності – історії взаємодій користувача, оскільки втілення даного сценарію прогнозу наступних елементів, що перегляне/придбає користувач є застосуванням методології РС на основі контенту, що, як відомо та описано у розділі 1, має суттєвий недолік – фокусуватися саме на елементах, з якими була взаємодія нещодавно, тим самим значно звужуючи перелік можливих елементів.

З іншої сторони, РС на основі методів колаборативної фільтрації здатні оперувати усім наявним асортиментом товарів та послуг, проте нездатні вирішити проблему «холодного старту», оскільки вимагає від кожного елемента наявність історії взаємодії з ним. Також, методи матричної факторизації не працюють з послідовностями, що унеможлиблює їхнє застосування для задачі видачі послідовних рекомендацій.

В даній роботі вперше представлено методологію, що полягає у поєднанні сильних сторін методів колаборативної фільтрації та методів побудови рекомендаційних систем на основі контенту, оскільки КФ враховує схожість користувачів між собою, тим самим розширюючи перелік можливих елементів, над якими проводиться оцінювання релевантності, а РС на основі контенту фокусується власне на вирішенні задачі (2.7), що є ключовою в межах цього дослідження.

Варто зазначити, що збільшення кількості товарів та послуг, над якими відбувається операція ранжування, в умовах аналізу виключно послідовностей

взаємодій користувачів полягає у збільшенні ємності моделі глибинного навчання, тобто у збільшенні кількості параметрів, що має за мету можливість обробки довших послідовностей. Такий підхід є неоптимальним, оскільки складніші моделі довше навчаються та повільніше видають результати.

Таким чином, замість збільшення складності моделі пропонується використовувати додаткові структури даних, що містили б необхідну інформацію про схожість елементів між собою навіть за умови відсутності нещодавньої взаємодії з ними. Отже, задачу (2.7) можна модифікувати і отримати наступне формулювання:

$$\hat{i}_{|S^u|+1}^u = \arg \max_{i \in \mathcal{I}_S \cup \mathcal{N}_i} \sum_{v \in \mathcal{N}_u} d(u, v) P(i|S^u), \quad (3.1)$$

де $\mathcal{N}_i$ – околиця елемента i – набір схожих з ним елементів, наявних у структурі даних, $\mathcal{N}_u$ – околиця користувача u включно з даним користувачем, а $d(u, v)$ – метрика схожості користувачів.

При вирішенні задачі (3.1) $\mathcal{N}_u$ виводиться методом КФ, а $P(i|S^u)$ обчислюється методами побудови РС на основі контенту. Саме з метою виведення множини $\mathcal{N}_i$ вводиться спеціальна структура даних, запропонована даною методологією.

Загалом, для реалізації методології необхідно виконати наступні етапи:

- ввести структуру даних $\mathcal{D}_J$, що містить закодовані взаємозв'язки між наявними елементами множини елементів $\mathcal{J}$;
- ввести метрику відстані між користувачами $d(u, v)$, що використовується для «зважування контекстних вподобань» користувачів в околиці користувача u ;
- ввести модель глибинного навчання, що моделює контекстні (умовні) ймовірності на основі сесій користувачів.

Окрім явно заданої задачі, модель має бути оптимізована відносно часу своєї роботи.

Ця умова є критично важливою для підприємств та бізнесів, що володіють великими асортиментами товарів та послуг, а також розраховані на масштабування клієнтської бази.

3.1 Метод побудови графу знань для представлення сесій користувачів

Головною об'єднуючою особливістю рекомендаційних систем, описаних вище є те, що вони працюють з табличними даними, причому різні види взаємодій описуються різними таблицями. Наступні РС, описані в цьому розділі, пропонують інтерпретувати взаємодії користувачів з елементами за допомогою інших структур даних – графів. Концептуальною основою даної пропозиції є те, що рекомендаційні дані мають природну графову структуру, де вузли відповідають сутностям (користувачі, елементи, категорії, атрибути), а ребра кодують відношення (взаємодії, подібності, таксономічні ієрархії). Така організація даних надає змогу РС навчатися представленням, що інкорпорує як локальну топологічну інформацію (тобто короткі причинно-наслідкові зв'язки), так і глобальні структурні характеристики графа (моделювання продуктових кошиків) [3-8, 69, 70].

Основною графовою структурою в рекомендаційних системах є дводольний граф користувач-елемент $\mathcal{G}_{\mathcal{U}\mathcal{I}} = (\mathcal{U} \cup \mathcal{I}, \mathcal{E}_{\mathcal{U}\mathcal{I}})$, що формально визначається як:

- $\mathcal{U} = \{u_1, u_2, \dots, u_m\}$ – множина користувачів;
- $\mathcal{I} = \{i_1, i_2, \dots, i_n\}$ – множина елементів;
- $\mathcal{E}_{\mathcal{U}\mathcal{I}} \subseteq \mathcal{U} \times \mathcal{I}$ – множина взаємодій користувач-елемент.

Матриця інцидентності $A_{\mathcal{U}\mathcal{I}} = [a_{ui}] \in \{0,1\}^{m \times n}$ кодує взаємодії згідно з індикаторною функцією $a_{ui} = 1[\text{користувач } u \text{ взаємодіяв з елементом } i]$.

Графи знань (knowledge graphs, KG), як реалізація даної структури даних, використовуються РС для представлення сутностей як вузлів та їхніх відношень як орієнтованих ребер, забезпечуючи семантично збагачену основу для

рекомендаційних систем. У контексті послідовних рекомендацій графи знань моделюють складні взаємозалежності між користувачами, елементами, сесіями та контекстуальною інформацією [3-5].

Графи знань формально задаються як трійка $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, де $\mathcal{E}$ є множиною сутностей (користувачі, елементи, категорії), $\mathcal{R}$ – множиною типів відношень (придбав, переглянув, належить-до), а $\mathcal{T} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ містить множину трійок (h, r, t) , що представляють певні твердження (факти) [3, 4, 69].

Розширюючи базові користувач-елемент граfi інкорпоруванням допоміжної інформації, формуються гетерогенні граfi знань $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, де множина сутностей декомпозується наступним чином [3, 5]:

$$\begin{aligned}\mathcal{E} &= \mathcal{U} \cup \mathcal{I} \cup \mathcal{C} \cup \mathcal{A}, \\ \mathcal{C} &= \{c_1, c_2, \dots, c_k\} \quad (\text{категоріальні сутності}), \\ \mathcal{A} &= \{a_1, a_2, \dots, a_l\} \quad (\text{атрибутивні сутності}).\end{aligned}$$

У свою чергу множина відношень задається зліченою множиною якісних атрибутів:

$$\mathcal{R} = \{\text{взаємодіє, належить_до, темпорально_слідує}\}.$$

При роботі з графовими РС необхідно перш за все отримати граф знань з наявних даних та сесій. Нехай дано колекцію користувацьких сесій $\mathcal{S} = \{s_1, s_2, \dots, s_N\}$, де кожна сесія $s_i = \{i_1^{(s)}, i_2^{(s)}, \dots, i_{|s|}^{(s)}\}$ представляє послідовність взаємодій з елементами. Тоді метою даної задачі є конструкція графа знань $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$, де:

- 1) $\mathcal{V}$ представляє множину елементів (вузлів);
- 2) $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ представляє відношення (ребра);
- 3) $\mathcal{W}: \mathcal{E} \rightarrow \mathbb{R}^+$ призначає ваги ребрам на основі статистичних мір.

Для вирішення даної задачі запропоновано використовувати статистичні підходи.

Один з запропонованих автором дисертації підходів ґрунтується на метриці попарної загальної інформації (pointwise mutual information, PMI). PMI – це статистична метрика, що адресує проблему того, наскільки більш імовірно для двох елементів з'явитися разом, в порівнянні до того, якби вони були статистично незалежними. Для двох елементів $i, j \in \mathcal{I}$ PMI визначається як:

$$\text{PMI}(i, j) = \log \frac{P(i, j)}{P(i)P(j)} = \log \frac{N \cdot c_{i, j}}{c_i \cdot c_j}.$$

У формулі визначення $\text{PMI}(i, j)$ вище N – загальна кількість сесій, $c_{i, j}$ – кількість сесій, що містять обидва елементи i та j , а c_i та c_j – індивідуальні частоти елементів по всіх сесіях

PMI базується на основних принципах теорії інформації та теорії ймовірностей. На відміну від простих показників спільної появи, які враховують лише абсолютну частоту одночасного виникнення двох подій, PMI враховує очікувану частоту таких подій при припущенні їхньої статистичної незалежності [3-5, 69].

Таким чином, принциповою різницею PMI від частотних показників є можливість розрізнення справжніх асоціацій від випадкових збігів. Якщо дві події є статистично незалежними, їх PMI дорівнюватиме нулю. Якщо вони позитивно корельовані (з'являються разом частіше, ніж очікувалося), PMI буде позитивним. Якщо вони негативно корельовані, PMI буде від'ємним.

Отримані значення PMI використовуються для конструкції матриці суміжності графа знань. Вага ребра між елементами i та j в графі знань тоді становить:

$$w_{ij} = \frac{\text{PMI}(i, j)}{\max_{k, l} \text{PMI}(k, l)}.$$

Ця нормалізація забезпечує, що ваги ребер належать відрізку $[0,1]$.

Окрім PMI для виділення релевантних пар елементів поміж сесій користувачів пропонується використовувати статистику TF-IDF, що порівнює частоту кожного елемента в загальній історії покупок до частоти сесій, що налічують даний елемент. Мотивація використання цієї метрики приходить з галузі інформаційного пошуку, де дана статистика допомагала знаходити ключові терми для представлення документів.

Основна ідея TF-IDF полягає у вирішенні двох конфліктуючих аспектів текстового аналізу. По-перше, «локальна важливість» передбачає, що якщо терм часто зустрічається в документі, він ймовірно важливе для розуміння контексту цього документа. По-друге, глобальність (всюдисущість) терму полягає в тому, що якщо слово зустрічається майже у всіх документах вхідного корпусу, воно не несе специфічної інформації та робить неможливим відрізнити один документ від іншого.

Для елемента i в сесії s частота термів (елементів) визначається як:

$$\text{TF}(i, s) = \frac{f_{i,s}}{\sum_{j \in s} f_{j,s}},$$

де $f_{i,s}$ представляє частоту елемента i в сесії s . Це вимірює відносну важливість елемента i в сесійному контексті.

Обернена частота документів вимірює, наскільки рідкісним або поширеним є елемент по всьому корпусу сесій:

$$\text{IDF}(i) = \log \frac{|S|}{|\{s \in S : i \in s\}|},$$

де $|S|$ – загальна кількість сесій, а знаменник підраховує сесії, що містять елемент i .

Метрика TF-IDF для елемента i в сесії s є простим добутком попередніх двох метрик:

$$\text{TF-IDF}(i, s) = \text{TF}(i, s) \times \text{IDF}(i).$$

TF-IDF визначається простотою та ефективністю реалізації та є швидким в обчисленні навіть для великих корпусів [3-5]. На відміну від Word2Vec/Node2Vec чи BERT, метод не потребує тривалого тренування нейронної мережі, оскільки являє собою суто статистичний підхід. Результати методу легко інтерпретувати, оскільки вага документа оцінюється на основі конкретних слів.

Проте, TF-IDF не розуміє семантичного значення слів та ігнорує порядок термів у документі. До того ж, при великій кількості термів у словнику вектори стають дуже розрідженими, що ускладнює обчислення і вимагає значних обсягів оперативної пам'яті.

У роботі автор дисертації пропонує гібридний метод розрахунку ваг графу взаємодій [5]:

$$w_{ij}^{\text{TF-IDF}} = \left(\frac{\overline{\text{TF-IDF}}(i) + \overline{\text{TF-IDF}}(j)}{2} \right),$$

де $\overline{\text{TF-IDF}}(i)$ представляє середній TF-IDF елемента i по всіх сесіях, що його містять.

Також в роботах автора дисертації було запропоновано використання правил асоціацій для зв'язування елементів в графові знань.

Зокрема, враховуючи, що, правило асоціації може бути задано імплікацією форми $X \Rightarrow Y$, де $X, Y \subseteq \mathcal{I}$ – множини елементів, в контексті даного дослідження, набуває подальшого розвитку застосування добування асоціативних правил для рекомендаційних систем, шляхом трактування даних імплікації як причинно наслідкових зв'язків так: «якщо елементи з множини X з'являються в сесії користувача, то, враховуючи схожі транзакції, користувач доповнить кошик

елементами з множини Y ». Для побудови правил використовується метрика підтримки (support) [3-5, 10-12]:

$$\text{support}(X) = \frac{|\{s \in \mathcal{S} : X \subseteq s\}|}{|\mathcal{S}|},$$

що вимірює частоту елемента (елементів) серед усіх сесій; а також метрику «підйому» (lift) [3-5, 10-12]:

$$\text{lift}(X \Rightarrow Y) = \frac{\text{support}(X \cap Y)}{\text{support}(X)\text{support}(Y)},$$

що вимірює, наскільки більш імовірно набір елементів Y з'явиться при наявності X , порівняно з незалежною появою обидвох наборів, що є схожим з метрикою PMI.

Таким чином, автором дисертації у межах методу побудови графу причинно-наслідкових зв'язків для елементів i та j запропоновано визначати вагу спрямованого ребра від i до j як функцію, що залежить від метрики lift та support [3, 4]:

$$w_{i \rightarrow j} = \text{support}(i \cup j) \cdot \mathbb{1}[\text{lift}(i \Rightarrow j) > \varepsilon],$$

де ε – гіперпараметр порогового відсіювання значень метрики.

Ці статистичні підходи мають низку переваг:

- PMI та TF-IDF обчислення масштабуються лінійно з кількістю сесій, роблячи підхід придатним для великих наборів даних;
- PMI, TF-IDF та дерева FP-росту можуть обчислюватися на кількох машинах одночасно, дозволяючи значно прискорити швидкість побудови графів причинно-наслідкових зв'язків;

- всі вищенаведені алгоритми підтримують інкрементальні оновлення, тобто при надходженні нових сесійних даних нема потреби в перезапуску процедури для всього набору даних.

Отже, узагальнюючи, даний метод можна представити у вигляді блок-схеми, що зображена на рисунку 3.1.

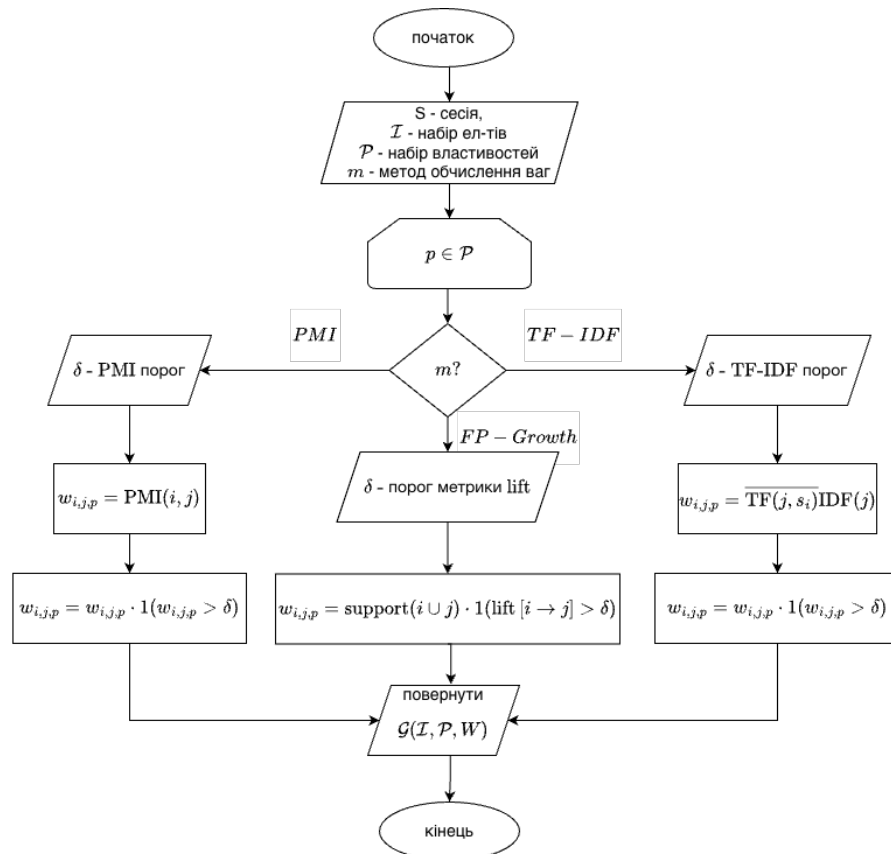


Рисунок 3.1 – Алгоритм методу побудови графу знань

3.2 Алгоритми навчання представлень вершин графів

Після отримання графу постає необхідність чисельного представлення вершин одержаної структури для можливості обробки даної інформації архітектурами нейронних мереж. Серед найпоширеніших алгоритмів отримання вкладень вершин можна навести алгоритм DeepWalk [71]. Даний алгоритм генерує стохастичні блукання на графі користувач-елемент та застосовує підхід

пропуску елементів (skip-gram) для навчання стиснених представлень. Нехай $f: V \rightarrow \mathbb{R}^2$ – функція генерації вкладень вершин графу. Тоді для вершини $v \in V$ оптимізаційна цільова функція формулюється як задача відновлення з контексту скіп-грам, тобто максимізації правдоподібності, подібно до (2.6):

$$\max_f \sum_{c \in \mathcal{N}_{wl}(v)} \log P(c|f(v)),$$

де $\mathcal{N}_{wl}(v)$ представляє контекстні вузли, отримані зі стохастичних блукань, що ініціюються в v . DeepWalk використовує рівномірний розподіл при визначенні переходу від вершини $v \in V$ до вершини $v \in V$.

У свою чергу, алгоритм Node2Vec [72], що пропонується до використання автором дисертації, вводить наступну функцію «ненормалізованих ймовірностей» переходу, що залежить від ваги ребра w_{vx} та контролюється гіперпараметрами p (параметр повернення) та q (параметр дослідження):

$$\pi_{vx} = \alpha_{pq}(t, x) \cdot w_{vx},$$

де функція $\alpha_{pq}(t, x)$ визначається як:

$$\alpha_{pq}(t, x) = \begin{cases} \frac{1}{p}, & \text{якщо } d_{tx} = 0, \\ 1, & \text{якщо } d_{tx} = 1, \\ \frac{1}{q}, & \text{якщо } d_{tx} = 2, \end{cases}$$

де d_{tx} – функція відстані між вершинами графу t та x . Параметр повернення контролює, наскільки часто в процесі стохастичного блукання буде обрано початкову вершину, а параметр дослідження – наскільки далеко в глибину графу відносно початкової вершини може зайти стохастичне блукання. Таким чином може бути обрана будь-яка вершина з околиці сусідів 2-го порядку вершини v [73].

Узагальнюючи, алгоритм отримання векторів-вкладень вершин графа знань зображений на рисунку 3.2 у вигляді блок-схеми.

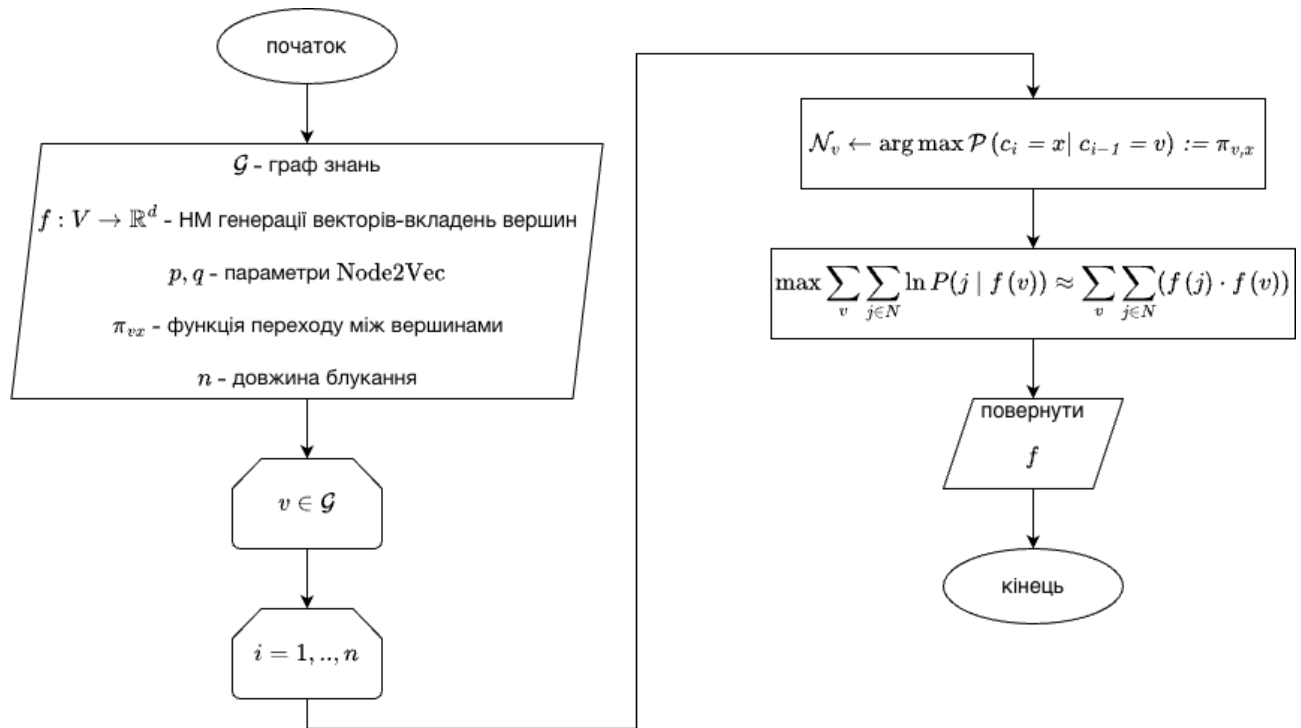


Рисунок 3.2 – Алгоритм Node2Vec чисельного представлення вершин графу

У контексті РС, $p > 1$ буде «стимулювати» систему досліджувати околиці більшої кількості елементів, гарантуючи унікальність згенерованої послідовності, тоді як $q > 1$ змушуватиме алгоритм охоплювати якомога більше елементів, з якими найчастіше взаємодіяли при взаємодії з поточним елементом, гарантуючи «повноту контексту» послідовності [5, 72, 73].

