

[0000-0003-4816-4680] **В. В. Міхав**¹, аспірант,

e-mail: mihaw.wolodymyr@gmail.com

[0000-0001-8791-0063] **Є. В. Мелешко**¹, д-р техн. наук, професор,

[0000-0001-6951-2002] **О. М. Дресв**¹, канд. техн. наук, доцент,

[0000-0002-1596-4123] **А. О. Лавданський**², канд. техн. наук, доцент

¹Центральноукраїнський національний технічний університет
просп. Університетський, 8, м. Кропивницький, 25006, Україна

²Черкаський державний технологічний університет
б-р Шевченка, 460, м. Черкаси, 18006, Україна

МОДЕЛЬ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ ДЛЯ КОМП'ЮТЕРНИХ МЕРЕЖ ТИПУ P2P

У цій статті проведено дослідження алгоритмів роботи peer to peer мереж. Проведено порівняльний аналіз методів пошуку даних у централізованих, децентралізованих неструктурованих та децентралізованих структурованих однорангових мережах. Виявлено, що існуючі методи роботи однорангових мереж не застосовують для покращення пошуку даних рекомендаційні системи. Розроблено математичну модель рекомендаційної системи для децентралізованої P2P мережі з огляду на вподобання користувачів та кількість переходів до завантаження даних. Розроблено метод формування рекомендацій для користувачів комп'ютерної мережі типу peer to peer на основі запропонованої математичної моделі.

У P2P мережах виникає проблема індексації та пошуку файлів на різних пристроях мережі. З різних причин шукані файли можуть бути недоступні для користувача, навіть якщо вони були додані раніше до системи та проіндексовані. Наприклад, комп'ютери, що містять потрібний файл або таблиці маршрутизації до нього чи його частин, вийшли з мережі або застосовуються технології побудови P2P мережі з ймовірнісними методами пошуку, що не завжди знаходять далеко розташовані від комп'ютера користувача файли, тощо.

Рекомендаційні системи використовуються для побудови списків рекомендацій користувачам на основі їх попередніх дій, зокрема, лайків, оцінок, переглядів, скачувань тощо. Такі системи дозволяють полегшити пошук при великій кількості об'єктів у системі, доповнюючи класичну пошукову видачу рекомендаціями, а в деяких ситуаціях навіть заміняють пошук. Також рекомендаційні системи можуть застосовуватися для ранжування результатів класичного пошуку. Таким чином, вони можуть різними способами поєднуватися зі звичайними пошуковими алгоритмами. В P2P мережах застосування рекомендаційних систем може мати додаткову користь. Якщо користувач шукає конкретний файл, що був доданий до мережі раніше, і файл з різних причин не знайдено, можна надати користувачу список рекомендацій з огляду на його вподобання і, можливо, пошуковий запит.

Запропоновані у роботі модель та метод формування списків рекомендацій у peer to peer мережі розраховані на загальний випадок і не прив'язані до конкретного пошукового запиту, можуть застосовуватися у неструктурованих та структурованих P2P мережах для ознайомлення користувача з контентом, який йому може сподобатися, на основі прогнозування його вподобань. Це може збільшити загальний інтерес користувачів до контенту мережі.

Ключові слова: комп'ютерні мережі, однорангові мережі, peer to peer, рекомендаційні системи, пошук даних.

Вступ. У цій роботі запропоновано математичну модель рекомендаційної системи для однорангової комп'ютерної мережі типу peer to peer.

P2P – це однорангові комп'ютерні мережі, що базуються на принципі рівноправності вузлів мережі і характеризуються тим, що

будь-які вузли можуть з'єднуватися між собою для надання послуг один одному, на відміну від традиційної архітектури, в якій лише окрема категорія учасників (сервери) може надавати певні сервіси іншим [1]. Відомими прикладами використання P2P мереж є торенти для обміну файлами та децентралізовані

блокчейн-платформи, включаючи Bitcoin, також їх можна використовувати для управління ресурсами в мережі IoT, паралельного програмування та машинного навчання, кешування даних та резервного копіювання [1-5]. У рамках цієї роботи розглядалося застосування P2P мереж для розподіленого зберігання та обміну даними.

У P2P мережах виникає проблема індексації та пошуку файлів на різних пристроях мережі [6-7], а алгоритмів рекомендаційних систем, орієнтованих на таку архітектуру, не було знайдено під час проведеного дослідження. Найближчий до рекомендаційних систем знайдений алгоритм – децентралізований інтелектуальний пошук, що використовує коефіцієнти подоби [6], але він призначений саме для пошуку.

