

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

ГОРДІЙЧУК-БУБЛІВСЬКА ОЛЕНА ВАСИЛІВНА

Кваліфікаційна наукова
праця на правах рукопису
УДК 004.652

**Методи та засоби опрацювання великих даних в
розподілених інформаційних системах**

122 – комп’ютерні науки

**Дисертація на здобуття наукового ступеня
доктора філософії**

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

_____ /О.В. Гордійчук-Бублівська/

Науковий керівник

Кирик Мар’ян Іванович

доктор технічних наук, професор

АНОТАЦІЯ

Гордійчук-Бублівська О.В. Методи та засоби опрацювання великих даних в розподілених інформаційних системах. – Кваліфікаційна наукова праця на правах рукопису. Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 122 “Комп’ютерні науки”. – Національний університет «Львівська політехніка», Львів, 2024.

Зміст анотації. Дисертація присвячена розробленню методів і засобів для створення оброблення великих даних в розподілених інформаційних системах. В сучасних інформаційних системах різного призначення дедалі частіше виникає проблема своєчасного надання результатів обчислень користувачам, аналізу багатовимірних інформаційних масивів та відкинення надлишкових даних, знаходження взаємозв’язків між подіями. Алгоритми оброблення даних дають змогу, зокрема, визначати і пропонувати відповідні товари і послуги для користувачів комерційних і промислових систем інтернету речей, провівши аналіз їх попередніх дій. Це, у свою чергу, сприяє цілісному вирішенню завдання підвищення рівня обслуговування користувачів. Рекомендаційні алгоритми також використовуються для автоматизації процесів моніторингу стану системи, управління нею, окремими її компонентами. Враховуючи сказане, в умовах інтенсивного розвитку систем машинного навчання та штучного інтелекту, адаптація та вдосконалення існуючих, а також створення нових методів опрацювання великих даних є актуальною науковою задачею, розв’язання якої сприяє більш ефективному функціонуванню розподілених інформаційних систем.

В першому розділі «Аналіз функціонування основних технологій та методів оброблення великих обсягів даних в розподілених системах» проаналізовано ключові тенденції розвитку методів оброблення великих даних. Досліджено алгоритми та технології для оптимізації даних в розподілених інформаційних системах. Розглянуто рекомендаційні системи як засіб для оброблення інформації про користувачів та їх взаємодію з товарами і

послугами. Також проаналізовано переваги методів інтелектуального оброблення великих обсягів даних та машинного навчання.

В другому розділі «Дослідження методів та алгоритмів автоматичного оброблення великих даних» було розглянуто методи матричної факторизації матриць даних у вигляді розкладу на множники для спрощення здійснення подальшого аналізу за допомогою алгоритмів сингулярної декомпозиції SVD, Fed SVD, Funk SVD. Також проведено аналіз роботи алгоритму сингулярної декомпозиції в розподілених системах. Сформовано загальну структуру розподіленої рекомендаційної системи для формування передбачень користувачам про наступні товари чи послуги на основі попередніх дій на основі модифікованого SVD.

В третьому розділі «Розроблення моделі оброблення даних в розподілених інформаційних системах з використанням методів машинного навчання» було запропоновано алгоритми формування рекомендацій для підвищення ефективності обробки великих даних в промислових системах, зокрема таких, які інтегрують засоби промислового інтернету речей. Розроблено алгоритми для розподіленої обробки даних великих обсягів в промислових системах з обміном результатами навчання між вузлами, а також надання рекомендацій на основі меншої кількості даних про користувачів і товари, порівняно з існуючими. Усі алгоритми було зображено за допомогою блок-схем. Розроблено метод модифікованого Fed SVD, що передбачає використання локально згенерованих даних для захисту перед пересиланням по мережі, що дозволяє підвищити надійність та точність, скоротити тривалість обчислень. Вдосконалено метод надання рекомендацій Funk SVD для підвищення точності і швидкості оброблення інформації за рахунок оброблення меншої кількості, але більш важливих даних. Представлено модель розподіленої рекомендаційної системи на основі запропонованих методів та результати її роботи у вигляді графіків.