3.3 Методи побудови нейронних мереж для обробки графових даних

3.3.1 Метод на основі модифікації моделей згорткових мереж

Однією з найперших архітектур нейронних мереж, що були пристосовані до роботи з графовими даними є графові згорткові мережі (graph convolutional networks, GCN) [74].

Основне призначення GCN у цьому контексті – виконання рекурсивної агрегації інформації для пошуку найвідповідніших заданому контексту елементів. У дводольному графі $\mathcal{G}$, де вершини-вузли представляють користувачів та елементи, а ребра позначають спостережувані взаємодії (наприклад, кліки, покупки або оцінки) GCN працює шляхом ітеративної агрегації інформації про ознаки з локального оточення вузла для оновлення його векторного представлення (ембедингу). Через кілька шарів графової згортки вузол може накопичувати інформацію від сусідів на все більшій відстані від цільової точки (кількості переходів). Цей механізм, який часто називають «поширенням вкладень» [72, 73], надає змогу вузлу користувача включати латентні ознаки не лише елементів, з якими він безпосередньо взаємодіє, але й користувачів зі схожими уподобаннями (сусіди другого порядку) та елементів, які ці користувачі обрали (сусіди третього порядку). У розріджених наборах даних, де прямі взаємодії користувач-елемент обмежені, це працює особливо ефективно, надаючи перевагу перед традиційними методами, оскільки змішуючи представлення розріджених вузлів з ознаками їхніх сусідів, простір ембедингів будується відповідно до структури графа.

Шар графової згортки, що стоїть в основі даної архітектури, агрегує інформацію від суміжних вузлів згідно з правилом:

$$h^{(l+1)} = \sigma \left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} h^{(l)} W^{(l)} \right),$$

де $\tilde{A} = A + I$ є матрицею суміжності з «петлями», $\tilde{D}$ – матрицею степенів вершин, а $W^{(l)}$ – навчуваною матрицею лінійної трансформації для шару l [5, 74].

GCN ґрунтується на спектральній теорії графів, оскільки згортка графу з певним оператором під впливом перетворення Лапласа перетворюється на добуток лапласіана графу з перетворенням цього оператора, причому:

$$\mathcal{L}\{\mathcal{G}\} = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}.$$

LightGCN спрощує GCN архітектуру для використання в РС шляхом використання лінійних функцій активацій проміжних шарів та ознакових трансформацій [5]:

$$e_u^{(k+1)} = \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u| |\mathcal{N}_i|}} e_i^{(k)},$$

$$e_i^{(k+1)} = \sum_{u \in \mathcal{N}_i} \frac{1}{\sqrt{|\mathcal{N}_u| |\mathcal{N}_i|}} e_u^{(k)}.$$

Для задач колаборативної фільтрації лінійний підхід, як правило, є достатнім, оскільки задача видачі рекомендацій – тобто пошуку відповідності типу «користувач-елемент» – по суті зводиться до обчислення подібності у багатовимірному просторі.

Фінальне представлення конструюється як лінійна комбінація всіх шарів:

$$e_u = \sum_{k=0}^K \alpha_k e_u^{(k)}.$$

3.3.2 Метод на основі модифікацій моделей механізму уваги

Окрім згорткових нейронних мереж, механізми уваги також широко використовуються для роботи з графовими даними. Серед найперших таких застосувань можна привести мережі уваги над графами (graph attention networks, GAT) [5, 10-12]. GAT вирішують фундаментальне обмеження звичайних

графових згорткових мереж – нездатність моделі адаптивно зважувати внесок сусідніх вузлів. У традиційних GCN всі сусіди поточного елемента розглядаються як рівнозначні з точки зору агрегації, хоча у реальних мережах значимість кожного сусіда часто суттєво варіює.

GAT використовують маскований механізм само-уваги, який дає змогу навчити індивідуальні ваги для кожного сусіда кожної заданої вершини:

$$h_i^{(l+1)} = \sigma \left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l)} W^{(l)} h_j^{(l)} \right).$$

Для вузла i коефіцієнт уваги до вузла j обчислюється як відносна релевантність останнього для формування нової латентної репрезентації [73]:

$$\alpha_{ij}^{(l)} = \frac{\exp(\text{LeakyReLU}(a^T [Wh_i || Wh_j]))}{\sum_{k \in \mathcal{N}(i)} \exp(\text{LeakyReLU}(a^T [Wh_i || Wh_k]))}.$$

Проте, як видно з опису мереж GAT, вони пристосовані лише до роботи з графовими даними, що не задовольняє ключовій вимозі представленої методології враховувати контекст поточної сесії, тобто унеможлиблює використання одночасно з графами сесійних даних у межах одного нейромережевого модулю на основі механізму уваги.

Таким чином, автором дисертації було вперше модифіковано механізм само-уваги, що уможлиблює врахування наявних взаємозв'язків між елементами, що представлені графом $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, де $\mathcal{V}$ – множина елементів, а $\mathcal{E}$ – множина ребер, причому задана $\text{Adj}(\mathcal{G})$ – матриця суміжності графу [3]. Нехай додатково до графу та набору даних сесій користувачів $S = [v_{i_1}^{(0)}, v_{i_2}^{(1)}, \dots, v_{i_k}^{(t-1)}]$, $v \in \mathcal{V}, i_1, i_2, \dots, i_k \in |\mathcal{V}|$ задано матрицю вкладень вершин $E \subset \mathbb{R}^{|\mathcal{V}| \times d}$. Врахувавши вищезазначене, автором запропонована модель глибинного навчання, що обчислює наступну найкращу пропозицію за заданим алгоритмом:

- 1) обчислити для кожного $v_{i_k}^{(t-1)}$ вектори вкладення E_{i_k} ;
- 2) обчислити для кожного $v_{i_k}^{(t-1)}$ суму векторів вкладень сусідів першого порядку $E_{N_{i_k}}$;
- 3) обчислити зважену суму $E_\Sigma = \alpha E_{N_{i_k}} + (1 - \alpha) E_{i_k}$, де $\alpha \in [0,1]$;
- 4) обчислити матриці Q, K, V через одношарову згорткову нейронну мережу з розміром фільтру, рівним одиниці;
- 5) обчислити коефіцієнти само-уваги $A_\Sigma = \text{SelfAttention}(Q, K, V)$;
- 6) обчислити вектор-кандидат найкращого наступного елементу $\hat{e}_{pred} = \text{ANN}(A_\Sigma)$.

Далі, маючи вектор-кандидат $\hat{e}_{pred}$ та вектор наступної пропозиції, виділений з навчальної (тестової) вибірки e_{true} представлена РС навчається шляхом мінімізації перехресної ентропії.

Незважаючи на суттєві покращення значень метрик ранжування, що демонструються застосуванням даного методу в порівнянні з моделями на основі мереж довгої короткострокової пам'яті [4], даний метод потребує суттєвого покращення, оскільки застосування суми векторів вершин-сусідів робить «вплив» кожного з елементів, що представлені цими векторами, на поточний елемент рівномірним, не зважаючи на особисті потреби користувача та порядок перегляду елементів в сесії, а також постає необхідність регулювати гіперпараметр α , що робить мережу, що спродукована даним методом занадто негнучкою.

Саме тому в межах цього дисертаційного дослідження автором дисертації вперше представлена модифікація попереднього методу, що, застосовуючи лінійну комбінацію традиційних архітектур механізмів уваги, вводить нову архітектуру моделей глибинного навчання, що називається змішаною увагою (mixed attention) [5].

Нехай $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ позначає граф знань, де $\mathcal{V}$ – множина вузлів (наприклад, елементів у каталозі), а $\mathcal{E}$ – множина ребер. Кожне ребро $e \in \mathcal{E}$ характеризується кортежем (u, v, w, r) , що з'єднує вузли $u, v \in \mathcal{V}$ з вагою $w \in \mathbb{R}$ та типом

семантичного відношення (або властивістю) $r \in \mathcal{R}$. Для виокремлення специфічних реляційних структур у графі вводиться оператор семантичної проєкції P :

$$P: (\mathcal{G}, r) \mapsto \mathcal{G}_r,$$

де $\mathcal{G}_r = (\mathcal{V}, \mathcal{E}_r)$ – індукований підграф, що містить лише ребра типу відношення r . Цей оператор надає змогу моделі вибірково звертати увагу на різні семантичні контексти (наприклад, «куплено», «переглянуто»).

Таким чином, поведінка користувача визначається як часова послідовність взаємодій. Нехай сесія користувача визначається як $S = [(i_1, r_1), (i_2, r_2), \dots, (i_{t-1}, r_{t-1})]$, де кожен кортеж представляє взаємодію з елементом $i_k \in \mathcal{V}$ за певної властивості дії r_k . Тоді, з метою охоплення латентної семантики цих взаємодій визначаються три дійснозначані матриці вкладень: $E_I \in \mathbb{R}^{|\mathcal{V}| \times d}$ для представлення елементів, $E_R \in \mathbb{R}^{|\mathcal{E}| \times d}$ для представлення властивостей (відношень) та $E_T \in \mathbb{R}^{|\mathcal{T}| \times d}$ для представлення часових міток, де d – розмірність вкладення.

Метод «змішаної уваги» запропонований автором дослідження з метою інтеграції гетерогенних джерел інформації шляхом обчислення оцінок уваги за трьома різними модальностями: «чистими» залежностями послідовності на рівні елементів, шаблонами переходів на рівні властивостей та локальною топологією графа. Формально, задаючи перетворення запиту (query), ключа (key) та значення (value), вихід змішаної уваги формується через конкатенацію незалежних блоків уваги (МНА):

$$\text{MixedAttention}(Q, K, V, \mathcal{G}) = \left[\text{MNA}_S(Q(E_S), K(E_S), V(E_S)), \parallel_r \text{MNA}_\mathcal{G}(Q(\mathcal{G}_r), K(\mathcal{G}_r), V(\mathcal{G}_r)) \right],$$

де $\parallel$ позначає операцію конкатенації вздовж виміру ознак.

Перший компонент, MHA_S , фіксує послідовні залежності в просторах елементів в контексті сесії S аналогічно стандартній самоувазі (self-attention) у трансформерах. Другий компонент, MHA_G , відповідає за пошук контексту у графові знань. Зокрема, для цільового елемента i_t на кроці t частина компоненту MHA_G обчислюється шляхом агрегації представлень його сусідів першого порядку $\mathcal{N}(i_t)$ у спроектованому підграфі $\mathcal{G}_r$. Це надає змогу моделі динамічно зважувати важливість безпосередніх сусідів на основі поточного контексту сесії, поєднуючи «глобальну» інформацію з графа знань із «локальною» інформацією щодо поточної сесії користувача.

Підводячи підсумок, процедура отримання значень уваги (attention scores) запропонованим методом змішаної уваги, представлена у вигляді блок-схеми на рисунку 3.3.

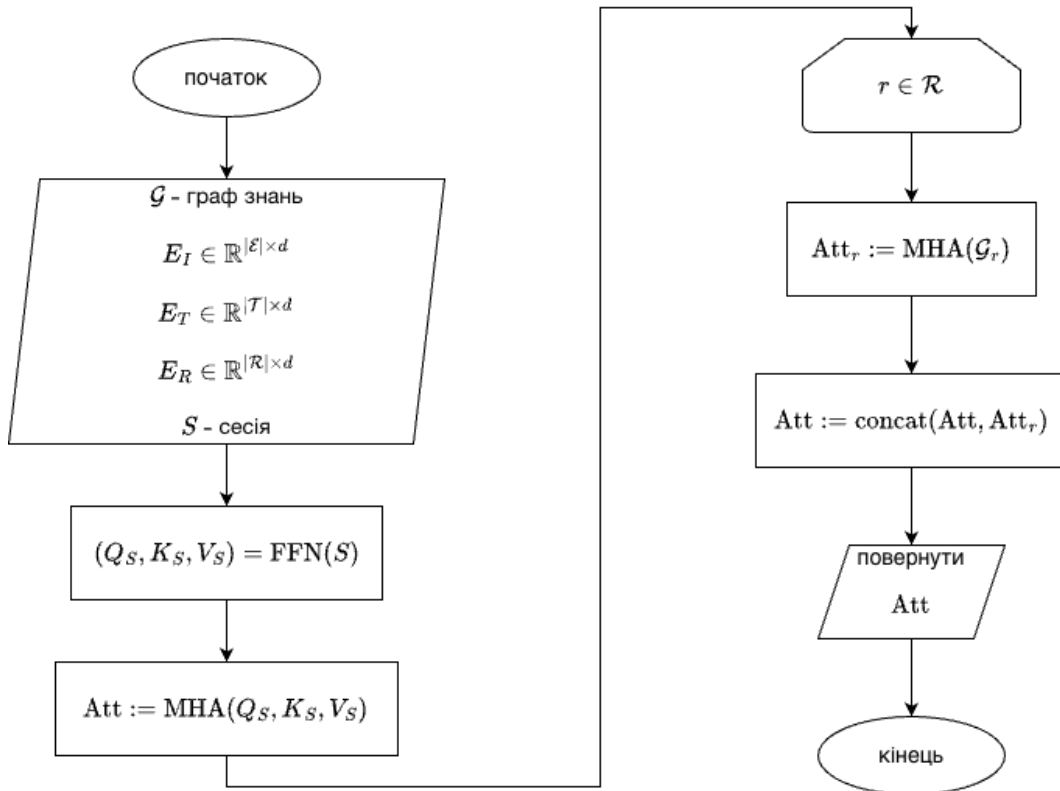


Рисунок 3.3 – Алгоритм отримання attention scores запропонованим методом змішаної уваги

Як видно з опису методу, він є напрочуд простим, проте ефективним рішенням. За рахунок конкатенації виходів шарів багатоголової уваги, застосованих до кожної проекції графу знань та послідовності (сесії) користувача, отримано спосіб динамічного зважування внесків кожного елемента на вибір користувачем наступної найкращої пропозиції, а також модифікації ваг векторів-вкладень елементів відповідно до типу взаємодії, що робить метод ефективним для застосування на гетерогенних графах знань [5].

3.4 Подальший розвиток методу дистиляції знань для рекомендаційних систем

Оскільки всі архітектури, наведені вище, потребують для своєї експлуатації або задані графи знань, або використання одночасно сесійних даних разом з графами знань, необхідним є постійне підтримання актуальності даного графу, а також використання великої кількості параметрів, що здатні обробляти матриці інцидентності великих розмірів. Дані умови експлуатації графових нейронних мереж є суттєвими обмеженням для їхнього використання на мобільних пристроях а також ставлять під сумнів можливість РС, створених на їх основі, надавати відповіді за короткий проміжок часу. Методологія дистиляції знань (knowledge distillation, KD) [5, 74] адресує дану проблему, пропонуючи певний компроміс між простотою мережі та її точністю через «передачу знань» від складних «моделей-вчителів» до компактних «моделей-студентів».

Основна ідея методу полягає в тому, що знання нейронної мережі зосереджені не лише в навчених параметрах, а в відображенні від входів до виходів. Зокрема, коли добре натренована модель-вчитель виконує обробку вхідного набору даних, вона призначає значення ймовірності належності кожному класу для кожного кортежу. Ці ймовірності репрезентують те, що в літературі називають «прихованим знанням» (dark knowledge) [74] – це відображення впевненості моделі у своїх «судженнях».

Основний механізм дистиляції знань полягає у масштабованому softmax-відображенні. Для моделі з логіт-вектором $z = [z_1, z_2, \dots, z_K]$ для K класів стандартний softmax обчислює:

$$p_i = \frac{\exp(z_i)}{\sum_{j=1}^K \exp(z_j)}.$$

В якості масштабування використовується параметр температури τ , що контролює «м'якість» (рівномірність) розподілу ймовірностей:

$$p_i(\tau) = \frac{\exp(z_i/\tau)}{\sum_{j=1}^K \exp(z_j/\tau)}.$$

Низька температура ($\tau \rightarrow 1$) забезпечує «впевненість» моделі в тому, що модель-вчитель коректно передбачає кожен клас в кожному випадку. При цьому висока температура ($\tau \gg 1$) перетворює розподіл відповідей моделі-вчителя на більш рівномірний, виконуючи роль регуляризації при навчанні моделі-студента.

Модель-студент в свою чергу навчається шляхом мінімізації композитної функції втрат відносно відповідей моделі-вчителя та відносно заданих міток класів:

$$\begin{aligned}\mathcal{L}_K &= \tau^2 \cdot \text{KL}(p^T(\tau) \| p^S(\tau)), \\ \mathcal{L}_C &= -\sum_{i=1}^K y_i \log p_i^S, \\ \mathcal{L} &= (1 - \alpha)\mathcal{L}_C + \alpha\mathcal{L}_K,\end{aligned}$$

де $\alpha \in [0,1]$ є гіперпараметром «пріоритету задачі».

В $\mathcal{L}_K$ фактор τ^2 забезпечує, що градієнти залишаються подібною величини при виборі значення температури [74].

Дистиляція є агностичною (незалежною) відносно архітектури моделі. Модель-вчитель і модель-студент можуть мати повністю різні архітектури, що і

є мотивацію використання даного методу для задачі видачі персоналізованих рекомендацій. До того ж, дистиляція знань має низку інших переваг над «звичайним процесом» навчання моделі:

- моделі-студенти часто збігаються швидше, ніж моделі, навчені з нуля, оскільки пряма оптимізація шарів, що відповідають за «приховані знання» полягає у обчисленні більшої кількості значень градієнту при процесі оптимізації [74];
- дистиляція може працювати з синтетичними даними у випадку відсутності «міток класів», оскільки виходи «моделі-вчителя» надають інформацію про квазі-класи [75].

Наприклад, дистиляція моделі BERT, що має близько 340 млн. параметрів в DistilBERT, що є моделлю з 66 млн. параметрів досягає 60% зменшення розміру та двократного прискорення інференсу з лише 3% втратою точності [76]. Причому, DistilBERT використовується на мобільних пристроях для вирішення задачі розуміння команд користувача додатками Siri та Google Assistant.

До того ж, дистиляція може виконуватися не лише над вихідними шарами, знання можуть передаватися з проміжних [77]:

$$\mathcal{L}_{feature} = \|h^S - W_{proj}h^T\|_2^2,$$

де h^S та h^T є проміжними представленнями від моделей студента та вчителя відповідно, а W_{proj} є матрицею, що «вирівнює» їхні розмірності.

У межах методу онлайн-дистиляції обидві моделі навчаються спільно [78]:

$$\mathcal{L}_{online} = \mathcal{L}_C^T + \mathcal{L}_C^S + \alpha \mathcal{L}_K.$$

У деяких випадках належно дистильовані моделі досягають вищої точності, ніж їхні «вчителі», завдяки ефекту регуляризації [79]. У межах дослідження було вирішено використовувати власне онлайн-дистиляцію.

Емпірично встановлено, що оптимальні значення температури в задачах дистиляції знань зазвичай лежать у діапазоні $\tau \in [3, 20]$, який забезпечує достатнє згладжування вихідного розподілу вчителя та водночас зберігає його інформативність для студента. У численних дослідженнях було показано, що підвищення температури збільшує ентропію softmax-розподілу, роблячи видимими так звані «приховані знання» – відносні подібності між усіма класами, а не лише ймовірність правильного класу, що може суттєво покращити навчання меншої моделі [74].

Разом з тим занадто високі температури (наприклад $\tau \geq 100$) призводять до майже рівномірного розподілу ймовірностей по класах, різко зменшуючи контраст між правильним і хибними класами та погіршуючи прогнозну здатність моделі-студента, оскільки сигнал від учителя стає слабко структурованим. Додаткові дослідження даного феномену показують, що в цьому режимі корисна інформація про відносні переваги класів втрачається, градієнти стають малими й мало відрізняються між прикладами, що ускладнює оптимізацію й може заводити навчання в плато [5, 74].

Навпаки, надто низькі значення температури ($\tau \rightarrow 1$) фактично зводять процедуру дистиляції до звичайного навчання з вчителем з «жорсткими» мітками, коли розподіл моделі-учителя стає майже one-hot-закодованим і при цьому модель-студент майже не отримує додаткової «прихованої» інформації про структуру простору класів. У такій ситуації переваги дистиляції над класичним навчанням неочевидні, оскільки процес оптимізації дедалі більше визначається лише справжніми мітками, як у стандартній крос-ентропії [74].

Проте, як було отримано у дослідженні автора, саме така конфігурація значення температури гарантувала найкращу точність моделі-учня [3-5, 74]. Незважаючи на заявлені недоліки, саме «режим» навчання з вчителем виявився найкращим застосуванням методу дистиляції знань при задачі спрощення архітектури моделі-учня шляхом використання лише даних поточної сесії користувача без урахування графів знань [4, 5].