Рекомендаційні системи будують списки рекомендацій користувачам на основі їх попередніх дій (лайків, оцінок, переглядів, скачувань тощо). Такі системи дозволяють полегшити пошук при великій кількості об'єктів, доповнюючи класичну пошукову видачу рекомендаціями, а в деяких ситуаціях навіть замінюють пошук, а також можуть застосовуватися для ранжування результатів пошуку. Таким чином, рекомендаційні системи можуть різними способами поєднуватися зі звичайними пошуковими алгоритмами.

В P2P мережах застосування рекомендаційних систем може надати додаткову користь. Оскільки в P2P мережах немає серверів, на яких зберігаються усі дані, а файли розподілені по всіх пристроях системи, то не завжди всі файли доступні. Отже, при пошуку файлів слід враховувати не тільки релевантність запиту, а й доступність файлів та швидкість їх завантаження. Рекомендаційні системи можуть запропонувати користувачу файли, що зможуть також зацікавити його і є доступними для швидкого завантаження у певний момент часу, якщо потрібний файл на цей момент недоступний. Також поєднання пошукової та рекомендаційної систем дасть змогу збалансувати результати пошукової видачі за кількома параметрами, а саме, релевантністю, доступністю, швидкістю завантаження, схожістю, рекомендованістю. Було вирішено побудувати математичну модель такої системи для подальшої розробки методів та алгоритмів на її основі, а також програмної моделі для тестування алгоритмів.

Мета та задачі дослідження. Метою цієї роботи є розробка математичної моделі рекомендаційної системи для однорангової комп'ютерної мережі типу peer to peer, яка має полегшити пошук даних у децентралізованих мережах.

Для досягнення поставленої мети необхідно вирішити такі завдання:

– Провести дослідження існуючих методів роботи peer to peer мереж.

– Розробити математичну модель рекомендаційної системи для децентралізованої комп'ютерної peer to peer мережі.

– Розробити метод формування рекомендацій для користувачів peer to peer мережі на основі запропонованої математичної моделі.

Виклад основного матеріалу. Спочатку розглянемо відомі алгоритми індексації та пошуку даних у P2P мережах.

Технології P2P мереж можна розділити на такі три види:

– Централізована однорангова мережа.

– Неструктурована децентралізована однорангова мережа.

– Структурована децентралізована однорангова мережа.

Централізована однорангова мережа.

Інформація про індекс файлів зберігається на центральному сервері, кожний дочірній вузол повинен підключатися до нього, щоб знайти ресурси. Найбільшою перевагою такого підходу є просте обслуговування і швидке індексування. Однак, оскільки вся мережа значною мірою залежить від центрального сервера, якщо він вийде з ладу – вся система стане непрацездатною. Тож виникає проблема «єдиної точки відмови». Прикладом такої мережі є файлообмінний клієнт BitTorrent [7, 8]. Для роботи з ним потрібен «трекер» (tracker) – спеціалізований сервер, що працює за протоколом HTTP [8]. Трекер потрібен для того, щоб клієнти могли знайти один одного. Фактично на трекері зберігаються IP-адреси, вхідні порти клієнтів та хеш-суми, що унікально ідентифікують об'єкти, які беруть участь у завантаженнях. За стандартом імена файлів на трекері не зберігаються, і дізнатися їх по хеш-сумах не можна. Однак на практиці трекер часто, крім своєї основної функції, виконує і функцію невеликого веб-сервера. Такий сервер зберігає файли метаданих і описи поширюваних файлів, надає статистику завантажень по різних файлах, показує поточну кіль-

кість підключених «пірів» (учасників файлообміну) тощо. Для підвищення працездатності системи у BitTorrent допускається існування декількох трекерів.

Неструктурована децентралізована однорангова мережа. Цей тип мережі використовує алгоритм пошуку Flooding і транслює запит файлу, який потрібно знайти, усім вузлам мережі для будь-якого пошуку [1, 6]. Коли вузол хоче завантажити файл, він створює запит із назвою файлу або ключовим словом і надішле його усім вузлам, підключеним до нього. Якщо в цих вузлах є шуканий файл, встановлюється з'єднання з відповідним вузлом для завантаження. Якщо ні, запит продовжують пересилати на сусідні вузли, доки не буде знайдено розташування файлу. Типовий представник цього методу таких мереж – Gnutella рання версія. Слід зазначити, що, коли масштаб мережі стає дуже великим, цей метод пошуку почне викликати «шторм трансляції», який серйозно знизить пропускну здатність мережі. Хоча цей метод дозволяє уникнути проблеми «єдиної точки відмови» централізованих однорангових мереж, він все одно є дуже неефективним. Для усунення проблеми «шторму трансляції» було розроблено багато алгоритмів, але всі вони полягають у тому, що при пошуку файлу опитуються не всі вузли мережі, а тільки їх частина. Такі алгоритми знайдуть файл, який дійсно є в мережі, не завжди, але вони є певним компромісом між швидкістю роботи мережі й ефективністю пошуку даних. Найвідоміші з них [1, 6, 9, 10]:

– Метод широкого первинного пошуку (Breadth First Search, BFS) з додаванням до пошукового запиту параметра часу життя (Time-to-level, TTL). Параметр TTL визначає максимальну кількість переходів, якими можна пересилати запит. При типовому пошуку початкове значення для TTL становить зазвичай 5-7, яке зменшується щоразу, коли запит надсилається. Коли TTL дорівнює 0, повідомлення більше не передається.

– Метод випадкового широкого первинного пошуку (Random Breadth First Search, RBFS) було запропоновано як поліпшення «наївного» підходу BFS. У методі RBFS пошуковий запит пересилається лише частині вузлів мережі, вибраній у випадковому порядку. Якій саме частині вузлів – це параметр методу RBFS. Перевага RBFS полягає в тому, що не потрібна глобальна інформація про

стан контенту мережі; вузол може отримувати локальні рішення так швидко, як це потрібно. З другого боку, цей метод є ймовірнісним. Тому деякі великі сегменти мережі можуть бути недосяжними.

– Інтелектуальний пошуковий механізм (Intelligent Search Mechanism, ISM). Поліпшення швидкості й ефективності пошуку інформації за допомогою цього методу досягається за рахунок мінімізації витрат на зв'язки, тобто на кількість повідомлень, що передаються між вузлами, та мінімізації кількості вузлів, які опитуються для кожного пошукового запиту. Щоб досягти цього, для кожного запиту оцінюються лише ті вузли, які найбільше відповідають цьому запиту. Інтелектуальний пошуковий механізм складається з двох компонентів – профайлу (profile) та способу його ранжування, так званого рангу релевантності. Кожний вузол мережі будує інформаційний профайл для кожного із сусідніх вузлів. Профайл містить останні відповіді кожного з вузлів. За допомогою рангу релевантності здійснюється ранжування профайлів вузлів для вибору тих сусідніх, які надаватимуть найбільш релевантні документи на запит. Механізм профайлів використовується для того, щоб зберігати останні запити, а також кількісні характеристики результатів пошуку. При реалізації моделі ISM використовується єдиний стек запитів розміром $O(TN)$, в якому зберігається T запитів для N вузлів. Як тільки стек заповнюється, відбувається заміна «останнього найменш використовуваного» (Least Recently Used, LRU) для збереження останніх запитів. Функція «ранг релевантності» (Relevance Rank, RR) використовується вузлом P для виконання оперативної класифікації його сусідів та визначення тих, які слід опитувати першими за запитом q . Для обчислення рангу релевантності кожного вузла P P_i порівнює q з усіма запитами q_j у структурі профайлу, для якого відомий список відповідей на попередні запити, та обчислюється $RR(P_i, q)$:

$$RR(P_i, q) = \sum_{j \in Q} Sim(q_j, q)^a \cdot S(P_i, q_j), \quad (1)$$

де Q – множина запитів q_j , на які була відповідь у вузла P_i , $j \in Q$; $S(P_i, q_j)$ – кількість результатів, що повертаються вузлом P_i на запит q_j ; метрика Sim розраховується за правилом косинуса:

$$Sim(q_j, q) = \frac{q_j \cdot q}{|q_j| \cdot |q|} \quad (2)$$

і забезпечує вищий ранг вузла, який повертає більше результатів; параметр a – дозволяє збільшувати вагу запитів, найбільш подібних до поточного. Метод ISM ефективно працює у мережах, де вузли містять деякі спеціалізовані відомості. Зокрема у мережі Gnutella якість пошуку дуже залежить від «оточення» вузла, з якого задається запит. Ще одна проблема методу ISM полягає в тому, що пошукові повідомлення можуть зациклюватися, тому не в змозі досягти деяких частин мережі. Цю проблему можна вирішити, вибираючи невелику випадкову підмножину вузлів і додаючи до набору релевантних вузлів для кожного запиту, тоді механізм ISM став охоплювати більшу частину мережі.