В четвертому розділі «Практичні аспекти реалізації моделі розподіленої обробки даних великих обсягів» запропоновано розроблений метод розподіленого формування рекомендацій Mod FedSVD, який, за допомогою

модифікованого методу Funk SVD, дозволяє визначення рекомендацій користувачами з використанням власних даних; також метод передбачає виділення приватної та публічної частини локальних результатів для оновлення глобальної моделі рекомендацій з меншими ризиками для даних з кінцевих пристроїв. Представлено результати розроблення архітектури розподіленої рекомендаційної системи для оброблення великих даних, обґрунтовано вибір технологій для її реалізації. Також визначено функціонал системи та засоби, які забезпечать її стабільну роботу. Представлено програмну модель визначення рекомендацій користувачам. Здійснено аналіз отриманих результатів.

Основні наукові результати дисертації опубліковано в 25 працях, зокрема: 11 статей – у наукових фахових періодичних виданнях України; 5 - у закордонних фахових періодичних виданнях; 9 публікацій – у матеріалах міжнародних та всеукраїнських наукових, науково-технічних конференцій.

Ключові слова: великі дані, розподілені системи, Промисловий Інтернет Речей (ІоТ), рекомендаційні системи, матрична факторизація даних, SVD, Funk SVD.

ABSTRACT

Hordiichuk-Bublivska O. Methods and means of big data processing in distributed information systems. - Qualifying scientific work on the rights of the manuscript. Dissertation for the degree of Doctor of Philosophy in the specialty 122 "Computer Science". - National University "Lviv Polytechnic", Lviv, 2024.

Abstract content. The dissertation is devoted to the development of methods and tools for creating big data processing in distributed information systems. In modern information systems of various purposes, the problem of timely provision of calculation results to users, analysis of multidimensional information arrays and rejection of redundant data, finding relationships between events is increasingly occurring. Data processing algorithms make it possible, in particular, to determine and offer appropriate goods and services for users of commercial and industrial Internet of Things systems, after analyzing their previous actions. This, in turn, contributes to a holistic solution to the task of improving the level of user service. Recommendation algorithms are also used to automate the processes of monitoring the state of the system and managing it and its components. Considering the above, in the conditions of intensive development of machine learning and artificial intelligence systems, adaptation and improvement of existing, as well as creation of new methods of processing big data is an urgent scientific task, the solution of which contributes to more efficient functioning of distributed information systems.

The first chapter "Analysis of the functioning of the main technologies and methods of processing large volumes of data in distributed systems" analyzes the key trends in the development of methods of processing large data. Algorithms and technologies for data optimization in distributed information systems are studied. Recommender systems are considered as a means of processing information about users and their interaction with goods and services. The advantages of the methods of intelligent processing of large volumes of data and machine learning are also analyzed.

In the second chapter "Research of methods and algorithms for automatic processing of big data", the methods of matrix factorization of data matrices in the

form of factorization were considered to simplify further analysis using the algorithms of singular decomposition SVD, Fed SVD, and Funk SVD. An analysis of the operation of the singular decomposition algorithm in distributed systems was also carried out. The general structure of a distributed recommender system is formed to make predictions to users about the next goods or services based on previous actions based on the modified SVD.

In the third chapter, "Development of a data processing model in distributed information systems using machine learning methods", algorithms for generating recommendations for increasing the efficiency of big data processing in industrial systems, in particular those that integrate industrial Internet of Things tools, were proposed. Algorithms have been developed for distributed processing of large volumes of data in industrial systems with the exchange of learning results between nodes, as well as providing recommendations based on a smaller amount of data about users and products, compared to existing ones. All algorithms were depicted using flowcharts. A method of modified Fed SVD has been developed, which involves the use of locally generated data for protection before forwarding over the network, which allows for increased reliability and accuracy and reduces the duration of calculations. The Funk SVD recommendation method has been improved to increase the accuracy and speed of information processing by processing a smaller amount of but more important data. The model of the distributed recommender system based on the proposed methods and the results of its work in the form of graphs are presented.

In the fourth chapter, "Practical aspects of the implementation of the model of distributed processing of large volumes of data", the developed method of distributed formation of Mod FedSVD recommendations is proposed, which, with the help of the modified Funk SVD method, allows the definition of recommendations by users using their data; also, the method involves separating private and public parts of local results to update a global recommendation model with lower risks for data from end devices. The results of the development of the architecture of the distributed recommender system for processing big data are presented, and the choice of technologies for its implementation is substantiated. The functionality of the system

and the means that will ensure its stable operation are also determined. A software model for determining user recommendations is presented. An analysis of the obtained results was carried out.