У межах цього дослідження набув подальшого розвитку метод дистиляції знань в контексті застосування до послідовних рекомендаційних систем.

Узагальнюючи процедуру отримання дистильованої моделі пошуку взаємозв'язків між елементами поточної сесії користувача та його довгостроковими вподобаннями, алгоритм зображено на блок-схемі на рисунку 3.4.

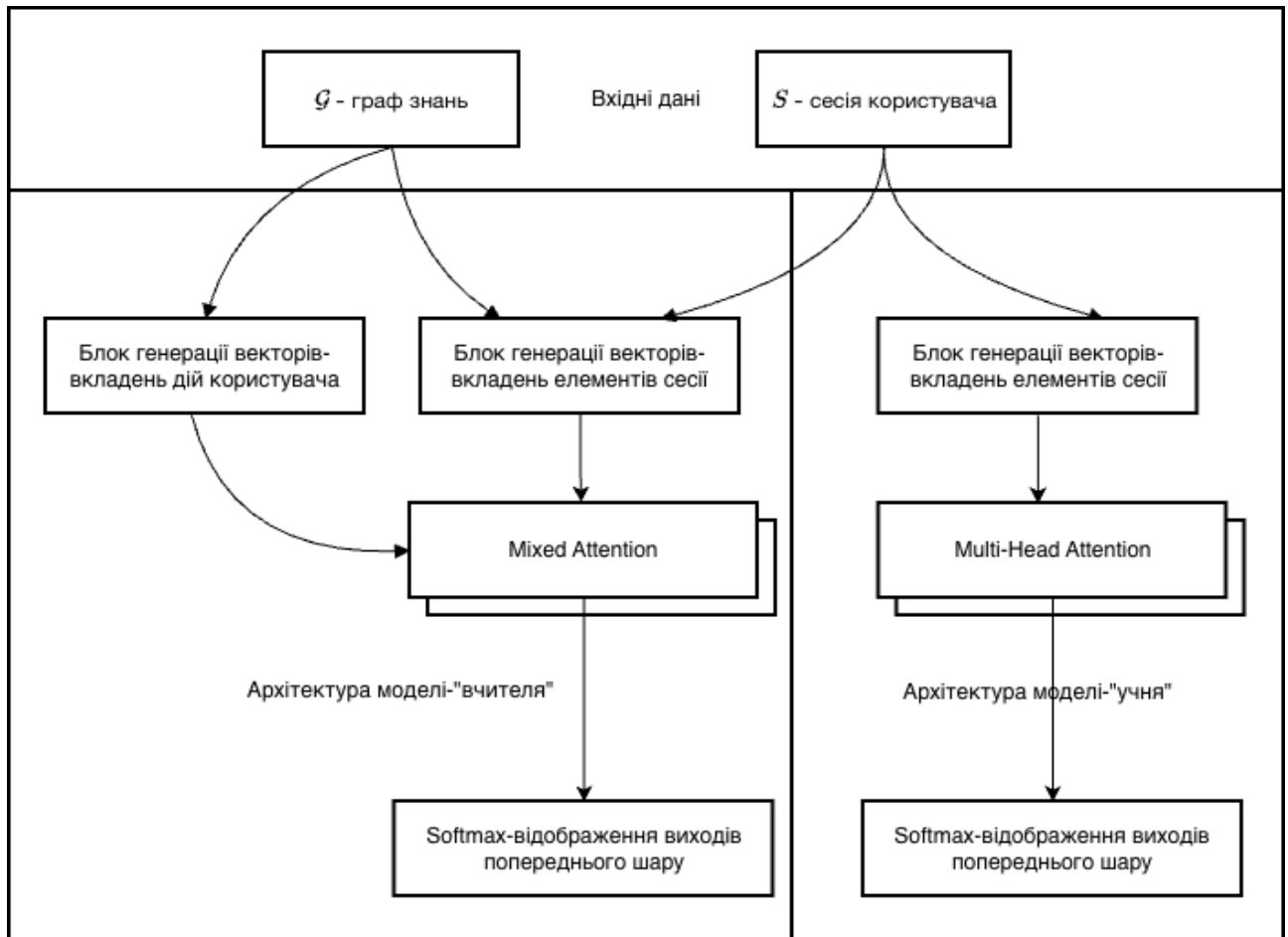


Рисунок 3.4 – Застосування методу дистиляції знань для отримання модулю співставлення елементів в сесії з вподобаннями користувача з метою зважування рейтингів елементів

Метою застосування даного методу, як видно з схематичного опису на рисунку вище, є інкапсуляція знань моделі вчителя щодо структури графу знань, тобто отримання набору векторів-вкладень, що відображають

взаємозв'язки елементів між собою без прямого обходу графу знань в ширину на етапі інференсу, що додатково економить час обробки мережею вхідних кортежів даних.

3.5 Аналіз методів на основі моделей нейронних диференціальних рівнянь

Оскільки профілі користувачів, що відображають їхні вподобання, змінюються з часом, а відповідно й зв'язки елементів між собою, то досить природнім є припущення, що еволюцію «рейтингу» елемента можна описати через диференціальне рівняння:

$$\frac{dr_{ui}}{dt} = \phi(\theta_u, \theta_i, t),$$

де θ_u – параметри профілю користувача, θ_i – характеристики елемента, а $t \in \mathbb{R}^+$ – (неперервний) час. При заданій початковій умові $r_{ui}(0)$ дане рівняння можна вирішити аналітично або чисельно [2, 3, 80].

Раніше в контексті даного дослідження були розглянуті архітектури рекурентних нейронних мереж, саме що здатні обчислювати майбутні значення подібних часових рядів, визначених на домені дискретного часу. Проте саме в цьому й основне обмеження подібних архітектур – вони припускають рівномірний розподіл часових міток [80].

З метою подолання нечутливості таких моделей були введені архітектури нейронних звичайних диференціальних рівнянь (neural ODE), що власне моделюють функцію ϕ як нейронну мережу [1, 2, 80-82].

За заданого початкового стану $r(t_0)$, стан у будь-який майбутній момент часу t_i є розв'язком задачі Коші:

$$r(t_1) = r(t_0) + \int_{t_0}^{t_1} \phi(r(t), t, \theta) dt, \quad (3.2)$$

Інтеграл у рівнянні (3.2) обчислюється за допомогою чисельних методів розв'язків ЗДР, наприклад, Рунге-Кутта 4-го порядку.

Автором дисертаційного дослідження набули подальшого розвитку методи побудови нейронних мереж на основі архітектур кодувальник-декодувальник (автокодувальник) та нейронних ЗДР з метою покращення точності апроксимації деякого часового ряду з нерівномірно розподіленими часовими мітками [1, 2]. Шляхом задання кодувальника рекурентною нейронною мережею GRU, а оновлення динаміки прихованого стану нейронною ЗДР була отримана модель LODE-GRU-VAE, що продемонструвала хороші результати в задачі апроксимації динаміки заданих часових рядів, досягнувши значення метрики R^2 рівному 0.8, що надало змогу реконструювати сам часовий ряд шляхом екстраполяції [1].

Загалом, шари мережі LODE-GRU-VAE задаються низкою рівнянь:

$$\begin{aligned} h''_{\tau} &= \text{Sol}(f_{\theta}, h_{\tau-1}, t_{\tau-1}, t_{\tau}), \\ u_{\tau} &= \sigma(W_{xu}x_{\tau} + W_{hu}h''_{\tau} + b_u), \\ r_{\tau} &= \sigma(W_{xr}x_{\tau} + W_{hr}h''_{\tau} + b_r), \\ h'_{\tau} &= \tanh(W_{xh'}x_{\tau} + W_{hh'}(r_{\tau} \odot h''_{\tau}) + b_{h'}), \\ h_{\tau} &= u_{\tau} \odot h''_{\tau} + (1 - u_{\tau}) \odot h'_{\tau}, \end{aligned}$$

де $\text{Sol}(f_{\theta}, h_{\tau-1}, t_{\tau-1}, t_{\tau})$ позначає чисельне рішення рівняння (3.2), де інтегрування функції $\phi(r(t), t, \theta) := h_{\tau-1}, r(t_0) := f_{\theta}, t_0 := t_{\tau-1}$, а $t_1 := t_{\tau}$.

Таким чином, модель LODE-GRU-VAE описується формулами і є варіаційним кодувальником. Кодування вхідних даних x у даній моделі характеризується умовним ймовірнісним розподілом $q(z/x)$, де z – випадковий вектор у латентному просторі. Функцією втрат у даному випадку є математичне сподівання втрат відносно z :

$$\mathbb{E}_{z \sim q(z/x)} L(z, q) = \mathbb{E}_{z \sim q(z/x)} \ln p(x, z) - D_{KL}(q(z/x) \parallel p(z)),$$

де $p(x, z)$ – сумісний розподіл x та z , $D_{KL}(A \parallel B)$ – розходження Кульбака-Лейблера, що визначає ступінь «несхожості» розподілів A і B .

Обґрунтування варіаційного кодувальника базується на методі Expectation Maximization (EM). Для автокодувальника вхід та ціль – один і той самий вектор x . Тому декодувальник при подачі на вхід коду z повертає умовний розподіл $p(x/z)$, а функція похибок визначає правдоподібність «прив’язки» функції похибок до виходу останнього шару мережі [1].

Подальше удосконалення LODE-GRU-VAE шляхом апроксимації розподілу самих часових міток за допомогою випадкових процесів – нормальних (гаусівських) та пуасонівських – надало змогу точніше виконувати екстраполяцію часового ряду та досягти R^2 рівному 0.9146 [2].

Застосування нейронних ЗДР в контексті рекомендаційних систем є перспективним та вкрай малодослідженим напрямком, проте перші кроки були зроблені та вилились в модель ReCODE [9, 81].

Оцінка рейтингу елементів $\hat{r}_{ui}$ для елемента i в певний час t розкладається в межах ReCODE на два адитивні компоненти: оцінку «безумовного» рейтингу $\hat{r}_{ui}^{\text{sta}}$ та оцінку коливання рейтингу відносно часу $\hat{r}_{ui}^{\text{dyn}}$:

$$\hat{r}_{ui} = \hat{r}_{ui}^{\text{sta}} + \hat{r}_{ui}^{\text{dyn}}.$$

«Безумовний» рейтинг користувача відносно елемента, моделюється мережею, що генерує вектори вкладень користувача e_u та елемента e_i . Для цих цілей авторами ReCODE використовуються загальноприйняті методи побудови рекомендаційних системи, наприклад, матрична факторизація або штучна нейронна мережа, здатна обробляти послідовності, як SASRec.

Основна ж увага дослідження приділяється моделюванню саме динамічної компоненти рейтингу. Процедура моделювання використовує нейронні ЗДР для

знаходження чисельного розв'язку задачі Коші, причому рівняння (3.2) набуває вигляду:

$$\begin{aligned}\hat{r}(t_{i+1}) &= \hat{r}(t_i) + \int_{t_i}^{t_{i+1}} \phi(r(t), t, \theta) dt, t_i = \overline{0, T}, \\ \hat{r}(t_0) &:= \hat{r}_{\text{sta}}\end{aligned}\quad (3.3)$$

Навчання параметрів (3.3) проводиться з використанням попарної функції втрат ранжування. Ця функція побудована таким чином, щоб прогнозована оцінка рейтингу для елемента, з яким була взаємодія i^+ була вищою, ніж для такого, з яким користувач не стикався i^- :

$$\mathcal{L} = -\sum_{u \in \mathcal{U}} \sum_{t \in \mathcal{S}_u} \log \sigma(\hat{r}_{u,t^+} - \hat{r}_{u,t^-}),$$

де σ – сигмоїдна функція. Градієнти для компонента ОДР обчислюються за допомогою методу спряженої чутливості (adjoint sensitivity), що забезпечує ефективність використання пам'яті під час навчання [80, 81].

Проте, навіть з урахуванням застосування методу спряженої чутливості, дослідження на тему порівняння нейронних ЗДР з «класичними» в контексті роботи з часовими рядами рекурентними нейронними мережами, такими як LSTM, та авторегресивними статистичними моделями, критикують велику обчислювальну складність нейронних ЗДР [2, 82].

Зокрема, автори демонструють, що для досягнення порівнянної точності на наборах даних, що були досліджені авторами нейронних ЗДР потребують значно довшого часу навчання – збіжність досягається на порядок повільніше, ніж у LSTM або простих авторегресійних моделей.

3.6 Метод на основі моделей простору станів

Проаналізувавши сильні та слабкі сторони методів на основі моделей нейронних диференціальних рівнянь, можна дійти висновку, що для багатьох

практичних задач прогнозування часових рядів такі моделі є надмірно параметризованими та складними для аналізу. Цей висновок підкріплюється неможливістю автора дисертації досягти завершення навчання моделей LODE-VAE-P, LODE-GRU-VAE, LODE-VAE-N на наборах даних, що використовувалися для оцінки точності прогнозування наступної найкращої дії користувача методу на основі модифікацій моделей механізму уваги. Отже, постає необхідність у синтезі методу, що використовує простіші моделі, які здатні забезпечити аналогічну точність прогнозування при значно меншому часі навчання.

До таких методів відносяться моделі простору станів (state-space models, SSM), що постали як альтернативи архітектурам на основі механізмів уваги для моделювання довгострокових поведінкових патернів, причому маючи лінійну обчислювальну складність [83, 84].

Розвиток глибоких SSM являє собою перспективний напрямок розвитку моделей глибинного навчання для вивчення і прогнозування послідовностей, що знаходить своє застосування в РС в тому числі [4, 5, 9].

В основі сучасних SSM стоїть фреймворк операторів поліноміальних проекцій високого порядку (high-order polynomial projection operator, HiPPO) [83]. HiPPO розглядає сімейство часово-змінних мір μ_t , що задані на інтервалі на $(-\infty, t]$, та шукає оптимальні поліноміальні апроксимації функцій над цими мірами. Нехай задано неперервну функцію $f: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$, що представляє інтенсивність взаємодій користувача, HiPPO визначає коефіцієнти $c(t) \in \mathbb{R}^N$ як оптимальну проекцію:

$$c(t) = \arg \min_{g \in G} \|f(x)\mathbb{1}_{x \leq t} - g\|_{L^2(\mu_t)}, \quad (3.4)$$

де $G = \text{span}\{g_n\}_{n=0}^{N-1}$ є поліноміальним базисом, ортогональним відносно міри μ_t .

Дана оптимізаційна задача (3.4) допускає розв'язання через систему лінійних звичайних диференціальних рівнянь:

$$\frac{dc(t)}{dt} = A(t)c(t) + B(t)f(t), \quad (3.5)$$

де матриці $A(t)$ та $B(t)$ залежать від обраної міри та базисних функцій, забезпечуючи оптимальну компресію історії взаємодій [83-85].

Автори теорії HiPPO-операторів пропонують використовувати два базиси, відносно яких відбувається декомпозиція вхідного часового ряду. Кожен з базисів є видозміною ортогонального базису з поліномів Лежандра. Поліноми Лежандра $P_n(x), n = \overline{1, N}$ визначені на відрізку $[-1, 1]$ це многочлени, що формують ортогональний базис, що визначається низкою умов:

$$\int_{-1}^1 P_n(x)P_m(x)dx = 0, n \neq m,$$

$$P_n(1) = 1.$$

HiPPO-LegS (Scaled Legendre) використовує міру $d\mu_t(x) = \frac{1}{t} \mathbb{1}_{[0,t]}(x)dx$, що зважає всі історичні взаємодії однаково, підходить для захоплення довгострокових користувацьких уподобань, виражаючи матриці керування та переходів станів (3.5) як:

$$A_{nk}^{\text{LegS}} = \begin{cases} (2n+1)^{\frac{1}{2}}(2k+1)^{\frac{1}{2}}, & \text{якщо } n > k, \\ n+1, & \text{якщо } n = k, \\ 0, & \text{якщо } n < k, \end{cases}$$

$$B_n^{\text{LegS}} = (2n+1)^{\frac{1}{2}}.$$

З іншого боку, HiPPO-LegT (Translated Legendre) застосовує ковзне вікно міри $d\mu_t(x) = \frac{1}{\theta} \mathbb{1}_{[t-\theta, t]}(x)dx$ з параметром довжини часового вікна θ , продукуючи часово-інваріантні матриці (3.5), що мають більшу вагу для нещодавніх подій:

$$A_{nk}^{\text{LegT}} = \frac{1}{\theta} \begin{cases} (-1)^{n-k}(2n+1), & \text{якщо } n \geq k, \\ 2n+1, & \text{якщо } n \leq k, \end{cases}$$

$$B_n^{\text{LegT}} = \frac{(-1)^n}{\theta} (2n+1).$$

Модель структуровних простору станів (S4) [85] пропонує оптимізувати матрицю зміни стану A через наступну декомпозицію:

$$A = \Lambda - PQ^*,$$

де $\Lambda \in \mathbb{C}^{N \times N}$ є діагональною, а $P \in \mathbb{C}^{N \times R}$ є матрицею з низьким рангом. Ця структура називається DPLR декомпозицією та надає змогу зволяє ефективно обчислювати параметри c_t через дискретизацію системи (3.5) та подальшої операції швидкого перетворення Фур'є, зменшуючи складність з $O(N^2L)$ до $O(NL \log L)$ для довжини послідовності L [85].

Задля переходу від неперервної системи у просторі станів (3.5) до дискретної системи, автори фреймворку S4 використовують білінійне перетворення, також відоме як перетворення Тастіна (Tustin's transformation) [84, 85].

Дане перетворення послуговується кроком дискретизації Δ :

$$x_k = \left(I - \frac{\Delta}{2} \cdot A\right)^{-1} \left(I + \frac{\Delta}{2} \cdot A\right) x_{k-1} + \left(I - \frac{\Delta}{2} \cdot A\right)^{-1} \Delta \cdot B u_k. \quad (3.6)$$

Нехай $\bar{A} = \left(I - \frac{\Delta}{2} \cdot A\right)^{-1} \left(I + \frac{\Delta}{2} \cdot A\right)$, $\bar{B} = \left(I - \frac{\Delta}{2} \cdot A\right)^{-1} \Delta \cdot B$. Нехай задана система у просторі станів:

$$\begin{aligned} x_k &= \bar{A}x_{k-1} + \bar{B}u_k, \\ y_k &= \bar{C}x_k \end{aligned} \quad (3.7)$$

де $\bar{C}$ – певна матриця дійсних чисел. Тоді, підставляючи рівняння для x_k до y_k та «розгортаючи» рекурентне відношення до послідовності довжини L :

$$y_k = \bar{C}\bar{A}^{L-1}\bar{B}u_0 + \bar{C}\bar{A}^{L-2}\bar{B}u_1 + \dots + \bar{C}\bar{B}u_{L-1}. \quad (3.8)$$

Очевидно, що права частина рівняння (4.8) є згортою деякого ядра $\bar{K} := (\bar{C}\bar{A}^i\bar{B})_{i=0,L}$ з вхідною послідовністю u , а тому дану згортку можна обчислити за допомогою швидкого перетворення Фур'є [85].

Таким чином, рівняння (3.5) дискретизовано методом Тастіна (3.6), де отримано рекурентне відношення, що надає змогу «вбудовувати» такий функціонал у обчислювальні комірки звичайних рекурентних нейронних мереж і до прикладу вирішувати задачу sequence-to-sequence (3.7), або ж моделювати дану систему за допомогою згорткових мереж з метою навчання ядра згортки $\bar{K}$. На рисунку 3.5 зображені всі етапи переходу від вирішення ЗДР (3.4) до системи (3.8):

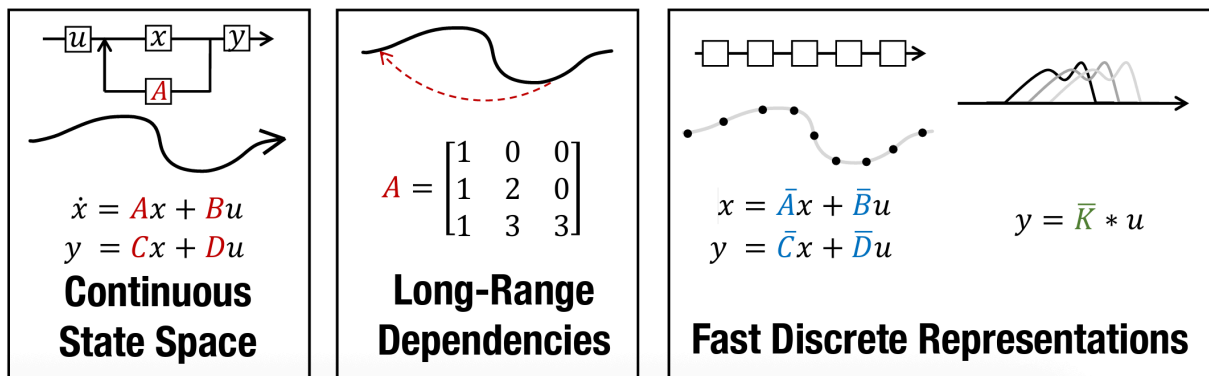


Рисунок 3.5 – Перехід від неперервної системи у просторі станів до дискретної системи, параметризованої HiPPO-матрицями [85]

Застосування SSM до вирішення задач рекомендаційних систем залишається перспективним та мало дослідженим напрямком, але перші кроки в цьому напрямку вже зроблено. Наприклад, модель SS4Rec [86] використовує S4

для обробки даних, що містять нерегулярні інтервали між взаємодіями користувачів.