– Методи «більшості результатів за мінусовою евристикою» (RES). В цих методах кожний вузол пересилає запит підмножині своїх вузлів, утвореній на підставі деякої узагальненої статистики. Запит у методі RES є задовільним, якщо видається Z або більше результатів (Z – деяка стала). У методі RES вузол q пересилає запити до k вузлів, які видали найбільші результати останніх m запитів. Метод RES подібний до методу ISM, але використовує більш просту інформацію про вузли. Його головний недолік порівняно з ISM – відсутність аналізу параметрів вузлів, зміст яких пов'язаний із запитом. Тому метод RES характеризується швидше як кількісний, а не якісний підхід. RES хороший тим, що він маршрутизує запити у великі сегменти мережі (які, можливо, також містять більш релевантні відповіді). Він також захоплює сусідів, які менш переважані, починаючи з тих, котрі зазвичай повертають більше результатів.

– Метод «випадкових блукань» (Random Walkers Algorithm, RWA) полягає в тому, що кожний вузол випадково пересилає повідомлення із запитом, що називається «посилкою», одному зі своїх вузлів. Щоб скоротити час, необхідний на отримання результатів, ідея однієї «посилки» розширена до « k -посилок», де k – кількість незалежних посилок, послідовно запущених з пошукового вузла. Очікується, що « k -посилок» після T кроків досягне тих самих результатів, що й одна посилка за kT кроків. Цей алгоритм нагадує метод RBFS, але в RBFS кожний вузол пересилає повідомлення запиту частині з його сусідів. До того ж, в RBFS передбачається експоненційне збільшення повідомлень, що пере-

силаються, а в методі випадкових блукань – лінійне. Обидва методи – і RBFS, і RWA – не використовують жодних явних правил для того, щоб адресувати пошуковий запит до найбільш релевантного змісту.

– Метод «адаптивного ймовірнісного пошуку» (Adaptive Probabilistic Search, APS). В APS кожний вузол розгортає на своїх ресурсах локальний індекс, що містить значення умовних ймовірностей для кожного сусіда, який може бути обраний наступного переходу майбутнього запиту. Головна відмінність від RWA – те, що у APS вузол використовує зворотний зв'язок від попередніх пошуків (як умовних ймовірностей) замість цілком випадкових переходів. Тому метод APS часто дає кращі результати, ніж RWA.

Структурована децентралізована однорангова мережа. Нині найбільшого поширення набула структурована децентралізована мережа [11-14], тобто мережа на основі DHT (distributed hash table, розподілена хеш-таблиця) [14]. DHT використовує більш структурований метод маршрутизації на основі пар ключ-значення. Для кожного вузла мережі обчислюється хеш-значення, воно стає його ідентифікатором. Для кожного файлу, що треба розмістити у мережі, також обчислюється хеш-значення такою самою хеш-функцією (воно стає ключем файлу), після чого файл записується на вузол, ідентифікатор якого максимально близький до ключа файлу за певною метрикою відстані. Кожний вузол мережі зберігає власну таблицю маршрутизації у мережі, де міститься інформація про вузли, ідентифікатори яких максимально близькі до нього за обраною метрикою. Прикладом практичної реалізації є мережа I2P, файлообмінний клієнт iMule, криптовалюта Ethereum [1-5]. Слід зазначити, що в реальних системах у тій чи іншій формі сервери у системі все ж таки наявні, і DHT часто відіграє лише допоміжну функцію, тобто маршрутизація в мережі побудована не тільки на ній. Зокрема, в Ethereum DHT використовується лише як ефективний механізм вибору партнерів. На цей момент існують такі алгоритми роботи P2P мереж, що реалізують протокол DHT: Kademlia [15] і Chord [16], і серед них найчастіше використовується алгоритм Kademlia. Розглянемо принцип роботи алгоритму Kademlia. Хеш значення найчастіше обчислюється алгоритмом SHA-1, метрикою відстані є операція XOR.

Таблиця маршрутизації зберігається у вигляді так званих K-buckets. У кожному K-bucket вузла зберігається інформація про K вузлів мережі, чия відстань до нього знаходиться в межах інтервалу $[2^i, 2^{i+1})$. Протокол Kademlia містить чотири типи повідомлень: 1) PING – необхідний для перевірки існування конкретного вузла у мережі; 2) STORE запит, що дає змогу розмістити інформацію на заданому вузлі; 3) FIND_VALUE запит, що дозволяє знайти значення за ключем; 4) FIND_NODE запит, що використовується для пошуку найближчих K вузлів до заданого ідентифікатора (поведінка подібна до FIND_VALUE, тільки ніколи не повертає значення, завжди вузли).