The main scientific results of the dissertation were published in 25 works, in particular: 11 articles - in specialized scientific periodicals of Ukraine; 5 - in foreign professional periodicals; and 9 publications - in the materials of international and all-Ukrainian scientific, scientific, and technical conferences.

Keywords: big data, distributed systems, Industrial Internet of Things (IIoT), recommender systems, matrix factorization of data, SVD, Funk SVD.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Статті у фахових виданнях України:

1. Дослідження ефективності розподілених інфокомунікаційних систем на основі оброблення великих обсягів даних. Автори: Климаш М.М., Гордійчук – Бублівська О.В., Чайковський І.Б., Урікова О.М. Вісник університету “Україна”, січень, 2020
2. Дослідження оптимізації великих обсягів даних методами машинного навчання. Климаш М.М., Гордійчук – Бублівська О.В., Чайковський І.Б., Сиротинський О.І. Вісник Національного університету “Львівська політехніка”, 2020
3. “Аналіз концепції інтернету речей та динаміки її розвитку у різних галузях”, Вісник університету “Україна”, січень 2020, Пиріг Ю.В., Кайдан М.В, Гордійчук-Бублівська О.В, ст.171-186
4. М. Климаш, О. Гордійчук-Бублівська, І. Чайковський, Т. Данильченко, “Дослідження алгоритмів паралельного опрацювання інформації в базах даних”, Вісник «Інфокомунікаційні технології та електронна інженерія», №1, С. 51-62, 2021
5. Климаш М.М., Гордійчук-Бублівська О.В., Мрак В.Б., Браницький А.В., “Дослідження ефективності обробки великих даних в системах відеонагляду”, Інфокомунікаційні та комп’ютерні технології, №1 (01), 2021, ст.52-60.
6. М. Климаш, В. Мрак, О. Гордійчук-Бублівська, “Дослідження методів виділення динамічних об’єктів у відеопослідовностях”, Вісник «Інфокомунікаційні технології та електронна інженерія», №1, С. 63-75, 2021
7. М. Климаш, О. Гордійчук-Бублівська, Б. Коваль, “Модель інтелектуального аналізу даних в ІоТ”, Вісник «Інфокомунікаційні технології та електронна інженерія», №2, С. 11-18, 2021
8. Гордійчук-Бублівська О.В., Бешлей М.І., Кирик М.І., Климаш М.М, “Підвищення ефективності оброблення великих обсягів інформації

з використанням методу розподіленого аналізу даних”, ISSN 2412-4338 Телекомунікаційні та інформаційні технології. 2021. № 1 (70). DOI:10.31673/2412-4338.2021.021523

9. О. В. Гордійчук-Бублівська, Л. П. Фабрі. МАТРИЧНА ФАКТОРИЗАЦІЯ ВЕЛИКИХ ДАНИХ У ПРОМИСЛОВИХ СИСТЕМАХ. УІТ. 2022; Випуск 4, Номер 2: 68-73. <https://doi.org/10.23939/ujit2022.02.068>
10. М. Климаш, О. Гордійчук-Бублівська, І. Чайковський, О. Костів, “Дослідження ефективності використання розподілених баз даних в системах ІоТ”, Вісник «Інфокомунікаційні технології та електронна інженерія», Вип 2, №1, С. 12-18, 2022
11. О. Шпур, Я. Пиріг, О. Гордійчук-Бублівська, Ю. Пиріг, “Дослідження і проектування інформаційної системи вищого навчального закладу”, Вісник «Інфокомунікаційні технології та електронна інженерія», Вип 2, №1, С. 36-43, 2022

Статті у інших виданнях:

12. Distributed Singular Value Decomposition Method for Fast Data Processing in Recommendation Systems”, Przystupa, K.; Beshley, M.; Hordiichuk-Bublivska, O.; Kyryk, M.; Beshley, H.; Pyrih, J.; Selech, J. Energies 2021, 14, 2284
13. Beshley, M.; HordiichukBublivska, O.; Beshley, H.; Ivanochko, I. Data Optimization for Industrial IoT-Based Recommendation Systems. Electronics 2023, 12, 33. <https://doi.org/10.3390/electronics12010033>
14. Olena Hordiichuk-Bublivska, Halyna Beshley, Marian Kyryk, Yuliia Pyrih, Oksana Urikova, and Mykola Beshley. A Modified Federated Singular Value Decomposition Method for Big Data and ML Optimization in ІоТ Systems. In: Mikhailo Klymash, Andriy Luntovskyu, Mykola Beshley, Igor Melnyk, Alexander Schill. (eds) Emerging Networking in the Digital Transformation Age: Approaches, Protocols, Platforms, Best Practices, and Energy Efficiency. Lecture Notes in Electrical Engineering, 2023, Springer, Cham. 965, P.246-268