В свою чергу робота автора дисертації вперше надає підхід до моделювання інтенсивності покупок товарів, модельованих як латентні неоднорідні Пуасонівські випадкові процеси з метою прогнозування матриць суміжності графу знань та ваг ребер, за допомогою SSM моделей та HiPRO-операторів [4, 5, 9].

Нехай існує точковий кількісний процес N заданий на борелівській множині S . Даний процес називається Пуасонівським з середнім (математичним очікуванням) μ , якщо задовольняються наступні вимоги:

- 1) для підмножин $B_1, \dots, B_k \subset S$, $\forall i, j B_i \cap B_j = \emptyset$ випадкові величини $N(B_i)$ є (попарно) незалежними;
- 2) для кожної підмножини B кількість подій $N(B) \sim \text{Poisson}(\mu(B))$;
- 3) $N(t) - N(s) \sim \text{Poisson}\left(\int_s^t \lambda(u) du\right)$, де $\lambda(t)$ є функцією інтенсивності.

Дані $\lambda(t)$ інтенсивності пропонується моделювати через лінійну модель у просторі станів, «керовану» матрицею суміжності певного графу знань (причинно-наслідкових зв'язків між елементами) у поточний момент часу [4, 5, 9]:

$$\begin{aligned} \frac{d\lambda_i(t)}{dt} &= A\lambda_i(t) + B \cdot \text{Adj}(\mathcal{G}_t)_i, \\ y_i(t) &= I\lambda_i(t) \end{aligned} \quad (3.9)$$

де A, B є матрицями HiPRO-LegS або HiPRO-LegT, а I – тотожне вихідне відображення.

Нехай, причинно-наслідковий граф елементів $\mathcal{G}_t$ є невідомим (окрім множин елементів), відповідно матриця суміжності графу є невідомою. Таким чином вводиться припущення, що матриця суміжності графу є певною

(нелінійною) залежністю від кількості елементів, над якими відбувалася взаємодія:

$$\widehat{\text{Adj}(\mathcal{G}_t)} := \phi(W \odot N_P)$$

Тут $\phi: \mathbb{R} \times \mathbb{R} \rightarrow [0,1] \times [0,1]$ є певною нелінійною функцією, N_P – матриця попарних кількостей взаємодій з елементами, а W – матриця параметрів «зважування» рядків матриці N_P .

Параметри моделі оптимізуються через метод максимальної правдоподібності. Таким чином, для подій, що сталися впродовж спостережуваних часових міток взаємодій $\mathcal{T} = \{t_1, t_2, \dots, t_n\}$, логарифмічна правдоподібність відносно системи (4.6) обчислюється:

$$\mathcal{L} = \sum_{t \in \mathcal{T}} \log \lambda(t) - \int_{\mathcal{T}} \lambda(t) dt = \sum_{t \in \mathcal{T}} \log \lambda(t) - N_i$$

Загалом, введений в межах дослідження метод надається у вигляді блок-схеми на рисунку 3.6.

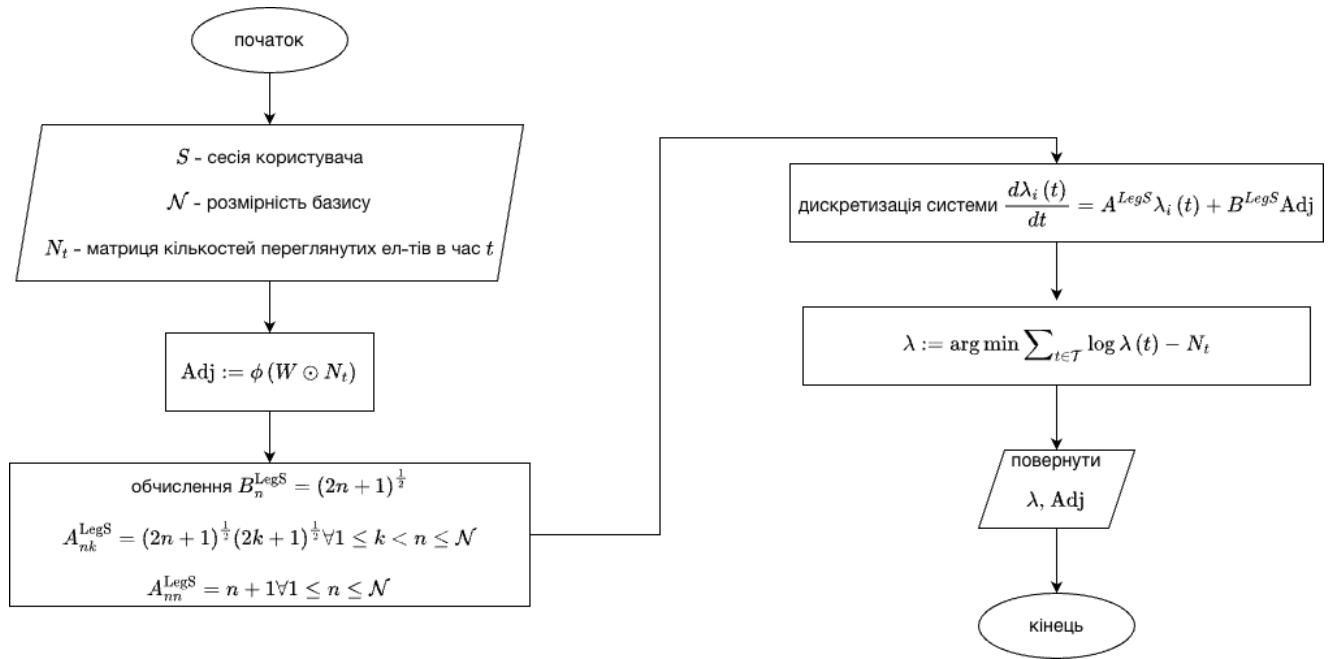


Рисунок 3.6 – Метод моделей у просторі станів для вирішення задачі видачі наступної найкращої пропозиції

Загалом, запропонований метод має суттєву перевагу в тому, що граф знань (причинно-наслідкових переходів) не задається як додаткова структура даних, що значно спрощує архітектуру моделі. Більше того, сама природа пуасонівських процесів робить їхнє застосування до задач видачі послідовних рекомендацій, підвидом якої є ключова розглянута задача дисертаційного дослідження – задача видачі наступної найкращої пропозиції – очевидним рішенням, що й надає змогу моделювати довгострокові залежності між елементами як матрицю суміжності невідомого графу знань, що залежить від кількості каузальних взаємодій над парами елементів, а короткострокові – як інтенсивність даного процесу.

Таким чином, у межах цього дисертаційного дослідження автором вперше представлена методологія побудови гібридних рекомендаційних системи на основі архітектур глибинного навчання та динамічних вкладень графів, що охоплює метод генерації графу знань на основі наявних історичних даних та статистичних метрик, метод, що полягає у модифікації моделі видачі

послідовних рекомендацій на основі механізму уваги для одночасного аналізу послідовностей та отриманого на попередньому кроці графу знань, що надає змогу враховувати поточний контекст та довгострокові залежності, збільшуючи точність результатів у межах вирішення задачі послідовних рекомендацій, застосування методу дистиляції знань з метою спрощення архітектури отриманої моделі на основі методу змішаної уваги, а також, як альтернативу попереднім ланкам, метод побудови гібридної мультимодальної рекомендаційної системи за допомогою спеціалізованих архітектур моделей простору станів, що ґрунтуються на операторах поліноміальної проекції високого порядку та моделюванні неоднорідних пуассонівських процесів з метою неявного моделювання графу знань у режимі реального часу. Схематично етапи методології зображені на рисунку 3.7.

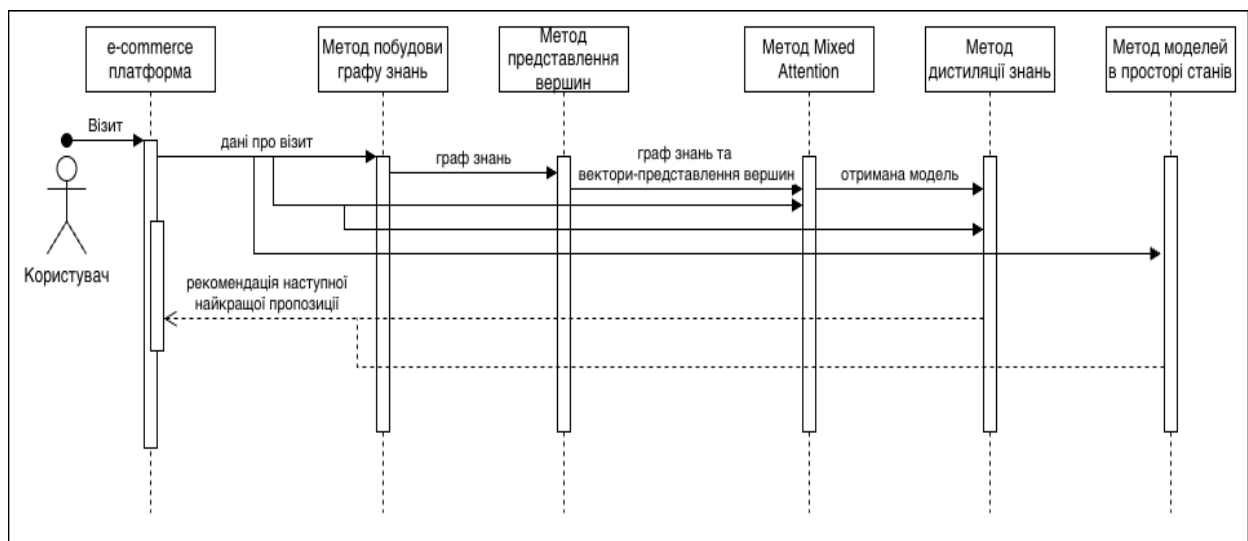


Рисунок 3.7 – Етапи методології створення гібридних рекомендаційних систем на основі архітектур глибокого навчання та динамічних вкладень графів з метою розв’язку задачі видачі наступної найкращої пропозиції

3.7 Метрики оцінювання якості прогнозів рекомендаційної системи

Для оцінок ефективності рекомендаційних систем зазвичай використовують метрики ранжування, оскільки важливим є не лише здатність РС розуміти, що є релевантним для користувача, але й вміння системи правильно впорядкувати результати, зважаючи на наявні довгострокові та короткострокові користувацькі уподобання.

Серед множини метрик ранжування, дві з них постали стандартом для оцінювання якості ранжування в задачах рекомендаційних систем та інформаційного пошуку – це середня точність в середньому (mean average precision, MAP) та нормалізований дисконтований загальний приріст (normalized discounted cumulative gain, NDCG) [3-5, 9, 87-89].

Mean Average Precision походить з галузі інформаційного пошуку, де вона вперше була представлена та застосована для оцінювання систем ранжування документів [89]. Метрика покликана вирішити фундаментальне обмеження стандартної метрики точності: вона розглядає всі позиції в ранжованому списку рівнозначно, ігноруючи той факт, що користувачі переважно фокусуються на результатах, що видаються в першу чергу.

Перед визначенням середньої точності необхідно визначити метрику точності (precision) на позиції k :

$$\text{Precision}@k = \frac{|\{i:i \in R_K \cap \text{Rel}\}|}{k},$$

де R_K представляє top- k рекомендації, а Rel позначає множину релевантних елементів. Precision@ k вимірює частку релевантних елементів серед перших k рекомендацій.

Далі для окремого користувача u середня точність (average precision, AP) обчислює середнє значень точності, розрахованих на кожній позиції, де з'являється релевантний елемент:

$$AP_u = \frac{1}{m_u} \sum_{k=1}^n \text{Precision@k} \cdot \text{rel}(k),$$

де n – загальна кількість рекомендованих елементів, m_u – кількість релевантних елементів для користувача u у повному каталозі, а $\text{rel}(k) \in \{0,1\}$ вказує, чи є елемент на позиції k релевантним.

Ключове спостереження полягає в тому, що AP обчислює метрику точності лише на позиціях, що містять релевантні елементи, таким чином підкреслюючи раннє видобування (появу серед кандидатів в рекомендацію) цієї позиції.

Якщо релевантні елементи з'являються рано, вони вносять високі значення precision ; якщо вони з'являються пізно, внесок зменшується попередніми нерелевантними елементами. Нарешті MAP агрегує AP по всіх користувачах:

$$MAP = \frac{1}{|U|} \sum_{u \in U} AP_u,$$

де U представляє множину користувачів. Це усереднення забезпечує єдину скалярну метрику, що характеризує загальну продуктивність системи.

Наступна метрика, нормалізований дисконтований кумулятивний (загальний) приріст, в свою чергу адресує основне обмеження MAP – ігнорування цієї метрикою позиції релевантності елементів вже в самому топі [88].

У реальних застосуваннях рекомендаційних систем досліджувані елементи (наприклад, товари) часто мають різні ступені релевантності для користувача. Наприклад, у рекомендації фільмів користувач може оцінити деякі фільми на 5 зірок (високо релевантні), а інші – на 3 зірки (помірно релевантні, або не варті уваги).

Основою $NDCG$ є метрика кумулятивного приросту (*cumulative gain*), що сумує показники релевантності (рейтинги) по ранжованих позиціях:

$$CG_p = \sum_{i=1}^p rel_i,$$

де rel_i – релевантність елемента на позиції i . CG розглядає всі позиції рівнозначно, що не відображає той факт, що користувачі фокусуються переважно на перших елементах в ранжованому списку.

Для того, щоб виправити даний недолік, необхідно ввести «штрафування» за розміщення релевантних елементів нижче за нерелевантні, що й виконує метрика дисконтування (discounted cumulative gain, DCG) [88]:

$$DCG_p = \sum_{i=1}^p \frac{rel_i}{\log_2(i+1)}.$$

Значення DCG залежить від довжини ранжування та розподілу рейтингів, що може суттєво відрізнятися для кожного користувача та від контексту. Для подолання цієї перешкоди виконується нормалізація DCG шляхом ділення на «ідеальне» значення DCG (IDCG) – максимально можливе значення даної метрики, що досягається, коли всі елементи відсортовані за зменшенням релевантності:

$$NDCG_p = \frac{DCG_p}{IDCG_p},$$

де

$$IDCG_p = \sum_{i=1}^{|REL|} \frac{rel_i^{\text{sorted}}}{\log_2(i+1)}.$$

Також, є важливим швидкодія рекомендаційних систем, оскільки час видачі релевантної пропозиції користувачеві складається з часу відправки даних про візит даного користувача, часу обробки даних рекомендаційною системою та часу повернення результату користувачеві, пропонується ввести метрику, що

відслідковувала би час обробки даних рекомендаційною системою, тобто час інфересу.

Проте, враховуючи те, що запропоновані у межах цього дослідження програмні реалізації гібридних рекомендаційних систем є моделями ранжування кандидатів в наступну найкращу пропозицію, дану метрику, якій надана назва «середній час інференсу» (average time of inference, ATI), також пропонується ввести як ранжувальну:

$$ATI@k = \frac{\sum_u \{\text{час, за який надано } k \text{ рекомендацій користувачу } u\}}{|U|}.$$

Саме метрика ATI@k має на меті демонстрації різниці в роботі всіх створених у межах запропонованої методології рішень, в умовах, наближених до продакшн-умов великих бізнесів.

Висновки до розділу 3

В цьому розділі здобувачем вперше представлено методологію, що полягає у поєднанні сильних сторін методів колаборативної фільтрації та методів побудови рекомендаційних систем на основі контенту, оскільки КФ враховує схожість користувачів між собою, тим самим розширюючи перелік можливих елементів, над якими проводиться оцінювання релевантності, а РС на основі контенту фокусується власне на вирішенні задачі видачі наступної найкращої пропозиції, що є ключовою в межах цього дослідження.

Насамперед, в межах запропонованої методології було описано метод побудови графу знань для представлення сесій користувача. У рамках даного методу було розглянуто перехід від табличного подання даних про поточну сесію користувача до графових структур, де користувачі, елементи, категорії та атрибути інтерпретуються як вузли, а взаємодії та інші відношення – як ребра. Основною конструкцією виступає дводольний граф «користувач-елемент» та його узагальнення у вигляді графів знань, що дозволяють моделювати складні

залежності між користувачами, товарами, сесіями та контекстом. Для побудови ваг у таких графах було запропоновано статистичні підходи на основі PMI та TF-IDF, а також їх гібридну комбінацію, яка дозволяє краще виділяти релевантні зв'язки між елементами. Додатково було розглянуто використання асоціативних правил із метриками підтримки та підйому для формування спрямованого графа причинно-наслідкових зв'язків між товарами. Окремо підкреслено, що запропоновані алгоритми мають хорошу масштабованість, підтримують розподілені обчислення й інкрементальні оновлення, що робить метод побудови графа знань придатним для роботи з великими потоками сесійних даних.

Другим кроком даного розділу було розглянуто методи побудови нейронних мереж для обробки графових даних у рекомендаційних системах, зосереджені на модифікаціях згорткових архітектур та механізмів уваги. Спочатку детально описано застосування графових згорткових мереж (GCN), основним призначенням яких є рекурсивна агрегація інформації з локального оточення вузлів у дводольному графі «користувач–елемент». У такому графі вершини представляють користувачів та елементи, а ребра кодують їхні взаємодії; GCN поетапно оновлює ембеддинги вузлів, поширюючи ознаки від сусідів першого, другого і наступних порядків, що особливо корисно у розріджених наборах даних. Окремо розглянуто спрощену архітектуру LightGCN, яка відмовляється від нелінійних активацій та складних перетворень ознак, що є достатнім для задач колаборативної фільтрації, де рекомендації зводяться до обчислення подібності представлень «користувач–елемент»; фінальне представлення формується як лінійна комбінація шарів графової згортки. Далі було проаналізовано мережі уваги над графами (GAT), які усувають обмеження класичних GCN завдяки маскованому механізму само-уваги, що навчає індивідуальні ваги для кожного сусіда та дозволяє моделі адаптивно зважувати їхній внесок при формуванні нового представлення вузла.

У межах дисертаційного дослідження було запропоновано модифікацію моделей механізму уваги, що полягає у одночасному враховуванні структури графа знань та послідовостей взаємодій у сесії користувача для передбачення

наступної найкращої пропозиції, що набув найменування змішаної уваги. Змішана увага вводить семантичні проєкції графа знань за типами відношень, використовує окремі багатоголові блоки уваги для послідовностей елементів, властивостей взаємодій і локальної топології графа, а потім об'єднує їхні виходи конкатенацією. Такий підхід забезпечує динамічне зважування внеску кожного елемента й типу зв'язку в контексті конкретної сесії, а також дозволяє гнучко модифікувати ембеддинги відповідно до режиму взаємодії, що робить метод ефективним для роботи з гетерогенними графами знань і послідовними рекомендаціями.

Наступним кроком було розглянуто подальший розвиток методу дистиляції знань для послідовних рекомендаційних систем, покликаний зменшити обчислювальні витрати графових моделей, що одночасно використовують сесійні дані та графи знань. Дистиляція знань інтерпретується як передача «прихованого» знання від потужної моделі-вчителя до компактної моделі-студента за допомогою масштабованого softmax з температурою, що наближає розподіл ймовірностей вихідних міток моделі-вчителя до рівномірного, інадаючи змогу вивчати структуру простору класів. У роботі підкреслено переваги підходу – швидшу збіжність, можливість навчання на синтетичних даних, архітектурну незалежність і потенційне збереження точності при значному скороченні розміру моделі – та показано, що оптимальні значення температури залежать від задачі, причому занадто високі або низькі значення погіршують якість навчання. У контексті проведеного дослідження було обрано конфігурацію, близьку до «класичного» навчання з учителем, яка забезпечила найкращу точність моделі-студента й надала змогу інкапсулювати знання про структуру графа знань у вектори-вкладення, що використовуються для зіставлення елементів поточної сесії з довгостроковими вподобаннями без прямого обходу графа під час інференсу.

Згодом у межах розділу був проведений огляд архітектур нейронних мереж на основі Neural ODE і перспективи їхнього застосування з метою побудови гібридної рекомендаційної системи. Шляхом проведення аналізу таких моделей

було продемонстровано, що інтерпретація кожного процесу, що змінює свої характеристики протягом часу, як системи звичайних диференціальних рівнянь надає змогу моделювати еволюцію прихованих станів користувачів та товарів у неперервному часі, без необхідності дискретизації часових проміжків. Проте, незважаючи на перспективність використання Neural ODE для моделювання часового ряду з метою інтерполяції рейтингу товару у користувача та прогнозу попиту на асортимент платформи, нещодавні дослідження показують надзвичайно велику часову та просторову складність такого методу.

Заключним етапом методології, що описаний у цьому розділі, представлено новий метод побудови гібридної рекомендаційної систем, які функціонують на основі моделей у просторі станів з використанням операторів поліноміальних проекцій високого порядку (HiPPO-операторів).

У межах цього розділу встановлено, що використання моделей на основі HiPPO-операторів робить можливим подолання обмеження моделей Neural ODE, та досягнення лінійної обчислювальної складності при збереженні високої точності прогнозування. HiPPO-моделі простору стану дозволяють ефективно стискати інформацію про довгі послідовності шляхом відображення цільової функції в ортонормальний базис, що ґрунтується на видозмінених поліномах Лежандра.

Також у межах цього розділу встановлено, що в контексті систем з неявним зворотнім зв'язком, якою є типова платформа електронної комерції, де дані про взаємодії можуть складатися лише з переглядів чи покупок товарів без явних числових оцінок уподобання, можна зробити припущення щодо можливості неявного моделювання графу знань лише на основі даних візитів користувачів, тобто вбудовування логіки пошуку по такому графу всередину латентного простору. Цей процес досягається шляхом експлуатації властивостей негомогенних пуасонівських процесів, які дозволяють моделювати часові закономірності у появі спільних елементів у послідовностях користувачів. Таким чином, замість явної побудови графу знань експертами предметної області, система може автоматично вивчати структуру знань з поведінкових даних, що

значно мінімізує людський фактор та робить запропоновану методологію масштабованою для великих каталогів товарів.