На цей час усі три види технологій побудови P2P мереж використовуються.

Як показує дослідження, у P2P мережах з різних причин може бути відсутній доступ до шуканих файлів. У такому разі користувачу доречно надати список рекомендацій на схожі файли. Однак моделей та методів роботи рекомендаційних систем для P2P мереж в існуючих працях знайдено не було. Розглянемо можливість побудови рекомендаційної системи для P2P мереж різного типу.

Найпростіше побудувати рекомендаційну систему у централізованих однорангових мережах, оскільки там наявний сервер, через який проходить інформація про завантаження файлів. Отже, на основі статистики завантаження файлів різними користувачами можна побудувати рекомендаційну систему на колаборативній фільтрації. Її точність можна підвищити, якщо надати можливість користувачам оцінювати файли та зберігати оцінки на сервері.

Більш складною задачею є побудова рекомендаційної мережі для децентралізованих P2P мереж – як неструктурованих, так і структурованих. Оскільки сервери у таких системах відсутні, то кожний окремий комп'ютер буде самостійно розраховувати собі рекомендації на основі доступної йому інформації. Будемо вважати, що кожний комп'ютер системи містить інформацію про k найближчих (поняття та метрика визначення відстані залежать від конкретного алгоритму реалізації мережі) вузлів мережі, розташованих на них файлів та статистику завантажень (i , можливо, оцінки користувачів файлам). Також він може запитати інформацію у своїх «сусідів» про їх «сусідів», про яких йому не відомо, і т. д. Чим

більшою буде глибина запиту, тим більше буде інформації про наявні у мережі файли і тим точнішими будуть рекомендації, але й довшим час формування рекомендацій та більшою ймовірністю виникнення проблеми «шторму трансляцій». Отже, при формуванні списку рекомендацій слід враховувати не тільки подоби користувачів, а й кількість переходів до файлів у мережі.

Було запропоновано наступну математичну модель роботи рекомендаційних систем у peer-to-peer мережах. У моделі наявні такі елементи:

$f_{i,k}$ – наявний файл k в бібліотеці користувача i ;

f_k – файл k в системі, незалежно від наявності його у користувача;

F_i – множина завантажених файлів користувача i ;

F – множина всіх фізично наявних файлів, файли дублюються у різних користувачів;

D – множина всіх оцінок користувачів файлам у системі (замість оцінки можна використовувати, наприклад, факт завантаження файлу, тоді D може приймати значення $\{0, 1\}$);

D_i – множина оцінок файлам користувача i ; $d_{i,k}$ – оцінка користувачем i окремого файлу k , $d_{i,k} \in [-1..1]$;

$v_{i,j}$ – коефіцієнт схожості вподобань користувачів i та j ;

p_i – ймовірність наявності в мережі користувача i в довільний момент часу;

$e(i, k)$ – наявність файлу k у користувача i (1 або 0);

$c_k(i, j)$ – мінімальна кількість переходів, яку потрібно здійснити від користувача i до користувача j ;

$c(i, k)$ – мінімальна кількість переходів від користувача i до користувача, в якого наявний файл k ;

$U(k)$ – множина користувачів, які мають завантажений файл k ;

$|U|$ – потужність множини, кількість елементів у множині (Приклад: $|U(k)|$ – кількість представлень файлу k);

N – кількість користувачів.

Нехай користувач i не має у власних списках F_i та D_i файлу k : $e(i, k)=0$, але відомо що цей файл є у наявності у користувача j . Тоді пропонуємо використати такі величини для визначення необхідності рекомендувати цей файл поточному користувачу:

$c(i, j)>0$ – чим менше, тим краще, бо менший шлях до передавання інформації;

$v_{i,j}$ – чим більше, тим краще, бо більше шансів схожих вподобань у користувачів;

$d_{j,k}$ – чим більше, тим краще, бо файл оцінено як хороший користувачем j ;

$|U(k)|$ – популярність файлу, чим більше, тим краще, бо розподілене отримання файлу значно скоротить час його завантаження. Впливає на оцінювання не безпосередньо, бо зі збільшенням цього числа буде більша множина рекомендацій від «сусідів».