15. Yiming, Z.; Fang, X.; Hordiichuk-Bublivska, O.; Beshley, H.; Beshley, M. Modified Masking-Based Federated Singular Value Decomposition Method for Fast Anomaly Detection in Smart Grid Systems. *Energies* 2023, 16, 5996. <https://doi.org/10.3390/en16165996>
16. Olena Hordiichuk-Bublivska, Halyna Beshley, Natalia Kryvinska and Mykola Beshley. A masking-based federated singular value decomposition method for anomaly detection in industrial internet of things. *International Journal of Web and Grid Services*, Vol. 19, No. 3. pp 287-317. <https://doi.org/10.1504/IJWGS.2023.133502>

Матеріали конференцій:

17. Modeling and Research of Processing Big Data Sets in Distributed Information Systems. Mykhailo Klymash, Olena Hordiichuk-Bublivska, Ihor Tchaikovskiy, Yuriy Deschynskiy. TCSET -2020
18. Дослідження моделей побудови розподілених інформаційних систем з великими обсягами даних. М.М. Климаш, О.В. Гордійчук – Бублівська, І.Б. Чайковський, Т.Є. Данильченко. PCSET-2020
19. “Research of Microservices Features in Information Systems using Spring Boot”, Mykhajlo Klymash, Olena Hordiichuk- Bublivska, Ihor Tchaikovskiy and Yulia Pyrih, PROBLEMS OF INFOCOMMUNICATIONS.SCIENCE AND TECHNOLOGY PIC S&T’2020 6-9 October 2020, Kharkiv, Ukraine.
20. M. Klymash, I. Chaikovskiy, N. Syvkova, O. Hordiichuk-Bublivska and M. Kyryk, "Research of Distributed Data Processing in Corporate Information Systems," 2021 IEEE 16th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), 2021, pp. 24-27, doi: 10.1109/CADSM52681.2021.9385244.
21. M. Klymash, M. Kyryk, I. Demydov, O. Hordiichuk-Bublivska, H. Kopets and N. Pleskanka, “Research on Distributed Machine Learning Methods in Databases,” 2021 IEEE 4th International Conference on

- Advanced Information and Communication Technologies (AICT), 2021, pp. 128-131, doi: 10.1109/AICT52120.2021.9628949.
- 22.M. Klymash, M. Kyryk, O. Hordiichuk-Bublivska, L. Fabri, H. Kopets, “Big Data Analysis in IIoT Systems Using the Federated Machine Learning Method”, IEEE 16 International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), February 22 - 26, 2022, Lviv-Slavske, Ukraine.
- 23.“Big Data Analysis in Smart Grid Systems”, Yu Jun, Olena Hordiichuk-Bublivska, Yan Lingyu, Marian Kyryk, Mykola Beshley, Hu Jiwei, 18th IMEKO TC10 Conference “Measurement for Diagnostics, Optimisation and Control to Support Sustainability and Resilience” Warsaw, Poland, September 26–27, 2022
- 24.Model of large sparse datasets processing efficiency in IIoT. M. Klymash, O. Hordiichuk-Bublivska, M. Kyryk, T. Andrukhiv, Y. Pyriih, CADSM 2023
- 25.M. Klymash, M. Kyryk, O. Hordiichuk-Bublivska, L.Fabri , “Research on the Automated Decision-Making Effectiveness in Industrial Automation System” 2023 IEEE 5th International Conference on Advanced Information and Communication Technologies (AICT), 2023