Також, у цьому розділі наведено та проаналізовано основні метрики оцінювання послідовних рекомендаційних систем. Mean Average Precision (MAP) вимірює точність рекомендацій шляхом усереднення точності при урахуванні кожного релевантного товару в ранжованому списку для кожного користувача. Проте, дана метрика демонструє нечутливість до позиції релевантних товарів, що є її суттєвим обмеженням.

У свою чергу, Normalized Discounted Cumulative Gain (NDCG) являє собою метрику, що враховує релевантність товарів та їхню позицію у ранжованому списку через застосування логарифмічного дисконтування, що значно знижує вагу товарів, розташованих у нижній частині списку. NDCG нормалізується відносно «ідеального» ранжування, що забезпечує можливість порівняння якості рекомендацій між системами з різними розмірами каталогів та розподілами релевантних товарів.

Також, у межах дослідження вперше запропоновано використання метрики *average time of inference*, що покликана дослідити швидкодію рекомендаційної системи, оскільки час власне видачі рекомендації є критично важливим в продакшн-середовищах.

РОЗДІЛ 4 ЕКСПЕРИМЕНТАЛЬНИЙ АНАЛІЗ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ ТА ДИНАМІЧНИХ ВКЛАДЕНЬ ГРАФІВ

Усіма запропонованими моделями використовується парадигма two-tower архітектури, що забезпечує ефективну генерацію кандидатів та масштабування системи завдяки розділенню моделей, що обробляють інформацію про користувачів та товарів.

4.1 Дворівнева (two-tower) архітектура рекомендаційних систем

Дворівнева архітектура є однією з основних архітектур сучасних рекомендаційних систем, що особливо зарекомендувала себе у сценаріях видачі рекомендацій серед мільйонів кандидатів [1-9, 30].

Дворівнева архітектура, як випливає з назви, складається з двох основних компонентів, що відповідальні за моделювання профілю користувача (user tower) та дослідження властивості елементів у контексті взаємодії останніх з користувачем (item tower).

User tower обробляє послідовності взаємодії користувача з елементами задля генерації векторів вкладень $e_u = f_{\text{user}}(\mathcal{H}_u, \mathcal{F}_u; \Theta_u)$, де $\mathcal{H}_u$ представляє історичну послідовність взаємодій користувача, $\mathcal{F}_u$ позначає закодовані ознаки користувача, а Θ_u – параметри функції f_{user} . У свою чергу item tower створює вектори вкладень елементів $e_i = f_{\text{item}}(\mathcal{A}_i, \mathcal{C}_i; \Theta_i)$, де $\mathcal{A}_i$ представляє атрибути елементу, а $\mathcal{C}_i$ позначає контекстуальні ознаки. Релевантність рекомендації при цьому обчислюється як $\hat{y}_{u,i} = e_u^T e_i$. Схематичне представлення дворівневої архітектури представлено на рисунку 4.1.

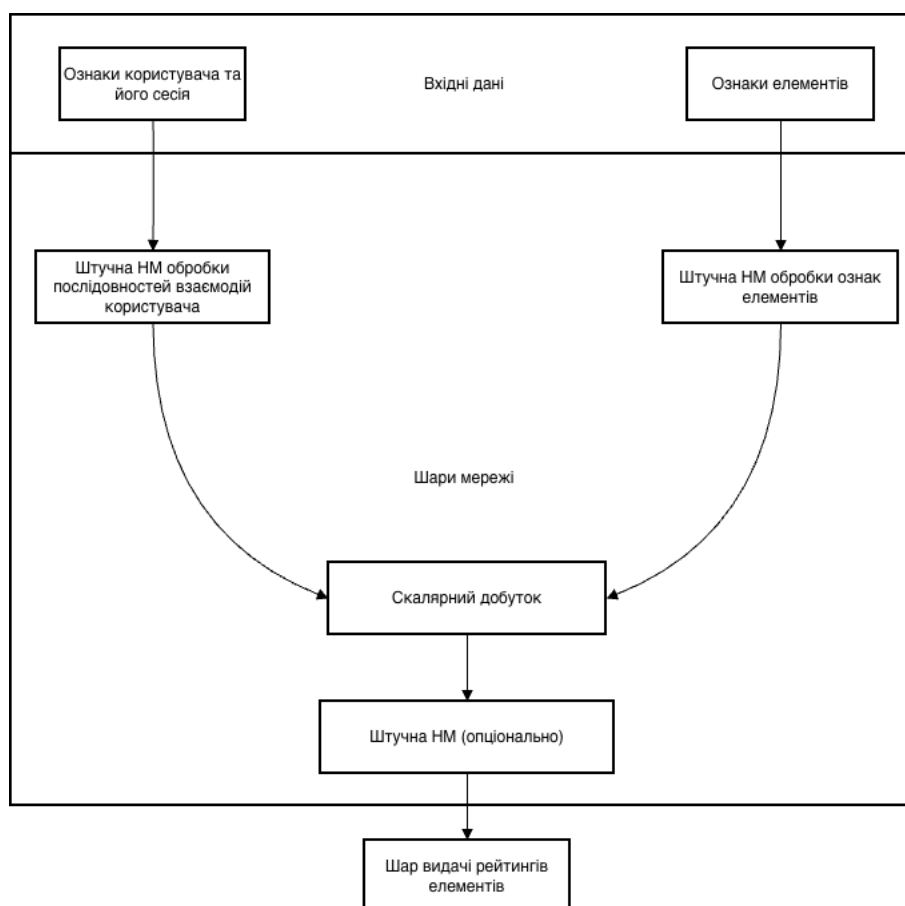


Рисунок 4.1 – Схема дворівневої (two tower) архітектури РС

4.2 Набори даних (датасети) для експериментів

Задля справедливої та неупередженої оцінки якості моделей, що є реалізаціями запропонованої методології, необхідним є обрання гетерогенних великих наборів даних, що послуговуються довірою в академічному колі. Переслідуючи цю мету, в межах цього дослідження було обрано два набори даних з домену електронної комерції – REES46 Open CDP датасет та набір даних RetailRocket.

4.2.1 Набір даних REES46 Open CDP

REES46 Open CDP датасетом представлено великомасштабний мульти-категорійний e-commerce датасет, де містяться приблизно 69 мільйонів подій по

6,5 мільйонах сесій з середньою довжиною сесії 10,355 елементів, зібраних протягом жовтня-листопада 2019 року. Датасет включає понад 50 000 унікальних продуктів, що охоплюють множинні категорії з трьома типами подій: перегляд (view), додавання до кошику (cart) та покупка (purchase). Схемою датасету включаються `user_id` (унікальний ідентифікатор користувача), `product_id` (унікальний ідентифікатор товару), `category_id` (категорія), `event_type` (тип взаємодії), `user_session` (ідентифікатор сесії) та `event_time` (часова мітка взаємодії). Сам набір даних представлений як реляційне відношення, схема якого надана у таблиці 4.1

Таблиця 4.1 – Схема набору даних REES46 OpenCDP

<i>Назва атрибуту</i>	<i>Тип атрибуту</i>	<i>Короткий опис</i>
<code>user_id</code>	unsigned bigint	Унікальний ідентифікатор користувача
<code>product_id</code>	unsigned bigint	Унікальний ідентифікатор предмету
<code>category_id</code>	unsigned bigint	Категорія предмету (напр., електроніка, одяг)
<code>event_type</code>	string	Тип взаємодії (покупка, переміщення до кошику, перегляд)
<code>user_session</code>	base64	Унікальний ідентифікатор сесії

4.2.2 Набір даних RetailRocket

Набір даних RetailRocket також покриває домен електронної комерції, проте має дещо меншу потужність.

В даному наборі налічується приблизно 9 мільйонів подій по 1,2 мільйонах сесій, причому в середньому на сесію припадає 9,5 елементів, що є схожим до OpenCDP показником.

Зібраний протягом червня-жовтня 2019, набір даних має інформацію щодо 24 000 унікальних продуктів у різноманітних категоріях. Схема набору даних наведена в таблиці 4.2.

Таблиця 4.2 – Схема набору даних RetailRocket

<i>Назва атрибуту</i>	<i>Тип атрибуту</i>	<i>Короткий опис</i>
vistorId	unsigned bigint	Унікальний ідентифікатор користувача
itemId	unsigned bigint	Унікальний ідентифікатор предмету
categoryId	unsigned bigint	Категорія предмету (напр., електроніка, одяг)
event	string	Тип взаємодії (покупка, переміщення до кошику, перегляд)
timestamp	unsigned bigint	Часова мітка події
transactionId	base64	Унікальний ідентифікатор сесії користувача

4.3 Конструкція графа знань на основі даних взаємодій (сесій)

Першою ціллю даного дослідження є створення методу автоматизованого синтезу графа знань безпосередньо з даних візитів користувачів [5].

В сконструйованому за допомогою метрики PMI на REES46 наборі даних графі знань містяться 42 384 вузли з 1 247 532 ребрами, причому середній степінь вершини дорівнює 29,4.

TF-IDF зважуванням значно зменшується щільність графу завдяки зменшенню кількості ребер до 748 553, причому кількість вузлів залишається сталою.

Аналізом правил асоціацій за допомогою дерев FP-росту ідентифікуються 65 000 правил по типу імплікації «якщо елемент i_a є в кошику, то елемент i_b також є» ($i_a \rightarrow i_b$) з середніми показниками впевненості 0,342 та підйому 2,87.

Для одержання векторів-вкладень вершин одержаного графу використовується алгоритм Node2Vec з такою конфігурацією:

- розмір вектору вкладень – 64;
- константа «повернення» $p = 1$;
- константа «дослідження» $q = 1$;
- довжина вікна для генерації skip-gram – 10;
- навчання алгоритму відбувається стохастичним градієнтним спуском з кроком 0.01.

В якості порівняння підходів було використано дві дворівневі моделі, що відрізняються лише моделлю обробки сесії користувача. Базова модель має в якості цієї моделі глибоку мережу LSTM (рис. 4.2)

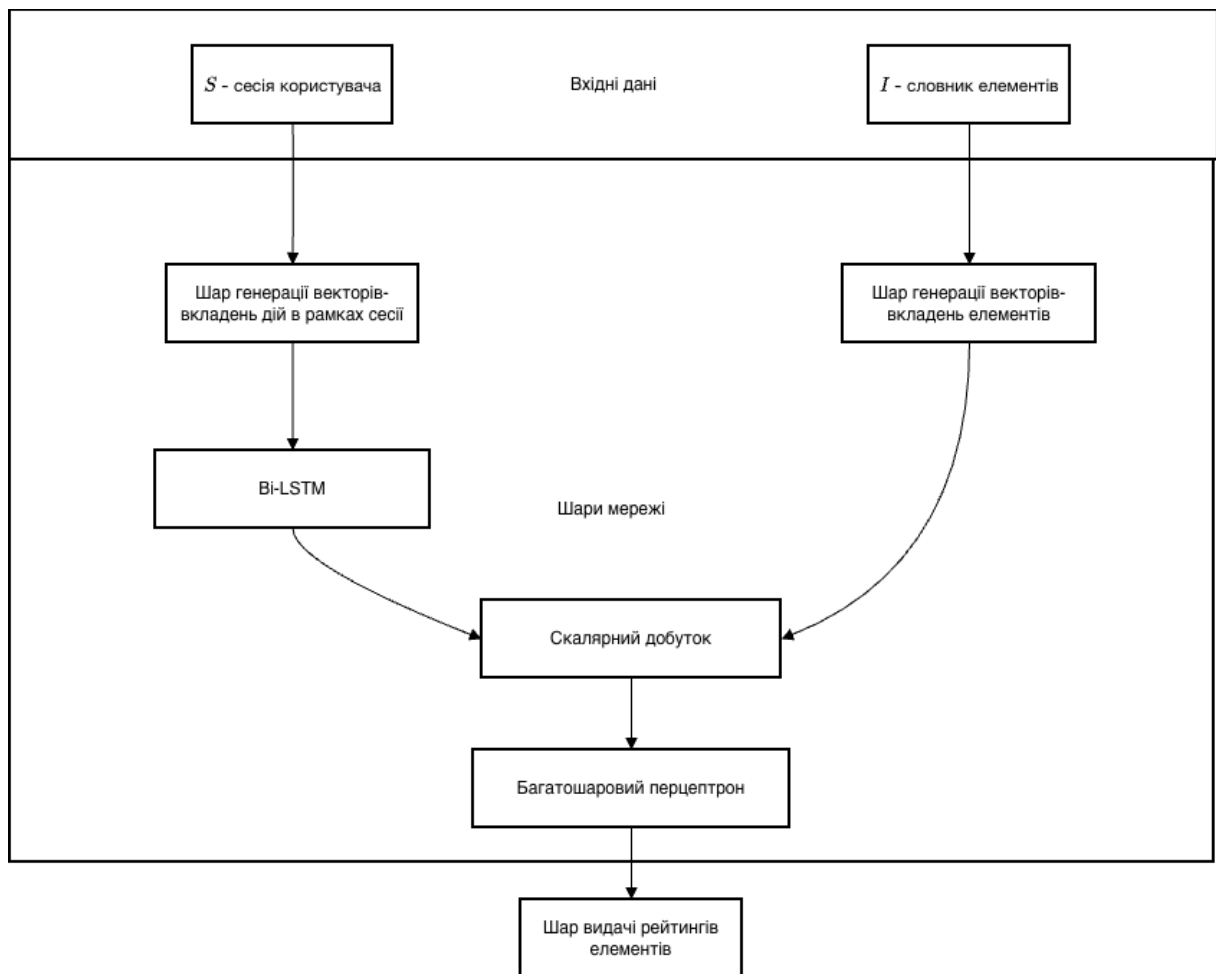


Рисунок 4.2 – Схема запропонованої моделі LSTM

В свою чергу, на рисунку 4.3 запропоновану графову нейронну мережу з механізмом уваги що охоплює послідовність візитів, а також досліджує всіх сусідів першого порядку кожної вершини, відвіданої в сесії. Модель названо запропонованою GNN [5].

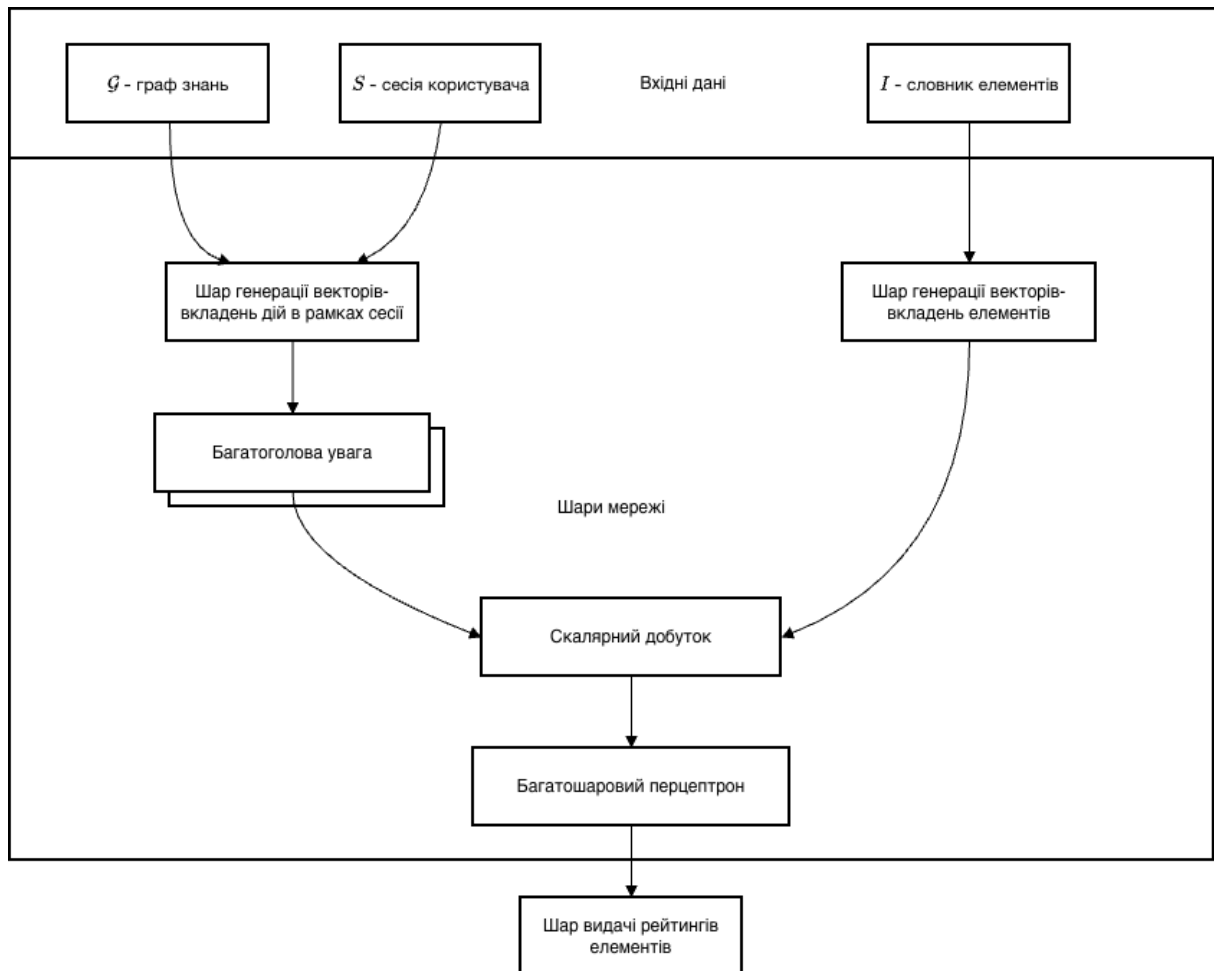


Рисунок 4.3 – Схема рекомендаційної системи на основі запропонованої графової нейронної мережі (запропонованої GNN)

Результати оцінювання ефективності моделювання графів знань на запропонованих моделях наведені а таблиці 4.3

Таблиця 4.3 – Оцінювання ефективності створених графів знань в задачі послідовних рекомендацій

Модель	$MAP@1$	$MAP@10$	$NDCG@1$	$NDCG@10$
LSTM	0.0389	0.0937	0.0388	0.1217
запропонована GNN + PMI	0.1442	0.2551	0.144	0.2988
запропонована GNN + TF-IDF	0.0844	0.1622	0.0845	0.1960
запропонована GNN + FP-Growth	0.0399	0.0887	0.0378	0.1150

Як видно з таблиці, графи, створені за допомогою PMI метрики демонструють найкращі результати, особливо в порівнянні з LSTM моделлю, що показує приріст у більш ніж три рази відносно метрик $MAP@1$ та $NDCG@1$, майже троекратне покращення у $MAP@10$ та майже вдвічі більший приріст $NDCG@10$. Метрики моделі, що використовує PMI, є найкращими в порівнянні з іншими методами побудов графу знань, завдяки великій кількості зв'язків між вершинами, проте недоліком є час навчання моделі – приблизно 10 хвилин за епоху – 610 секунд, що може бути неприйнятним для продакшн-середовищ, де графи можуть бути значно більші та щільніші.

TF-IDF підходом забезпечується компроміс продуктивність-ефективність, що досягається майже двократним покращенням у $MAP@1$ та $MAP@10$ порівняно з LSTM, при цьому навчання збільшене до 2 хвилин за епоху.

Дерева FP-росту конфігурацією демонструється мінімальне покращення відносно LSTM, що вказує на неефективність використання такого алгоритму над великим набором даних, що охоплює кілька доменів.