Вага рекомендації від користувача j становитиме

$$r(i, k, j) = \frac{v_{i,j} d_{j,k} p_j}{c(i, j)}. \quad (3)$$

Оцінкою швидкості завантаження файлу з одного джерела є:

$$t(i, k, j) \sim p_i p_j c(i, j), \quad (4)$$

де враховано ймовірність підключення до мережі двох користувачів одночасно та кількість сегментів, які потрібно подолати, щоб дістатися до файлу. Швидкість завантаження з групи джерел можна визначити наступним чином:

$$t(i, k) \sim \left(\sum_j^N \frac{e(j, k)}{p_i p_j c(i, j)} \right)^{-1}. \quad (5)$$

Алгоритм визначення рекомендації:

1) Обираємо глибину аналізування $n > 0$.

2) Для обраної глибини знаходимо множину рекомендаційних ваг для користувача i за виразом

$$R_i = \bigcup_{k=1}^{|F|} R(i, k), \quad (6)$$

$$\text{де } R(i, k) = \sum_{j=1}^N r(i, k, j) \begin{cases} 1, & 1 \leq c(i, j) \leq n \\ 0, & c(i, j) > n \end{cases}.$$

3) Для користувача обираємо рекомендацію як $k_i = k \text{ of } \max(R_i)$. Якщо треба створити список рекомендацій, обираємо M найкращих і ранжуємо у порядку спадання значення R_i .

Підсилення або послаблення критерію рекомендації можна реалізувати коефіцієнтами $\alpha, \beta, \gamma, \delta$, якщо скористатися наступним виразом розрахунку ваги рекомендації:

$$r(i, k, j) = \frac{v_{i,j}^\alpha d_{j,k}^\beta p_j^\gamma}{c(i, j)^\delta}. \quad (7)$$

Розглянемо різні випадки, при яких у P2P мережах може виникнути необхідність надання користувачу списків рекомендацій:

1) користувач нічого не шукає, а тільки зайшов у додаток \rightarrow надаємо користувачу список рекомендацій з огляду на його вподобання.

2) користувач шукає конкретний файл, що був доданий до мережі раніше, за його ключем (хеш-значенням); файл не знайдено з різних причин (комп'ютери, що містять файл або таблиці маршрутизації до файлу, вийшли з мережі, застосовуються технології P2P з ймовірнісним методом пошуку, наприклад, неструктуровані мережі та RBFS тощо) \rightarrow надаємо користувачу список рекомендацій з огляду на його вподобання і пошуковий запит.

3) користувач шукає файли за ключовими словами (залежно від практичної реалізації файлообмінні системи можуть містити таку функцію, якщо зберігають не тільки файли, а й ключові слова); хеш-значення ключових слів, представлених у запиті, не знайдено у мережі з різних причин \rightarrow надаємо користувачу список рекомендацій з огляду на його вподобання та пошуковий запит. Або знайдено хеш-значення ключових слів, їм відповідає декілька різних файлів \rightarrow формуємо пошукову видачу, ранжуємо список результатів рекомендаційною системою.

На основі запропонованої математичної моделі пропонується такий метод формування рекомендацій у P2P мережах для (1) випадку використання:

1-й етап. Збирається інформація з таблиць маршрутизації комп'ютерів користувача та його «сусідів» і їх «сусідів» з глибиною пошуку n . Необхідною інформацією є хеш-значення файлів, їх розташування і статистика завантажень (та оцінки користувачів за наявності).

2-й етап. На основі зібраної інформації для кожного файлу у створеному на кроці 1 списку обчислюється R_i за формулою (7) із запропонованої математичної моделі, але тільки для файлів, які поточний користувач раніше не завантажував.

3-й етап. Обираємо M файлів з найвищим значенням R_i та ранжуємо у порядку спадання цього значення.

4-й етап. Надаємо користувачу список рекомендованих файлів, сформований на кроці 3.

5-й етап. Зберігаємо шлях до рекомендованих файлів у таблиці маршрутизації на час t_x для швидкого їх пошуку на випадок, якщо протягом заданого проміжку часу користувач ними зацікавиться.

Результати досліджень. У роботі було розроблено математичну модель рекоменда-

ційної системи для децентралізованої P2P мережі з огляду на вподобання користувачів та кількість переходів до завантаження даних. Також розроблено метод формування рекомендацій для користувачів комп'ютерної мережі типу peer to peer на основі запропонованої математичної моделі.