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	14
ВСТУП.....	15
РОЗДІЛ 1. АНАЛІЗ ФУНКЦІОНУВАННЯ ОСНОВНИХ ТЕХНОЛОГІЙ ТА МЕТОДІВ ОБРОБКИ ВЕЛИКИХ ОБСЯГІВ ДАНИХ В РОЗПОДІЛЕНИХ СИСТЕМАХ.....	21
1.1. Тенденції розвитку обробки великих обсягів даних.	21
1.2. Ключові технології для оптимізації даних в розподілених системах.....	21
1.3. Рекомендаційні системи як засіб для обробки даних	28
1.4. Перспективи переходу до методів інтелектуальної обробки даних великих обсягів.....	36
Висновки до першого розділу.....	28
РОЗДІЛ 2. ДОСЛІДЖЕННЯ МЕТОДІВ ТА АЛГОРИТМІВ АВТОМАТИЧНОЇ ОБРОБКИ ВЕЛИКИХ ДАНИХ.	39
2.1. Ефективність використання алгоритму сингулярної декомпозиції.....	39
2.2. Аналіз роботи алгоритму сингулярної декомпозиції в розподілених системах	Ошибка! Закладка не определена.
2.3. Дослідження алгоритму федеративного сингулярного розкладу даних.	Ошибка! Закладка не определена.
2.4. Принцип обробки даних в розподілених системах за допомогою методів машинного навчання.....	67
Висновки до другого розділу.	69
РОЗДІЛ 3. РОЗРОБЛЕННЯ МОДЕЛІ ОБРОБКИ ДАНИХ В РОЗПОДІЛЕНИХ СИСТЕМАХ З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ.	70
3.1. Дослідження використання машинного навчання для управління ресурсами інформаційно-комунікаційних систем промислової інфраструктури	70

	13
3.2. Алгоритми машинного навчання для обробки даних користувачів промислових систем.....	Ошибка! Закладка не определена.
3.3. Аналіз використання алгоритмів машинного навчання для підвищення ефективності обробки великих даних	77
3.4. Модель для обробки даних великих обсягів в IoT системах	84
Висновки до третього розділу.....	93
РОЗДІЛ 4. ПРАКТИЧНІ АСПЕКТИ РЕАЛІЗАЦІЇ МОДЕЛІ РОЗПОДІЛЕНОЇ ОБРОБКИ ДАНИХ ВЕЛИКИХ ОБСЯГІВ.	94
4.1. Принципи роботи запропонованої моделі розподіленої обробки інформаційно-комунікаційних систем промислової інфраструктури	94
4.2. Опис особливостей проведеного експерименту	94
4.3. Результати проведеного експерименту.....	105
4.4. Практичні аспекти реалізації створеної моделі для обробки даних великих обсягів в IoT системах	113
Висновки до четвертого розділу.....	120
ВИСНОВКИ.....	122
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	124
ДОДАТОК А. АКТИ ВПРОВАДЖЕННЯ	144

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

1. SVD – Singular Value Decomposition;
2. ML – Machine Learning;
3. FL – Federated Learning;
4. RS – Recommender Systems;
5. IoT – Internet of Things;
6. IIoT – Industrial Internet of Things;
7. DS – Distributed Systems.

ВСТУП

Актуальність теми. Оброблення даних і їх оптимізація відіграють важливу роль у роботі інформаційних систем. Використання хмарних технологій сприяє зберіганню значного обсягу даних. Пошук та аналіз інформації у створеному об'ємі дає змогу, зокрема, контролювати стан систем Промислового Інтернету речей (ІоТ) і підвищує їх ефективність. Збір даних із різних підсистем надає можливість завчасно виявляти та запобігати проблемам. Рекомендаційні системи в епоху ІоТ надають персоналізовані рекомендації на основі історичних наборів даних користувачів, зібраних з персональних пристроїв. Також рекомендаційні системи забезпечують ефективний процес прийняття рішень, пропонуючи або підказуючи зацікавленим користувачам відповідні продукти, ресурси та інформацію. Розширення переліку послуг, які надають сучасні промислові системи, змінило підхід до їх проектування та використання. Замість централізованих систем, якими керують і які обслуговують виключно люди, тепер використовуються гнучкі та децентралізовані системи. Промисловий Інтернет речей (ІоТ), хмарне виробництво (СМfg) і машинне навчання дають змогу створювати більш ефективні промислові системи та надавати користувачам більш якісні послуги [1].

У промисловому Інтернеті речей дані збираються зі значної кількості термінальних і периферійних пристроїв. Розумне виробництво вважається єдиною системою, що включає етапи виробництва та продажу клієнтам. У промисловому Інтернеті речей інформація, зібрана з різних джерел, допомагає виявити слабкі місця у виробничому процесі та підвищити якість обслуговування клієнтів. Для трансформації існуючих або створення нових розумних виробництв необхідно використовувати сучасні програмно-технічні засоби [2].