4.4 Дослідження методу дистиляції знань для покращення швидкості відповіді моделей

Після виконаних висновків щодо ефективності PMI для створення графів знань, постала задача використання гетерогенних графів знань для подальшого

покращення роботи РС, таким чином була запропонована модель MixedAttention (рис. 4.4) [5]:

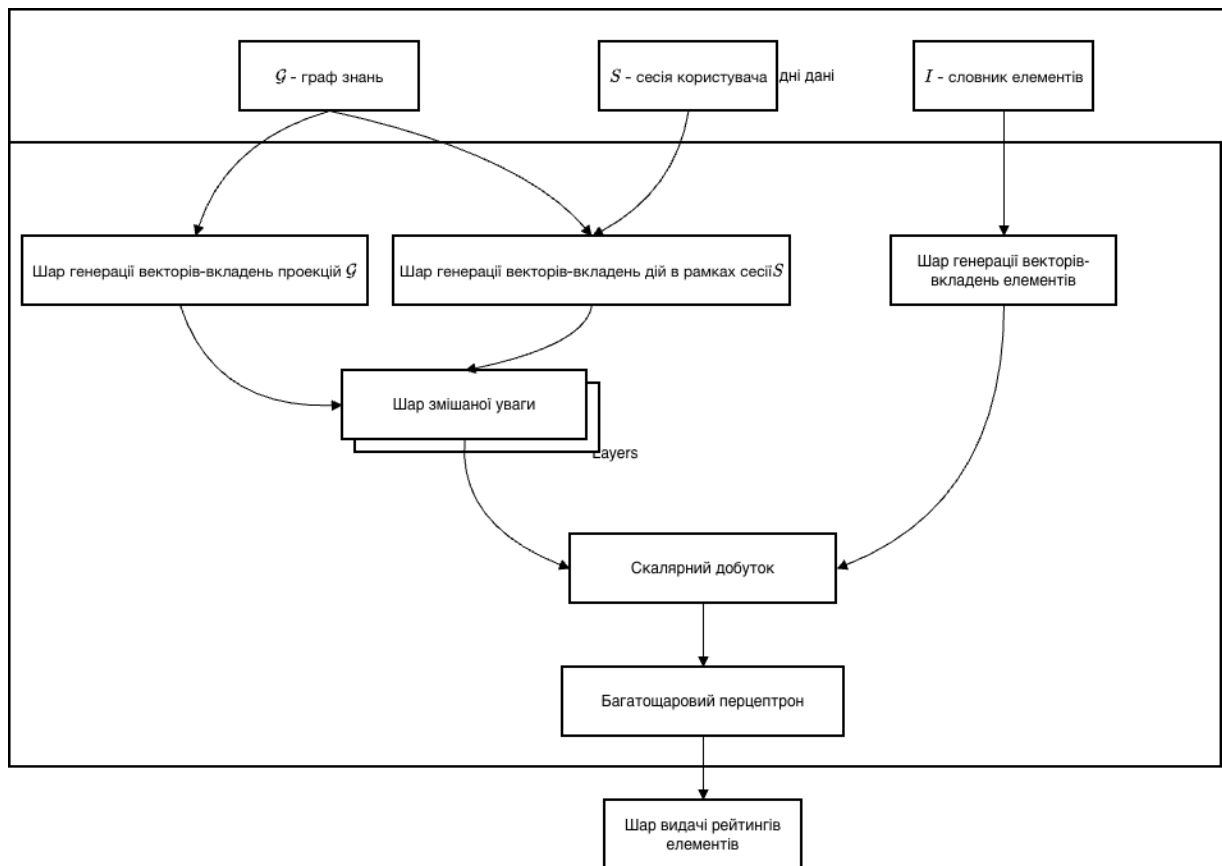


Рисунок 4.4 – Схема запропонованої моделі MixedAttention

Порівнюючи з моделлю LSTM отримані наступні результати (табл. 4.4):

Таблиця 4.4 – Оцінювання ефективності Mixed Attention в задачі послідовних рекомендацій

Модель	$MAP@1$	$MAP@10$	$NDCG@1$	$NDCG@10$
LSTM	0.0389	0.0937	0.0388	0.1217
Mixed Attention	0.1769	0.3003	0.1755	0.3487
Mixed Attention + Node2Vec	0.2	0.331	0.2025	0.3795

Таким чином отримано найкращу модель, що досягається не лише використанням запропонованого підходу Mixed Attention, а й додатковим попереднім навчанням векторів-вкладень елементів за допомогою алгоритму Node2Vec. Додатковою перевагою є відносно швидка збіжність моделей Mixed Attention, що полягає у максимізації метрик MAP (рис. 4.5) та NDCG (рис. 4.6) за кілька епох.

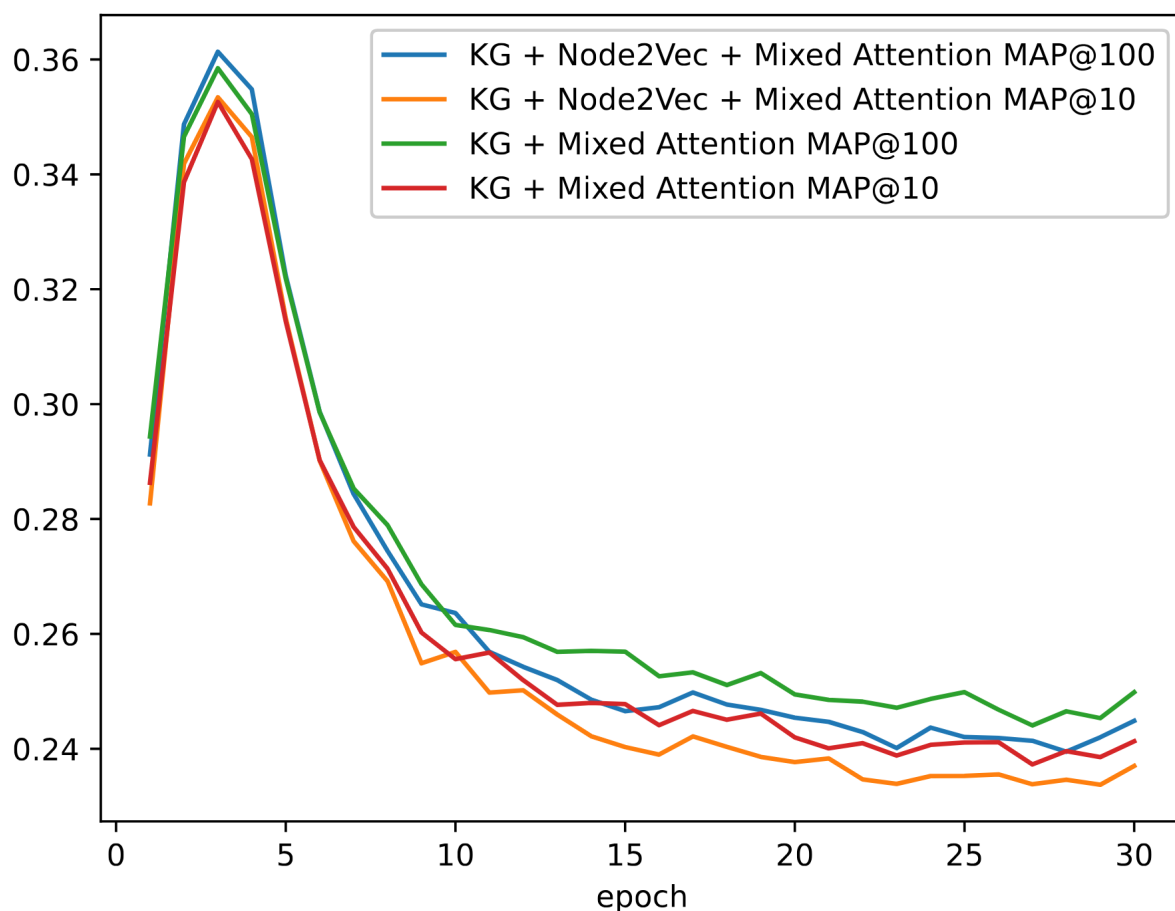


Рисунок 4.5 – Еволюція метрики MAP в Mixed Attention моделях

Як видно з рисунку 4.5. максимальне значення метрики MAP досягається при трьох епохах навчання, що є одним з доказів швидкої збіжності моделей, заснованих на представленому методі змішаної уваги.

Той самий висновок можна зробити, дивлячись на рисунок 4.5.

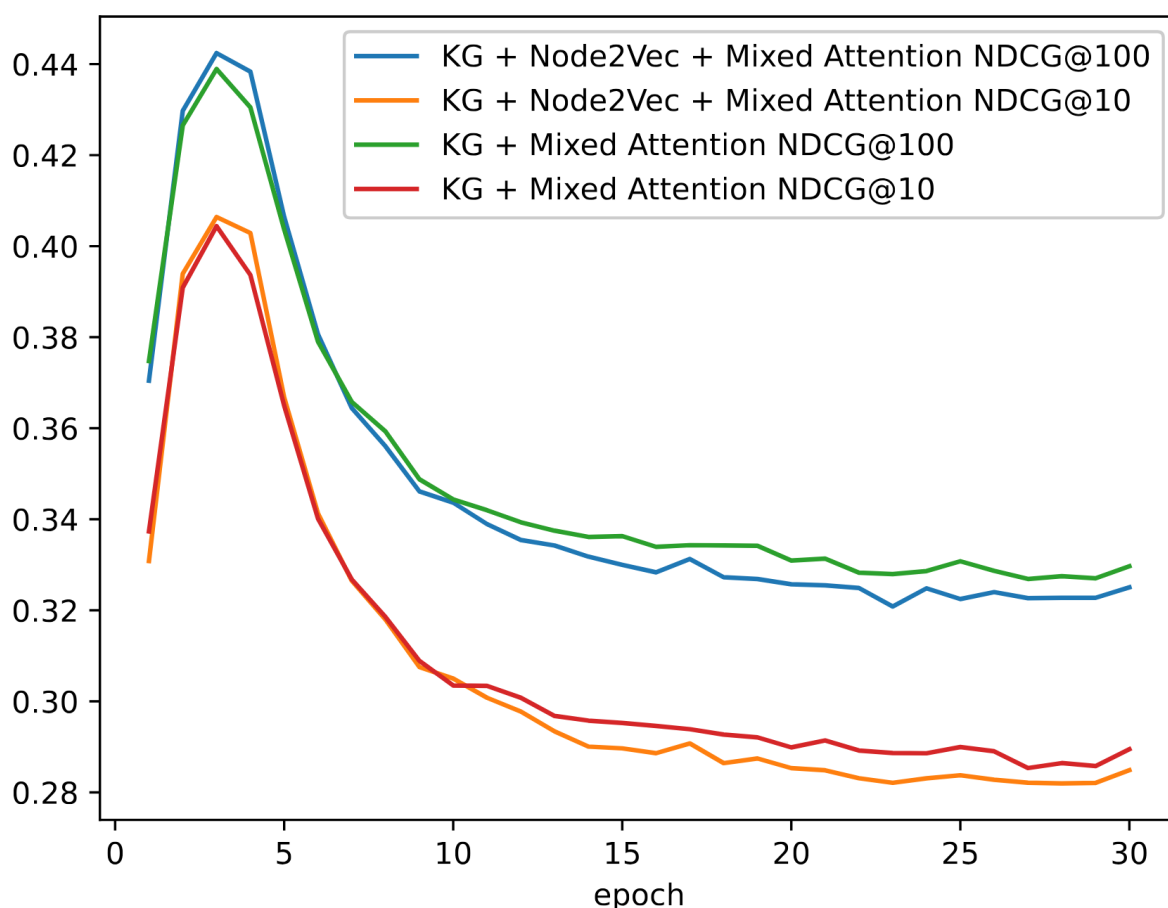


Рисунок 4.6 – Еволюція метрики NDCG в Mixed Attention моделях

Також, цікавим є погляд на вектори-вкладення товарів, в просторі яких, відповідно до запиту користувача, виконується інформований пошук. Передбачувано, LSTM модель, що навчалася виключно на даних візитів, моделює вектори-вкладення по типу «взаємозамінні товари в чеку», оскільки елементи, що вели до схожих наступних пропозицій, групувалися разом. Приклад найближчих 10 векторів до вектору «спортивні кеди Nike» зображені на рисунку 4.7.

Не дивлячись на те, що 5 з 10 рекомендацій є елементи з тієї ж категорії одягу чи взуття, інші 5 рекомендацій займає електроніка, особливо комп'ютерна периферія, що не можна пояснити очевидними асоціаціями.

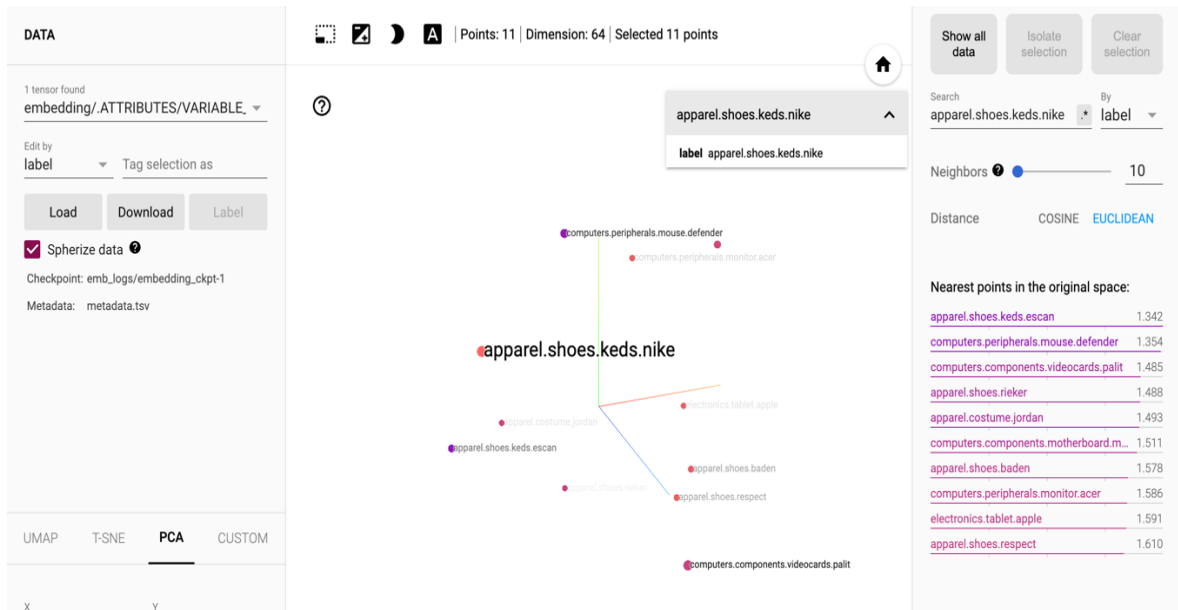


Рисунок 4.7 – Найближчі вектори до вектору «бігові кеди Nike» за моделлю LSTM

На відміну від результатів LSTM моделі, запропонована модель на основі механізму уваги видає багатообіцяючі результати, ставлячи поруч з вектором «бігові кеди Nike» не спортивне взуття, а фітнес-годинники, кухонні ваги, велосипеди та інші спортивні товари та електроніку, що чітко вказує на приналежність векторів темі «здоровий образ життя» одержаного графу знань (рис. 4.8).

Проте, модель LSTM є набагато швидшою за Mixed Attention за рахунок відносної простоти своєї архітектури, та надає top-100 рекомендацій за 1.67 секунди (метрика $ATI@100$), в той час як моделі Mixed Attention – за 2.7 секунд [5]. Таким чином існує необхідність оптимізації моделі відносно часу відповіді, зберігаючи точність. Для цього була застосована дистилляція знань (рис. 3.4).

Оскільки дистилляція знань використовує масштабовані softmax перетворення, а також залежить від коефіцієнту впливу дивергенції Кульбака-Лейблера α та параметру «температури» τ , у межах цього дослідження було виконано низку експериментів з підбору потрібних значень α та τ .

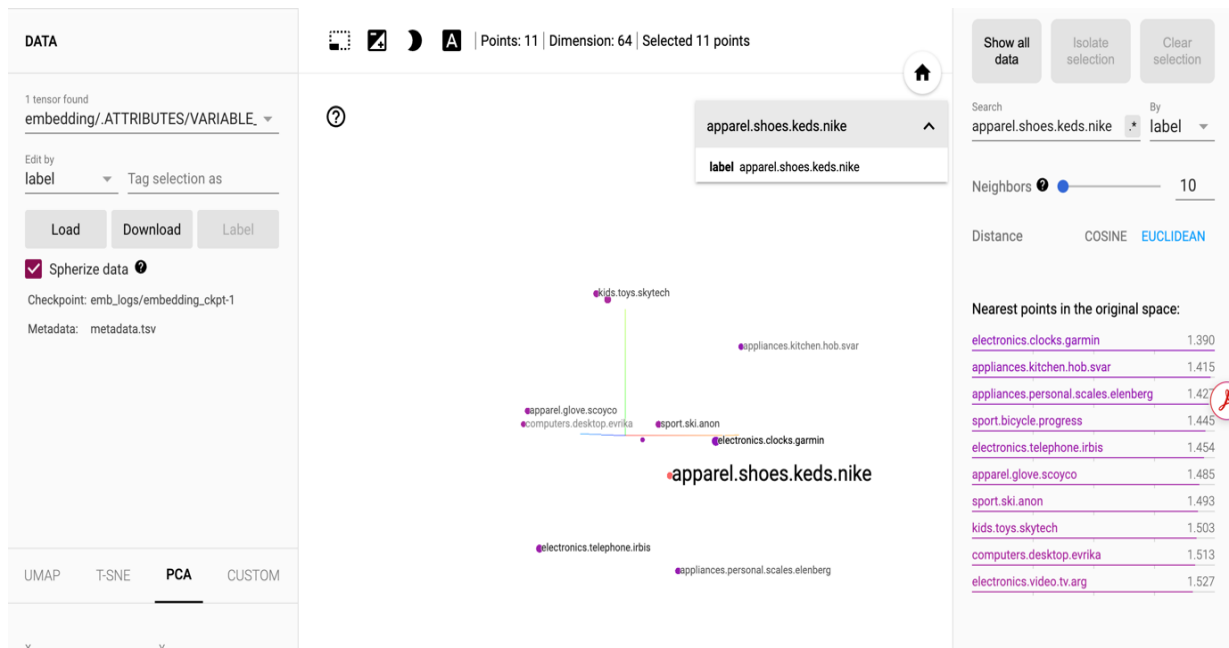


Рисунок 4.8 – Найближчі вектори до вектору «бігові кеди Nike» за моделлю Mixed Attention

Експерименти показали неочевидний результат – найкраще себе показала модель-студент, що навчалася при значенні температури рівній одиниці. Цей результат, хоч і суперечить дослідженню методу дистиляції знань в контексті задач класифікації, проте є пояснюваним в контексті даного дисертаційного дослідження. На відміну від наведених в оригінальному дослідженні задачах, введені РС на основі методу змішаної уваги використовують додатково графи знань, які модель-студент «ігнорує». Саме тому те саме «приховане знання» вже міститься у значеннях уваги (attention scores), які модель-студент і покликана апроксимувати.

Результати проведеної процедури наведені в таблиці 4.5.

Таким чином, оптимальною конфігурацією ($\tau = 1.0$, $\alpha = 1.0$) досягається MAP@1 0.1045 та NDCG@10 0.2500, що все ще значно краще LSTM моделі. ATI@100 на всіх конфігурація дорівнює приблизно 1.51 секунд, що дещо швидше, порівняно з LSTM baseline (1.67 секунд).

Таблиця 4.5 – Дослідження ефективності фреймворку дистиляції знань
відносно масштабування виходів Mixed Attention моделі

τ	α	$MAP@1$	$MAP@10$	$NDCG@1$	$NDCG@10$
0.5	0.1	0.0105	0.0368	0.0105	0.0574
0.5	0.5	0.0809	0.1769	0.0803	0.2204
0.5	0.7	0.0865	0.1856	0.0869	0.2333
0.5	1.0	0.1006	0.2004	0.1010	0.2473
1.0	0.1	0.0064	0.0348	0.0064	0.0564
1.0	0.5	0.0820	0.1735	0.0811	0.2182
1.0	0.7	0.0967	0.1894	0.0920	0.2351
1.0	1.0	0.1045	0.2039	0.1037	0.2500
2.0	0.1	0.0082	0.0352	0.0082	0.0543
2.0	0.5	0.0773	0.1736	0.0783	0.2187
2.0	0.7	0.0977	0.1898	0.0959	0.2344
2.0	1.0	0.1008	0.1980	0.0998	0.2472
5.0	0.1	0.0059	0.0329	0.0068	0.0520
5.0	0.5	0.0801	0.1721	0.0812	0.2162
5.0	0.7	0.0951	0.1887	0.0920	0.2344
5.0	1.0	0.0990	0.2011	0.0982	0.2485

Отже, можна дійти висновку, що фреймворк дистиляції знань є ефективним і в контексті задачі видачі персоналізованих рекомендацій найкращих наступних пропозицій.

4.5 Аналіз моделей простору станів для задачі видачі послідовних рекомендацій

В якості альтернативи до застосування фреймворку дистиляції знань, була побудована модель, що використовує виключно оператори поліноміальної

проекції високого порядку та працює напряду з сесійними даними, ліквідовуючи необхідність попереднього конструювання графу знань, зводячи всю РС до побудови та оптимізації однієї дворівневої нейронної мережі.

Спочатку дану систему було порівняно з попередніми на REES46 Open CDP датасеті (табл. 4.6).

Таблиця 4.6 – Оцінювання ефективності створених моделей в задачі послідовних рекомендацій

<i>Модель</i>	<i>MAP@1</i>	<i>MAP@10</i>	<i>NDCG@1</i>	<i>NDCG@10</i>
LSTM	0.0389	0.0937	0.0388	0.1217
запропонована GNN + PMI	0.1442	0.2551	0.144	0.2988
запропонована GNN + TF-IDF	0.0844	0.1622	0.0845	0.1960
запропонована GNN + FP-Growth	0.0399	0.0887	0.0378	0.1150
PMI - Mixed Attention	0.1769	0.3003	0.1755	0.3487
PMI - Mixed Attention + Node2Vec	0.2	0.331	0.2025	0.3795
HiPPO SSM	0.1719	0.2979	0.1556	0.3562

Як видно з таблиці вище, моделі простору станів поступаються своєю точністю виключно РС, що використовувала Mixed Attention, синтезовані графи знань за допомогою PMI та попередньо оптимізованими векторами вкладень за допомогою Node2Vec. Причому, ATI@100 HiPPO SSM моделі становить лише 1.92 секунд, що лише трохи повільніше моделі LSTM (1.67 секунд).

Проте, варто зазначити, що модель простору станів досить добре справляється з задачею неявного моделювання графу знань (рис. 4.9), оскільки простір векторів вкладень відображає причинно-наслідкові зв'язки, схожі на ті, що виходили в моделі Mixed Attention. Зокрема, послуговуючись вектором, що відповідає «біговим кедам Nike», отримано найближчі вектори-вкладення товарів, що відповідають фітнес-годинникам, навушникам, та малим

аудіоколонкам, що можна віднести до категорії «фітнес та захоплення» графу знань.