Обговорення результатів. Запропонована математична модель рекомендаційної системи для децентралізованої P2P розрахована на випадок, коли користувачам слід надати загальні рекомендації, незалежно від їх пошукового запиту. Рекомендації базуються на прогнозуванні вподобань користувачів, вподобання прогнозуються на основі зібраної інформації про дії схожих користувачів у системі. Також при формуванні вподобань враховується кількість переходів до потрібної інформації, щоб оптимізувати час формування рекомендацій та завантаження файлів.

Висновки. У цій статті проведено дослідження існуючих алгоритмів роботи peer to peer мереж. Проведено порівняльний аналіз методів пошуку даних у централізованих, децентралізованих неструктурованих та децентралізованих структурованих однорангових мережах. Виявлено, що однорангові мережі не застосовують для покращення пошуку даних рекомендаційні системи. Розроблено математичну модель рекомендаційної системи для децентралізованої P2P мережі з огляду на вподобання користувачів та кількість переходів до завантаження даних. Розроблено метод формування рекомендацій для користувачів комп'ютерної мережі типу peer to peer на основі запропонованої математичної моделі.

Наукова новизна проведеного дослідження полягає в такому:

– Розроблено математичну модель рекомендаційної системи для однорангової комп'ютерної мережі типу peer to peer для подальшої розробки комп'ютерної імітаційної моделі та тестування алгоритмів рекомендаційних і пошукових систем.

– Розроблено метод формування рекомендацій для користувачів комп'ютерної мережі типу peer to peer, що дозволяє врахувати особливості розподіленого зберігання даних у мережі та полегшити пошук потрібної інформації у ній.

Практичне значення. Запропоновано формули для відбору файлів у рекомендаційні списки, що враховують дії користувачів у мережі та час пошуку файлів.

Перспективи подальших досліджень.

Запропоновані модель та метод формування списку рекомендацій у peer to peer мережі розраховані на загальний випадок і не прив'язані до конкретного пошукового запиту. У подальших дослідженнях планується дослідити випадки, коли треба сформувати користувачу списки рекомендацій у відповідь на невдалий пошуковий запит з огляду на його зміст. В такому випадку рекомендації мають бути близькими до пошукового запиту, а при їх формуванні можна враховувати інформацію, зібрану з інших комп'ютерів під час відповідного пошукового запиту. Також планується розробити відповідні алгоритми та програмну модель для тестування запропонованих методів.

Список використаних джерел / References

- [1] Ju. Riposo, *Diffusion on the Peer-to-Peer Network*, LAP LAMBERT Academic Publishing, 2022.
- [2] S. G. M. Koo, *Multimedia Content Distribution Using Peer-to-Peer Overlay Networks: The Design and Analysis of the Next Generation Peer-to-Peer Networks*, VDM Verlag Dr. Müller, 2008.
- [3] A. Bellet, R. Guerraoui, M. Taziki, and M. Tommasi, "Personalized and private peer-to-peer machine learning", in *Proc. Twenty-First Int. Conf. on Artificial Intelligence and Statistics*, vol. 84, 2018, pp. 473-481. [Online]. Available: <http://proceedings.mlr.press/v84/bellet18a/bellet18a.pdf>.
- [4] D. S. Milojevic, V. Kalogeraki, R. Lukose et al., "Peer-to-peer computing", Technical Report HPL-2002-57, HP Labs, 2002. [Online]. Available: <https://www.cs.kau.se/cs/education/courses/dvad02/p2/seminar4/Papers/HPL-2002-57R1.pdf>.
- [5] D. Ricardo, *Introduction to Bitcoin: Understanding Peer-to-Peer Networks, Digital Signatures, the Blockchain, Proof-of-Work, Mining, Network Attacks, Bitcoin Core ... Safety*, Expiscor Books; 1st ed., 2021.
- [6] D. Zeinalipour-Yazti, V. Kalogeraki, and D. Gunopulos, "Information retrieval techniques for peer-to-peer networks", *Computing in Science & Engineering*, vol. 6, no. 4, pp. 20-26, 2004. doi: 10.1109/MCSE.2004.12.
- [7] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A survey and comparison of

- peer-to-peer overlay network schemes", *IEEE Communications survey and tutorial*, 2004. [Online]. Available: <https://snap.stanford.edu/class/cs224w-readings/lua04p2p.pdf>.
- [8] The BitTorrent Protocol Specification, 2017. [Online]. Available: http://www.bittorrent.org/beps/bep_0003.html.
- [9] V. Kalogeraki, D. Gunopulos, and D. Zeinalipour-Yazti, "A local search mechanism for peer-to-peer networks", *Proc. CIKM'02*, McLean VA, USA, 2002. [Online]. Available: <http://alumni.cs.ucr.edu/~csyiazti/downloads/papers/cikm02/cikm02.pdf>.
- [10] D. Zeinalipour-Yazti, "Information retrieval in peer-to-peer systems", M.Sc thesis, Dept. of Computer Science, University of California Riverside, 2003. [Online]. Available: <http://alumni.cs.ucr.edu/~csyiazti/papers/msc/html/>.
- [11] N. Shukla, D. Datta, and M. Pandey, "Towards software defined low maintenance structured peer-to-peer overlays", *Peer-to-Peer Netw. Appl.*, vol. 14, pp. 1242-1260, 2021. doi: 10.1007/s12083-021-01112-7.
- [12] Y. Klots, I. Muliar, V. Cheshun, and O. Burdyug, "Use of distributed hash tables to provide access to cloud services", *Collection of scientific works of the Military Institute of Kyiv National Taras Shevchenko University*, vol. 11 (67), pp. 85-95, 2020.
- [13] Yb. Shen, and T. R. Gadekallu, "Resource search method of mobile intelligent education system based on distributed hash table", *Mobile Netw Appl.*, 27, pp. 1199-1208, 2022. doi: 10.1007/s11036-022-01940-8.
- [14] S. Ratnasamy, I. Stoica, and S. Shenker, "Routing algorithms for DHTs: Some open questions. Peer-to-peer systems", in *Lecture Notes in Computer Science*, vol. 2429. Springer, Berlin, Heidelberg, 2002, pp. 45-52. doi: 10.1007/3-540-45748-8_4.
- [15] Kademia: A Design Specification, 2010. [Online]. Available: <https://xlattice.sourceforge.net/components/protocol/kademia/specs.html>.
- [16] I. Stoica, R. Morris, D. R. Karger, M. F. Kaashoek, and H. Chord Balakrishnan, "A scalable peer-to-peer lookup service for internet applications", *ACM SIGCOMM Computer Communication Review*, vol. 31 (4), 2001. doi: 10.1145/964723.383071.

V. V. Mikhav¹, *Postgraduate Sstudent*,
e-mail: mihaw.wolodymyr@gmail.com

Ye. V. Meleshko¹, *Dr. Tech. Sc., Professor*,

O. M. Driev¹, *Cand. Tech. Sc., Associate Professor*,

A. O. Lavdanskyy², *Cand. Tech. Sc., Associate Professor*

¹Central Ukrainian National Technical University

Universitetskiy ave., 8, Kropyvnytskyi, 25006, Ukraine

²Cherkasy State Technological University

Shevchenko blvd, 460, Cherkasy, 18006, Ukraine

A MODEL OF RECOMMENDER SYSTEM FOR P2P COMPUTER NETWORKS

In this article, a research of peer-to-peer network algorithms is conducted. A comparative analysis of data search methods in centralized, decentralized unstructured and decentralized structured peer-to-peer networks is carried out. It has been found that the existing methods of peer-to-peer networks do not use recommender systems to improve data search. A mathematical model of a recommender system for a decentralized P2P network has been developed, taking into account user preferences and the number of transitions to download data. A method of forming recommendations for peer-to-peer computer network users based on the proposed mathematical model has been developed.

In P2P networks, there is a problem of indexing and searching files on different network devices. For various reasons, searched files may not be available to a user, even if they were previously added to a system and indexed. For example, computers containing the desired file or routing tables to it or its parts have left a network, or P2P network construction technologies are used with probabilistic search methods that do not always find files located far from the user's computer, etc.

Recommender systems are used to build lists of recommendations for users based on their previous actions, including likes, ratings, views, downloads, etc. They make it possible to facilitate the search in a system with a large number of objects, supplementing the classic search, and in some situations even substituting the search. Recommender systems can also be used to rank classical search results. Thus, they can be combined with conventional search algorithms in various ways. In P2P networks, the use of recommender systems can have additional benefits. If a user is searching for a specific file that has been previously added to a network and the file is not found for various reasons, you can provide the user a list of recommendations based on his/her preferences and possibly search query.

A model and method of forming recommendation lists in peer-to-peer networks proposed in the paper are designed for the general case and are not tied to a specific search query, can be applied in unstructured and structured decentralized P2P networks to familiarize a user with a content that he/she may like based on the prediction of his/her preferences. This can increase the overall interest of users in a content of a network.

Keywords: *computer networks, peer-to-peer networks, P2P, recommender systems, data search.*

Стаття надійшла 04.01.2023

Прийнято 04.02.2023