Проте, не зовсім влучними є рекомендації акумуляторного пирососу, дитячих іграшок та телевізору у якості найкращої наступної пропозиції, що можна віднести до категорії «подарунки», проте останні асоціативні зв'язки викликають долю скепсису і можуть бути пояснені великою долею транзакцій, що відбувалися напередодні свят (таких як «чорна п'ятниця», Різдво, Новий рік).

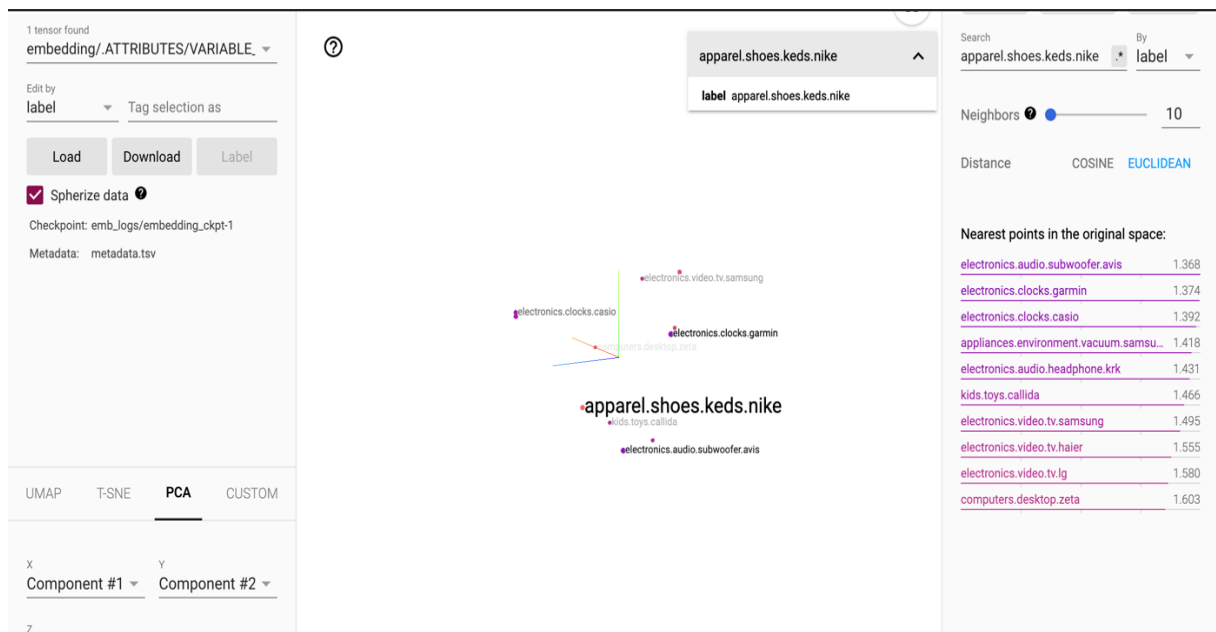


Рисунок 4.9 – Найближчі вектори до вектору «бігові кеди Nike» за моделлю HiPPO-SSM

Для більш репрезентативної валідації цього фреймворку, моделі PMI - Mixed Attention + Node2Vec та HiPPO SSM були протестовані на Retail Rocket наборі даних та порівняні з найпопулярнішими рішеннями, оглянутими в межах цього дослідження – SASRec [63], GRU4Rec [65] та BERT4Rec [67].

Результати порівняння показано в таблиці 4.7.

Таблиця 4.7 – Оцінювання ефективності створених моделей на наборі даних RetailRocket

Модель	<i>ATI@100</i>	<i>MAP@20</i>	<i>NDCG@10</i>
GRU4Rec [65]	-	0.0272	0.152
SASRec [63]	-	0.0312	0.240
BERT4Rec [67]	-	-	0.197
PMI - Mixed Attention + Node2Vec	0.63	0.0464	0.2722
HiPPO SSM	0.8	0.1580	0.2898

Експериментальними методами демонструється, що HiPPO SSM забезпечується найкраща продуктивність через лінійну складність моделі та вдалий вибір стратегії моделювання процесів взаємодій користувача з кожним елементом через негомогенні пуасонівські процеси. Саме властивості пуасонівських процесів роблять їх природним інструментом для задач послідовних рекомендацій, зокрема для центральної в дисертаційному дослідженні задачі видачі наступної найкращої пропозиції.

Висновки до розділу 4

У межах розділу здійснено експериментальну перевірку розроблених методів та моделей для побудови послідовних рекомендаційних систем на основі даних електронної комерції. Об'єктом дослідження стали набори даних REES46 Open CDP та RetailRocket. Експерименти проводилися з використанням дворівневої архітектури побудови рекомендаційних систем. У ролі базової моделі використано архітектуру LSTM.

Досліджено ефективність застосування графів знань, побудованих за допомогою методів Pointwise Mutual Information (PMI), TF-IDF та FP-Growth. Встановлено, що граф знань на основі PMI забезпечує найкращі результати. Зокрема, запропонована модель GNN з PMI продемонструвала значення метрик MAP@10 на рівні 0.2551 та NDCG@10 на рівні 0.2988, що суттєво перевищує

показники базової моделі LSTM ($\text{MAP@10} = 0.0937$, $\text{NDCG@10} = 0.1217$). Метод FP-Growth виявився найменш ефективним.

Проаналізовано ефективність застосування методу на основі модифікацій моделей механізму уваги. Рекомендаційна система, що використовує розроблений метод змішаної уваги та використовує граф знань, одержаний застосуванням метрики PMI, досягла значень $\text{MAP@10} = 0.3003$ та $\text{NDCG@10} = 0.3487$. Використання вкладень Node2Vec дозволило підвищити MAP@10 до 0.331 та NDCG@10 до 0.3795.

Досліджено вплив гіперпараметрів дистиляції знань. Встановлено, що оптимальні результати досягаються при значенні температурного параметра softmax, близькому до 1.0.

Проведено аналіз моделі на основі простору станів (SSM) з використанням HiPPO-оператора. Модель HiPPO-SSM продемонструвала високу точність виданих прогнозів на наборі даних REES46 ($\text{MAP@10} = 0.2979$, $\text{NDCG@10} = 0.3562$) та перевершила за показниками цих метрик ранжування найпопулярніші існуючі рішення GRU4Rec, SASRec та BERT4Rec на наборі даних RetailRocket, досягнувши значень $\text{MAP@20} = 0.1580$ та $\text{NDCG@10} = 0.2898$.

Оцінено метрику середнього часу видачі результатів моделі (ATI). Встановлено, що модель HiPPO-SSM демонструє схожу швидкодію ($\text{ATI@100} = 1.92$ мс для REES46) порівняно з базовою моделлю LSTM ($\text{ATI@100} = 1.67$ мс на REES46) та іншими архітектурами, що підтверджує її обчислювальну ефективність.

Візуальний аналіз рекомендацій підтвердив, що розроблені моделі генерують більш релевантні пропозиції порівняно з базовою моделлю LSTM.

Результати експериментів підтверджують ефективність запропонованих підходів, які поєднують методи глибокого навчання, графи знань та моделі простору станів, демонструючи приріст точності та швидкодії в умовах реальних систем електронної комерції.

ВИСНОВКИ

Дисертаційне дослідження присвячено розробленню послідовних рекомендаційних систем для корпоративних середовищ, орієнтованих на обробку великомасштабних даних взаємодій користувачів з асортиментом товарів і послуг. Актуальність роботи зумовлена необхідністю глибокого аналізу патернів поведінки користувачів на платформах електронної комерції та/або стримінгу як ключового чинника довгострокового утримання лояльної клієнтської бази та забезпечення стабільного зростання доходів компаній.

У роботі рекомендаційні системи розглядаються як спеціалізований клас алгоритмів статистичного та машинного навчання, що моделюють взаємодію користувачів із контентом, поєднуючи врахування довгострокових індивідуальних уподобань з короткочасними флуктуаціями, зумовленими зовнішніми чинниками (маркетингові кампанії, сезонність, святкові періоди тощо).

Основним призначенням рекомендаційних систем є формування персоналізованих списків найбільш релевантних елементів на основі аналізу історії взаємодій користувача з платформою у режимі, наближеному до реального часу.

Метою цієї дисертаційної роботи є підвищення точності прогнозування наступної найкращої дії користувача, одночасно забезпечуючи швидкість обчислень, необхідну для видачі рекомендацій у режимі реального часу.

Заради досягнення мети перш за все було проведено аналіз існуючих методологій побудови рекомендаційних систем. У першому та другому розділах дисертації здійснено огляд таких усталених підходів, як методології на основі правил, знань, фільтрації на основі контенту та контексту. Додатково освітлена задача колаборативної фільтрації та наведено опис існуючих методів розв'язання задачі, в тому числі й сучасних методів на основі моделей глибинного навчання. Завершальним етапом огляду є приведення постановки задачі видачі

послідовних рекомендацій та наведення сучасних методів її розв'язку на основі нейронних мереж, зокрема рекурентних та на основі механізму уваги.

У даному огляді показано, що, незважаючи на значний прогрес у застосуванні глибинних нейронних мереж до задач видачі персоналізованих рекомендацій, чинні підходи мають низку суттєвих обмежень. Зокрема, одним із ключових недоліків є обмежена довжина послідовностей, що подаються на вхід моделям послідовної рекомендації (наприклад, на основі рекурентних нейронних мереж або трансформерів), що фактично обмежує місткість пам'яті та ускладнює коректне моделювання довготривалих залежностей у поведінці користувачів. Іншою виявленою проблемою є схильність моделей, навчених на великих масивах даних із неявним відгуком, до переваги загалом популярних елементів, що супроводжується ігноруванням індивідуальних, менш поширених об'єктів. Це призводить до зниження різноманітності рекомендацій та потенційного зменшення залученості користувачів через надмірну уніфікацію пропозицій.

Наступною виконаною задачею досягнення мети дослідження є те, що сформульовано задачу прогнозування найбільш релевантних пропозицій для користувача, обрано типи взаємодій для дослідження, зібрано необхідні статистичні дані про взаємодію користувачів з платформою, зокрема онлайн-ритейлером. Поставлену задачу прогнозування наступної найкращої пропозиції запропоновано вирішити шляхом впровадження нової методології, вперше представленій у розділі 3 даного дослідження.

У роботі запропоновано методологію синтезу гібридної рекомендаційної системи, у межах якої кожна платформа інтерпретується як мережа взаємодій, що мотивує до побудови динамічних графів знань та їх інтеграції з архітектурами глибинного навчання на основі механізмів уваги. Додатково використано процедуру дистиляції знань для спрощення архітектури моделей, а також підхід до побудови моделей у просторі станів для апроксимації нестационарних часових рядів, що описують поведінку користувачів, із застосуванням модифікованих

операторів поліноміальної проекції високого порядку та аналізу нестационарних пуасонівських процесів.

Такі нововведення, виконані автором дисертаційного дослідження покликані збільшити точність отриманих рекомендаційних систем, примножити різноманіття виданих рекомендацій та оптимізації часу видачі рекомендацій цільовому користувачеві.

Для первинної обробки даних та автоматизованої побудови графів знань запропоновано використання методів статистичного аналізу, а саме алгоритмів асоціативних правил на основі дерев FP-росту, підходу TF-IDF та статистики попарної взаємної інформації (pointwise mutual information) з метою виявлення латентних взаємозв'язків між елементами.

Наступним етапом методології є розроблення методу побудови рекомендаційних систем на основі модифікованого механізму уваги, який надає змогу одночасно досліджувати вхідну послідовність поточної сесії користувача та збагачувати отриману інформацію шляхом ітерації по згенерованому раніше графу знань. Такий підхід покликаний істотно підвищити точність рекомендацій за рахунок поєднання локального контексту сесії з глобальною структурою взаємодій у графі.

Подальший етап полягає у застосуванні дистиляції знань для спрощення архітектури моделі, розробленої на попередньому кроці, що забезпечує зменшення обчислювальної складності та скорочення часу обробки вхідних даних рекомендаційною системою при збереженні прийняттого рівня якості рекомендацій.

Завершальним компонентом запропонованої методології є новий метод, що ґрунтується на застосуванні спеціального класу моделей машинного навчання для прогнозування та обробки часових рядів і послідовностей – моделей у просторі станів з використанням операторів поліноміальної проекції високого порядку (HiPRO-операторів). У межах цього підходу робиться припущення, що кількісні дані взаємодій користувачів з елементами описуються нестационарним пуасонівським процесом; відповідно, запропонований метод

спрямовано на апроксимацію інтенсивності цього процесу та неявне моделювання породженого ним графа знань. Це, у свою чергу, надає змогу одночасно розв'язувати задачу прогнозування попиту на конкретні елементи та задачу неявного моделювання графа причинно-наслідкових зв'язків між подіями.

Для оцінювання якості рекомендацій, що генеруються системою, побудованою на основі запропонованої методології, сформовано критеріальну базу, до якої включено метрики MAP (Mean Average Precision) та NDCG (Normalized Discounted Cumulative Gain) як стандартні показники якості ранжування у задачах рекомендації. З огляду на важливість продуктивності у корпоративних застосунках, критеріальну базу доповнено показником ATI (average time of inference), який характеризує середній час відповіді системи при формуванні рекомендацій на запит. Такий підхід забезпечує комплексну оцінку як точності, так і швидкодії запропонованого рішення.

Таким чином, шляхом прикладного застосування методології було синтезовано кілька гібридних рекомендаційних систем, що вирішують задачу видачі наступних найкращих пропозицій шляхом обробки вхідних послідовностей візитів користувачів, їхніх взаємодій з елементами (товарами, послугами), а також роботи з графами знань. Дані гібридні рекомендаційні системи на основі нейронних мереж та динамічних вкладень графів були апробовані на надвеликих наборах даних візитів користувачів до онлайн-магазинів роздрібної торгівлі, а їхня ефективність виміряна відносно введеної критеріальної бази.

У розділі 4 шляхом проведення низки експериментів над набором даних REES46 OpenCDP показано, що в порівнянні з бейзлайн-рішенням, що використовує моделі глибинних мереж на основі моделей LSTM з метою прогнозування послідовності взаємодії користувача з елементами, отриманими як результат застосування представленої методології моделями було досягнуто значного приросту метрик ранжування та скорочення часу видачі моделлю топ-100 найкращих пропозицій. Так, дистильована гібридна рекомендаційна система

на основі модифікованого механізму уваги демонструє приріст метрики $\text{MAP}@1$ на 169%, $\text{MAP}@10$ – на 117%, $\text{NDCG}@1$ – на 167%, а $\text{NDCG}@10$ – на 105%, при цьому час видачі топ-100 рекомендацій $\text{ATI}@100$ покращено на 10%.

У свою чергу побудована здобувачем гібридна рекомендаційна система на основі HiPPO-операторів демонструє покращення відносно метрики $\text{MAP}@1$ на 169%, $\text{MAP}@10$ – на 341%, $\text{NDCG}@1$ – на 217%, $\text{NDCG}@1$ – на 301%, $\text{NDCG}@10$ – 192%, при цьому час видачі топ-100 рекомендацій $\text{ATI}@100$ гірше на 14%.

Шляхом проведення експериментів над популярним набором даних RetailRocket показано, що синтезовані у результаті дослідження гібридна рекомендаційна система на основі модифікації моделей механізму уваги та гібридна рекомендаційна система на основі HiPPO-операторів мають перевагу в порівнянні з моделями, що є найпопулярнішими рішеннями в індустрії – GRU4Rec, SASRec та BERT4Rec.

Зокрема, гібридна рекомендаційна система на основі модифікованого механізму уваги продемонструвала сталий приріст значень метрики $\text{MAP}@20$ – на 70% відносно GRU4Rec, і 40% відносно SASRec, та метрики $\text{NDCG}@10$ – на 72% відносно GRU4Rec, 13% – SASRec та 38% – BERT4Rec. У свою чергу, метод на основі HiPPO-операторів надає покращення $\text{MAP}@20$ – на 480% в порівнянні з GRU4Rec та 406% відносно SAS4Rec, а відносно метрики $\text{NDCG}@10$ – на 90% відносно GRU4Rec, 20% відносно SAS4Rec та 47% – відносно BERT4Rec.

Додатково проведений візуальний аналіз доводить, що як мережі на основі модифікації моделей механізму уваги, так і мережі на основі застосування операторів поліноміальної проекції високого порядку (HiPPO-операторів) демонструють генерацію вкладень елементів, що зберігають структуру синтезованих графів знань, а у випадку з мережами на основі HiPPO-операторів – створюють її в процесі навчання мережі та оновлення відомостей щодо взаємодій користувачів з елементами.

Таким чином, поставлені в роботі задачі для досягнення мети дослідження виконано в повному обсязі.

Практична цінність дисертаційної роботи полягає в тому, що:

1) розроблена методологія побудови великомасштабних рекомендаційних систем, що здатні працювати з гетерогенними даними та вирішувати задачу видачі найкращої наступної пропозиції;

2) у межах методології вперше запропоновані швидкодійні гібридні рекомендаційні системи на основі дистиляції знань, а також моделей простору станів, що виконують неявне моделювання графу знань на основі даних візитів користувачів, що демонструють значний приріст у точності прогнозування наступної найкращої дії користувача, порівняно з іншими дослідженнями.

Результати роботи впроваджено у навчальний процес навчально-наукового інституту прикладного системного аналізу КПІ ім. Ігоря Сікорського.

Дисертаційна робота виконана у відповідності до планів науково-дослідних робіт:

- НДР «Розробка моделей та методів розв’язання задач передбачення на основі великих обсягів слабкоструктурованої інформації в умовах невизначеності», №2305-ф, Державний реєстраційний номер: 0122U000671, дата реєстрації: 26-03-2025;

- НДР «Розв’язання задач передбачення з використанням методів і моделей штучного інтелекту», №2314-ф, Державний реєстраційний номер: 0125U000244, дата реєстрації: 13-01-2025.

Всі теоретичні і практичні результати дисертаційної роботи у повній мірі опубліковано у закордонних періодичних наукових виданнях і вітчизняних фахових періодичних наукових виданнях, що входять до відповідного встановленого переліку, а також виконано їх належну апробацію на міжнародних наукових конференціях і семінарах.

За матеріалами дисертації опубліковано 8 робіт, з яких 5 – це статті у вітчизняних фахових періодичних наукових виданнях, що входять до переліку фахових видань затверджених МОН України за спеціальністю дисертації (2 з них включені до міжнародної наукометричної бази Scopus, 2 – до Web of Science та 1 – до фахового видання категорії «Б»), 1 стаття конференції, що увійшла до

іноземного періодичного видання, що індексується в наукометричній базі Scopus, та 2 – публікації у матеріалах конференцій (у тому числі в 1 міжнародній).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Nedashkovskaya, N. I., & Androsov, D. V. (2022). Generative time series model based on encoder–decoder architecture. *System Research and Information Technologies, 1*, 97–109.
2. Androsov, D. V. (2023). Neural ordinary differential equations for time series reconstruction. *Radio Electronics, Computer Science, Control, 4*, 69–75.
3. Androsov, D. V., & Nedashkovskaya, N. I. (2025). The hybrid sequential recommender system synthesis method based on attention mechanism with automatic knowledge graph construction. *System Research and Information Technologies, 1*, 7–18.
4. Androsov, D. (2024, November 20–22). Fast and scalable recommendation retrieval model with mixed attention and knowledge distillation. In *IDEAL 2024: 25th International Conference on Intelligent Data Engineering and Automated Learning*, Valencia, Spain.
5. Androsov, D. V., & Nedashkovskaya, N. I. (2025). Simple, fast and scalable recommendation systems via external knowledge distillation. *Radio Electronics, Computer Science, Control, 3*, 126–138.
6. Androsov, D. V., & Nedashkovskaya, N. I. (2025). System approach to multicriteria evaluation of session-based and sequential recommendation systems. *KPI Science News, 141*(4), 46–54 <https://doi.org/10.20535/kpissn.2025.4.343329>
7. Андросов, Д. В., & Недашківська, Н. І. (2022, листопад 22–29). Порівняльний аналіз моделей генерації рекомендацій на основі методів машинного навчання на графах. У *Системні науки та інформатика: збірник доповідей I Всеукраїнської науково-практичної конференції «Системні науки та інформатика»* (с. 9–13). НН ІПСА КІІ ім. Ігоря Сікорського.
8. Androsov, D. (2023, January 24–25). Recommendation retrieval on dynamic graphs. In *Proceedings of the 2nd International Conference on Emerging Technology Trends on the Smart Industry and the Internet of Things (TTSIT-2023)*.

9. Ko, H., Lee, S., Park, Y., & Choi, A. (2022). A survey of recommendation systems: Recommendation models, techniques, and application fields. *Electronics*, 11(1), 141. <https://doi.org/10.3390/electronics11010141>
10. Raza, S., Rahman, M., Kamawal, S., Toroghi, A., Raval, A., Navah, F., & Kazemeini, A. (2024). A comprehensive review of recommender systems: Transitioning from theory to practice. *arXiv*. <https://doi.org/10.48550/arXiv.2407.13699>
11. Xia, Z., Sun, A., Xu, J., Peng, Y., Ma, R., & Cheng, M. (2024). Contemporary recommendation systems on big data and their applications: A survey. *IEEE Access*, 12, 182857–182884. <https://doi.org/10.1109/ACCESS.2024.3517492>
12. Gomez-Urbe, C. A., & Hunt, N. (2015). The Netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems*, 6(4), Article 13, 1–19. <https://doi.org/10.1145/2843948>
13. Jacobson, K., Murali, V., Newett, E., Whitman, B., & Yon, R. (2016). Music personalization at Spotify. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys)*. 373–373. <https://doi.org/10.1145/2959100.2959120>
14. Covington, P., Adams, J., & Sargin, E. (2016). Deep neural networks for YouTube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys)*. 191–198. <https://doi.org/10.1145/2959100.2959190>
15. Zhou, R. (2024). Understanding the impact of TikTok's recommendation algorithm on user engagement. *International Journal of Computer Science and Information Technology*, 3(2), 194–199. <https://doi.org/10.62051/ijcsit.v3n2.24>
16. Chen, Y., Liu, Z., Li, J., McAuley, J., & Xiong, C. (2022). Intent contrastive learning for sequential recommendation. In *Proceedings of the ACM Web Conference 2022 (WWW)*. 2172–2182. <https://doi.org/10.1145/3485447.3512090>
17. Wang, X., Tang, K., Wu, H., Wang, M., Shen, Y., Hou, G., Tan, Z., Li, P., Zhuang, Y., & Lu, W. (2024). Agent-Pro: Learning to evolve via policy-level reflection and optimization. *arXiv*. <https://arxiv.org/abs/2402.17574>

18. Kannout, A., Elshenawy, L. M., & Mostafa, M. (2022). Over-speeding trend across self-reported driving aberrant behaviors: A simulator study. *Frontiers in Psychology, 13*, 966656. <https://doi.org/10.3389/fpsyg.2022.966656>
19. Uta, M., Aly, R., Hasibi, F., Balog, K., & de Rijke, M. (2024). Knowledge distillation for cross-domain recommendation. *ACM Transactions on Recommender Systems, 2*(2), Article 12. <https://doi.org/10.1145/3617823apastyle.apa>
20. Felfernig, A., Friedrich, G., Jannach, D., & Zanker, M. (2006). An integrated environment for the development of knowledge-based recommender applications. *International Journal of Electronic Commerce, 11*(2), 133–151.paperpile+1
21. Reiterer, S., Felfernig, A., & Jeran, M. (2014). An integrated knowledge engineering environment for constraint-based recommender systems. In *Proceedings of the 7th Workshop on Knowledge Engineering and Software Engineering*. 1–8.
22. Jannach, D., Zanker, M., & Fuchs, M. (2009). Constraint-based recommendation in tourism: A multiperspective case study. *Information Technology & Tourism, 11*(2), 139–155.
23. Bistarelli, S., Montanari, U., & Rossi, F. (1997). Semiring-based constraint satisfaction and optimization. *Journal of the ACM, 44*(2), 201–236.
24. Meseguer, P., Rossi, F., & Schiex, T. (2006). Soft constraints. In F. Rossi, P. van Beek, & T. Walsh (Eds.), *Handbook of constraint programming*. 281–328. Elsevier.
25. Ruttkay, Z. (1994). Fuzzy constraint satisfaction. In *Proceedings of the 1st IEEE Conference on Evolutionary Computation*. 542–547.
26. Gatzourou, A., & Sánchez-Marré, M. (2015). A case-based recommendation approach for market basket data. *IEEE Intelligent Systems, 30*(1), 20–27. <https://doi.org/10.1109/MIS.2013.140>
27. Shi, Y., Karatzoglou, A., Baltrunas, L., Larson, M., & Hanjalic, A. (2014). CARS2: Learning context-aware representations for context-aware recommendations. In *Proceedings of the 23rd ACM International Conference on Information and*

Knowledge Management (CIKM), 291–300.
<https://doi.org/10.1145/2661829.2662070>.

28. Lashinin, O., Krasilnikov, D., Milogradskii, A., & Ananyeva, M. (2024). SAFERec: Self-attention and frequency enriched model for next basket recommendation. *arXiv*. <https://doi.org/10.48550/arXiv.2412.14302>.

29. Singhal, R., Shroff, G., Kumar, M., Choudhury, S. R., Kadarkar, S., Virk, R., Verma, S., & Tewari, V. (2019). Fast online “next best offers” using deep learning. In *Proceedings of the 10th ACM Conference on Data Science (CoDS-COMAD)*, 208–211. <https://doi.org/10.1145/3297001.3297029>.

30. TensorFlow Recommenders. (2020). *TensorFlow recommenders (TFRS) documentation*. <https://www.tensorflow.org/recommenders>.

31. PyTorch. (2024). *Introduction to TorchRec*. https://docs.pytorch.org/tutorials/intermediate/torchrec_intro_tutorial.html.

32. NVIDIA. (2021). *NVIDIA Merlin: Open source library for recommender systems*. <https://developer.nvidia.com/merlin>.

33. Mackey, P., & Rouhani, B. D. (2020). TensorFlow Recommenders: An open-source framework for building recommender system models. *Google Research Blog*. <https://blog.tensorflow.org/2020/09/introducing-tensorflow-recommenders.html>.

34. Ivchenko, D., Van Der Staay, D., Taylor, C., Liu, X., Feng, W., Kindi, R., Sudarshan, A., & Sefati, S. (2022). TorchRec: A PyTorch domain library for recommendation systems. In *Proceedings of the 16th ACM Conference on Recommender Systems (RecSys '22)*, 482–483. <https://doi.org/10.1145/3523227.3547387>.

35. PyTorch. (2024). *TorchRec and FBGEMM 1.0 stable release*. <https://pytorch.org/blog/torchrec-fbgemm-1/>.

36. Hong, S., Choi, J., Lee, Y.-C., Kumar, S., & Park, N. (2024). SVD-AE: Simple autoencoders for collaborative filtering. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI)*, 2065–2073. <https://doi.org/10.48550/arXiv.2405.04746>.

37. Sugahara, K., & Okamoto, K. (2024). Hierarchical matrix factorization for interpretable collaborative filtering. *arXiv*. <https://doi.org/10.48550/arXiv.2311.13277>.
38. Xu, D., Ruan, C., Korpeoglu, E., Kumar, S., & Achan, K. (2021). Rethinking neural vs. matrix-factorization collaborative filtering: The theoretical perspectives. *arXiv*. <https://doi.org/10.48550/arXiv.2110.12141>.
39. Kurovski, M., & Wilhelm, F. (2019). On the effectiveness of low-rank approximations for collaborative filtering compared to neural networks. *arXiv*. <https://doi.org/10.48550/arXiv.1905.12967>.
40. Bobadilla, J., Dueñas-Lerín, J., Ortega, F., & Gutierrez, A. (2024). Comprehensive evaluation of matrix factorization models for collaborative filtering recommender systems. *International Journal of Interactive Multimedia and Artificial Intelligence*, 8(6), 15–23. <https://doi.org/10.9781/ijimai.2023.04.008>.
41. Park, S., Yoon, M., Lee, J., Park, H., & Lee, J. (2023). Toward a better understanding of loss functions for collaborative filtering. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM)*. <https://doi.org/10.48550/arXiv.2308.06091>.
42. Winlaw, M., Hynes, M. B., Caterini, A., & De Sterck, H. (2015). Algorithmic acceleration of parallel ALS for collaborative filtering: Speeding up distributed big data recommendation in Spark. *arXiv*. <https://doi.org/10.48550/arXiv.1508.03110>.
43. He, X., Zhang, H., Kan, M.-Y., & Chua, T.-S. (2017). Fast matrix factorization for online recommendation with implicit feedback. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 549–558. <https://doi.org/10.1145/3077136.3080662>.
44. Rendle, S., Krichene, W., Zhang, L., & Koren, Y. (2021). Revisiting the performance of iALS on item recommendation benchmarks. *arXiv*. <https://doi.org/10.48550/arXiv.2110.14037>.
45. Ammad-ud-din, M., Ivannikova, E., Khan, S. A., Oyomno, W., Fu, Q., Tan, K. E., & Flanagan, A. (2019). Federated collaborative filtering for privacy-

preserving personalized recommendation system. *arXiv*.
<https://doi.org/10.48550/arXiv.1901.09888>

46. Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2010). BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, 452–461.

47. Milogradskii, A., Lashinin, O., Petrov, A., Ananyeva, M., & Kolesnikov, S. (2024). Revisiting BPR: A replicability study of a common recommender system baseline. In *Proceedings of the 18th ACM Conference on Recommender Systems (RecSys)*, 877–882. <https://doi.org/10.1145/3640457.3688073>.

48. Shi, K., Zhang, J., Fang, L., Wang, W., & Jing, B. (2024). Enhanced Bayesian personalized ranking for robust hard negative sampling in recommender systems. *arXiv*. <https://doi.org/10.48550/arXiv.2403.19276>.

49. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T.-S. (2017). Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web (WWW)*, 173–182. <https://doi.org/10.1145/3038912.3052569>.

50. Dacrema, M. F., Cremonesi, P., & Jannach, D. (2020). Neural collaborative filtering vs. matrix factorization revisited. In *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys)*. <https://doi.org/10.1145/3383313.3412488>.

51. Hu, Z.-Y., Huang, J., Deng, Z.-H., Wang, C.-D., Huang, L., Lai, J.-H., & Yu, P. S. (2021). BCFNet: A balanced collaborative filtering network with attention mechanism. *arXiv*. <https://doi.org/10.48550/arXiv.2103.06105>.

52. Rafailidis, D., & Weiss, G. (2019). A neural attention model for adaptive learning of social friends' preferences. *arXiv*. <https://doi.org/10.48550/arXiv.1907.01644>.

53. Fang, J. (2021). Session-based recommendation with self-attention networks. *arXiv*. <https://doi.org/10.48550/arXiv.2102.01922>.

54. Liang, D., Krishnan, R. G., Hoffman, M. D., & Jebara, T. (2018). Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 World Wide Web Conference (WWW)*, 689–698. <https://doi.org/10.1145/3178876.3186150>.

55. Shenbin, I., Alekseev, A., Tutubalina, E., Malykh, V., & Nikolenko, S. I. (2020). RecVAE: A new variational autoencoder for top-N recommendations with implicit feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM)*, 528–536. <https://doi.org/10.1145/3336191.3371831>.
56. Liu, Z., Xu, Z., Jin, J., Shen, Z., & Darrell, T. (2023). Dropout reduces underfitting. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 22233–22248. <https://doi.org/10.48550/arXiv.2303.01500>.
57. Kuchaiev, O., & Ginsburg, B. (2017). Training deep autoencoders for collaborative filtering. *arXiv*. <https://doi.org/10.48550/arXiv.1708.01715>
58. Liu, Z., Wang, Y., Guan, W., Li, S., & Liang, Q. (2021). Contrastive learning for recommender system. *arXiv*. <https://doi.org/10.48550/arXiv.2101.01317>
59. Chang, B., Xu, C., Lê, M., Feng, J., Le, Y., Badam, S., Chi, E., & Chen, M. (2022). Recency dropout for recurrent recommender systems. *arXiv*. <https://doi.org/10.48550/arXiv.2201.11016>.
60. Yang, Z., Ding, M., Huang, T., Cen, Y., Song, J., Xu, B., Dong, Y., & Tang, J. (2024). Does negative sampling matter? A review with insights into its theory and applications. *arXiv*. <https://doi.org/10.48550/arXiv.2402.17238>.
61. Chen, H., Li, X., Lai, V., Yeh, C.-C. M., Fan, Y., Wang, Y., Zheng, V. W., & Chen, Y. (2023). Adversarial collaborative filtering for free. *arXiv*. <https://doi.org/10.48550/arXiv.2308.13541>.
62. Fang, H., Zhang, D., Shu, Y., & Guo, G. (2020). Deep learning for sequential recommendation: Algorithms, influential factors, and evaluations. *ACM Transactions on Information Systems*, 39(1), Article 11. <https://doi.org/10.1145/3426723>.
63. Kang, W.-C., & McAuley, J. (2018). Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*, 197–206. <https://doi.org/10.1109/ICDM.2018.00035>.
64. Zimdars, A., Chickering, D. M., & Meek, C. (2001). Using temporal data for making recommendations. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence (UAI)*, 580–588.

65. Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2016). Session-based recommendations with recurrent neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1511.06939>.
66. Li, J., Ren, P., Chen, Z., Ren, Z., Lian, T., & Ma, J. (2017). Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM Conference on Information and Knowledge Management (CIKM)*, 1419–1428. <https://doi.org/10.1145/3132847.3132926>.
67. Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., & Jiang, P. (2019). BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, 1441–1450. <https://doi.org/10.1145/3357384.3357895>.
68. Petrov, A. V. (2025). *Effective and efficient transformer models for sequential recommendation* (Doctoral dissertation, University of Glasgow).
69. Gao, C., Zheng, Y., Li, N., Li, Y., Qin, Y., Piao, J., Quan, Y., Chang, J., Jin, D., He, X., & Li, Y. (2023). Graph neural networks for recommender systems: Challenges, methods, and directions. *ACM Transactions on Recommender Systems*, 1(1), Article 3. <https://doi.org/10.1145/3568022>.
70. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems (NIPS)*, 26, 2787–2795).
71. Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). DeepWalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 701–710. <https://doi.org/10.1145/2623330.2623732>.
72. Grover, A., & Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 855–864. <https://doi.org/10.1145/2939672.2939754>.

73. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020). LightGCN: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 639–648. <https://doi.org/10.1145/3397271.3401063>.
74. Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv*. <https://arxiv.org/abs/1503.02531>.
75. Lopes, R. G., Fenu, S., & Starner, T. (2017). Data-free knowledge distillation for deep neural networks. *arXiv*. <https://doi.org/10.48550/arXiv.1710.07535>.
76. Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv*. <https://doi.org/10.48550/arXiv.1910.01108>.
77. Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., & Bengio, Y. (2015). FitNets: Hints for thin deep nets. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1412.6550>.
78. Anil, R., Pereyra, G., Passos, A., Ormandi, R., Dahl, G. E., & Hinton, G. E. (2018). Large scale distributed neural network training through online distillation. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*. <https://doi.org/10.48550/arXiv.1804.03235>.
79. Furlanello, T., Lipton, Z. C., Tschannen, M., Itti, L., & Anandkumar, A. (2018). Born again neural networks. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*. 80, 1607–1616. <https://doi.org/10.48550/arXiv.1805.04770>.
80. Chen, R. T. Q., Rubanova, Y., Bettencourt, J., & Duvenaud, D. (2018). Neural ordinary differential equations. In *Advances in Neural Information Processing Systems 31 (NIPS'18)*. Curran Associates Inc., Red Hook, NY, USA, 6572–6583.
81. Dai, S., Shao, N., Li, H., Chen, W., & Zhang, Y. (2024). ReCODE: Modeling repeat consumption with neural ODE. In *Proceedings of the 47th International Conference on Research and Development in Information Retrieval*

(SIGIR '24). Association for Computing Machinery, New York, NY, USA, 2599–2603. <https://doi.org/10.1145/3626772.3657936>.

82. Golovanev, Y., & Hvatov, A. (2022). On the balance between the training time and interpretability of neural ODE for time series modelling. arXiv. <https://arxiv.org/abs/2206.03304>.

83. Gu, A., Dao, T., Ermon, S., Rudra, A., & Ré, C. (2020). HiPPO: Recurrent memory with optimal polynomial projections. In *Advances in Neural Information Processing Systems (NIPS '20)*. Curran Associates Inc., Red Hook, NY, USA, Article 125, 1474–1487.

84. Gu, A., & Dao, T. (2023). Mamba: Linear-time sequence modeling with selective state spaces. arXiv. <https://arxiv.org/abs/2312.00752>.

85. Gu, A., Goel, K., & Ré, C. (2022). Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/2111.00396>.

86. Xiao, W., Wang, H., Zhou, Q., & Wang, Q. (2025). SS4Rec: Continuous-time sequential recommendation with state space models. arXiv. <https://doi.org/10.48550/arXiv.2502.08132>.

87. Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511809071>.

88. Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4), 422–446. <https://doi.org/10.1145/582415.582418>.

89. Baeza-Yates, R., & Ribeiro-Neto, B. (1999). *Modern information retrieval*. ACM Press; Addison-Wesley.

ДОДАТОК А СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Публікації, в яких опубліковані основні наукові результати дисертації:

1. Nedashkovskaya, N. I., & Androsov, D. V. (2022). Generative time series model based on encoder-decoder architecture. *System Research and Information Technologies*, 1, 97-109. (внесок: Недашківська Н.І. – наукове керівництво та формулювання концепції; Андросов Д.В. – розробка архітектури автокодувальника, проведення експериментів, аналіз результатів)
2. Androsov, D. V. (2023). Neural ordinary differential equations for time series reconstruction. *Radio Electronics, Computer Science, Control*, 4, 69-75. (внесок: Недашківська Н.І. – наукове керівництво та формулювання концепції; Андросов Д.В. – розробка архітектури мережі на основі нейронних ЗДР, проведення експериментів, аналіз результатів)
3. Androsov, D. V., & Nedashkovskaya, N. I. (2025). The hybrid sequential recommender system synthesis method based on attention mechanism with automatic knowledge graph construction. *System Research and Information Technologies*, 1, 7-18. (внесок: Недашківська Н.І. – наукове керівництво та редагування; Андросов Д.В. – розробка методу, проведення експериментів, аналіз результатів)
4. Androsov, D. V., & Nedashkovskaya, N. I. (2025). Simple, fast and scalable recommendation systems via external knowledge distillation. *Radio Electronics, Computer Science, Control*, 3, 126-137. (внесок: Недашківська Н.І. – наукове керівництво та редагування; Андросов Д.В. – розробка методу, проведення експериментів, аналіз результатів)
5. Androsov, D. V., & Nedashkovskaya, N. I. (2025). System approach to multicriteria evaluation of session-based and sequential recommendation systems. *KPI Science News*, 4, 7-15. (внесок: Недашківська Н.І. – наукове керівництво та редагування; Андросов Д.В. – аналіз літератури, порівняння ефективності рекомендаційних систем, аналіз результатів точності рекомендаційних систем)

6. Androsov, D. (2025). Fast and scalable recommendation retrieval model with mixed attention and knowledge distillation. In V. Julian et al. (Eds.), *Intelligent Data Engineering and Automated Learning – IDEAL 2024* (Lecture Notes in Computer Science, Vol. 15346, pp. 244–253). Springer. (внесок: Андросов Д.В. – аналіз літератури, створення методу побудови гібридної рекомендаційної системи, проведення експериментів, аналіз результатів)

Публікації, які засвідчують апробацію матеріалів дисертації:

7. Андросов, Д. В., & Недашківська, Н. І. (2022, листопад 22–29). Порівняльний аналіз моделей генерації рекомендацій на основі методів машинного навчання на графах. У *Системні науки та інформатика: збірник доповідей I Всеукраїнської науково-практичної конференції «Системні науки та інформатика»* (с. 9–13). НН ІПСА КПП ім. Ігоря Сікорського [Тези конференції]. (внесок: Андросов Д.В. – аналіз існуючих архітектур графових нейронних мереж в контексті рекомендаційних систем, проведення експериментів, аналіз результатів; Недашківська Н.І. – наукова редакція)

8. Androsov, D. (2023, January 24–25). Recommendation retrieval on dynamic graphs. In *Proceedings of the 2nd International Conference on Emerging Technology Trends on the Smart Industry and the Internet of Things (TTSIIT-2023)*. [Conference paper]. (внесок: Андросов Д.В. – набуття подальшого розвитку методу дистиляції знань в контексті побудови гібридних рекомендаційних систем, проведення експериментів, аналіз результатів)

ДОДАТОК Б АКТ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЙНОГО ДОСЛІДЖЕННЯ



УКРАЇНА

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
(КПІ ім. Ігоря Сікорського)
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ

пр-т Берестейський, 37, м. Київ, 03056, тел. (044) 204 82 82 тел. (044) 204 94 94
<https://www.kpi.ua> e-mail: mail@kpi.ua ЄДРПОУ 02070921

25.02.2026 № 03/09.00
на № _____ від _____

АКТ

про впровадження результатів дисертаційної роботи Андросова Дмитра Васильовича
на тему «Рекомендаційна система на основі нейронних мереж та динамічних вкладень
графів»

Цим актом підтверджується, що в процесі виконання дисертаційного дослідження Андросов Д.В. отримав низку наукових результатів стосовно розробки рекомендаційних систем, які ґрунтуються на моделях і методах глибокого навчання та використовують графові дані. Результати дисертаційного дослідження впроваджено у навчальний процес кафедри математичних методів системного аналізу навчально-наукового інституту прикладного системного аналізу КПІ ім. Ігоря Сікорського в межах навчальної дисципліни «Інтелектуальний аналіз великих сховищ даних (Big data analytics)» другого (магістерського) рівня вищої освіти за спеціальністю F4 «Системний аналіз та наука про дані». Автором дисертаційної роботи було розширено робочу програму цієї дисципліни, зокрема, додано розділ «Рекомендаційні системи», зміст якого включає методи і моделі колаборативної і контентної фільтрації, матричної факторизації та гібридні алгоритми на основі нейронних мереж. Додатково в рамках освітнього компоненту «Дизайн систем машинного навчання» другого (магістерського) рівня вищої освіти за спеціальністю F4 «Системний аналіз та наука про дані» студенти вивчають процес побудови, навчання, практичного використання рекомендаційних систем, зокрема розгортання в хмарні середовища, а також нові методи і модифікації запропоновані Андросовим Д.В. в рамках дисертаційного дослідження.

В. о. директора Навчально-наукового
інституту прикладного системного аналізу

Завідувач кафедри
математичних методів системного аналізу



К. В. Єфремов

О. Л. Тимошук