Велика кількість кінцевих пристроїв, які виконують функції локального збору та аналізу даних, надає змогу зменшити навантаження на центральні пристрої керування та краще врахувати специфіку конкретних підсистем [3].

Завдяки методам машинного навчання та штучного інтелекту промислові системи здатні навчатися на власних наборах даних і виправляти аварійні ситуації ще до їх виникнення [4]. Такий підхід спрощує процес усунення неполадок, організацію виробничого процесу та доставку товарів безпосередньо споживачам. Зміна параметрів системи може розглядатися як сигнал про необхідність модифікацій у алгоритмах її роботи. Проблема розумного промислового виробництва полягає в необхідності постійно обробляти значну кількість даних, які надходять від різних пристроїв. Для одержання швидкого і релевантного результату оброблення таких даних слід ефективно відділити найважливішу інформацію, що само по собі є актуальним завданням [1].

Над зазначеними проблемами працювали ряд дослідників (F. Ortega, A. González-Prieto, Z. Wang, H. Wu, Z. Jiang, P. Ju, J. Yang, Z. Zhou, Chen, M. Khan, Y. Jin, M. Li, Y. Xiang, C. Jiang, T. Zhao, Z. Ding). Однак досі виникають труднощі при необхідності визначити відповідності між даними про користувачів, щодо формування системи своєчасного та ефективного надання рекомендацій, використання їх для управління системою.

Тому, в умовах інтенсивного розвитку систем машинного навчання, штучного інтелекту, промислового інтернету речей, адаптація та вдосконалення існуючих, а також створення нових методів опрацювання великих даних є актуальною науковою задачею, розв'язання якої сприяє більш ефективному функціонуванню розподілених інформаційних систем.

Зв'язок роботи з науковими програмами, планами і темами. Дисертація виконувалася відповідно до тематичних планів науково-дослідних робіт Національного університету “Львівська політехніка”. Зокрема – в межах наукових досліджень, які виконуються відповідно до держбюджетної роботи кафедри телекомунікацій «Розроблення інноваційних методів та моделей побудови інтелектуальних інформаційно-комунікаційних систем для цифровізації промисловості, (2022–2024 рр.), № держреєстрації 0122U000817.

Метою дисертаційної роботи є розроблення методів та засобів опрацювання великих даних в розподілених інформаційних системах.

Для досягнення поставленої мети в роботі розв'язано такі **задачі**:

1. Проведено аналіз сучасного стану методів оброблення великих даних в розподілених інформаційних системах;
2. Вдосконалено метод SVD для автоматизації виконання функцій і завдань виробничого управління у багаторівневих структурах за допомогою рекомендаційних систем;
3. Розроблено модель автоматизації функціональних завдань керування, аналізу та оцінювання ефективності промислових систем розподіленої обробки даних;
4. Вдосконалено метод розподіленого оброблення користувацьких даних Fed SVD, забезпечуючи їх захист від стороннього втручання, з метою побудови рекомендаційних систем в ІоТ;
5. Вдосконалено метод надання рекомендацій в промислових системах Funk SVD, використовуючи адаптивну кількість даних про користувачів і товари для прийняття рішень про відповідні їм товари чи послуги;
6. Розроблено метод розподіленого формування рекомендацій Mod FedSVD, який, за допомогою модифікованого методу Funk SVD, передбачає визначення рекомендацій користувачами з використанням власних даних, а також передбачає виділення приватної та публічної частини локальних результатів для оновлення глобальної моделі рекомендацій;
7. Розроблено модель розподіленої рекомендаційної системи на основі запропонованого Mod FedSVD, що дозволяє підвищити точність та швидкість навчання глобальної системи рекомендації, обмінюючись публічними частинами локальних результатів, зберігаючи захищеними приватні дані.

Об'єкт дослідження – процеси оброблення великих даних.

Предмет дослідження – методи та засоби опрацювання великих даних в розподілених інформаційних системах.

Методи дослідження. У процесі розробки методу аналізу великих даних використано за основу алгоритм сингулярної декомпозиції. Для розроблення методів оброблення інформації в системі індустріального інтернету речей про користувачів використано методи машинного навчання, математичну статистику. Для побудови програмних продуктів застосовано об'єктно-орієнтоване проектування, методи паралельних та розподілених обчислень.

Наукова новизна одержаних результатів полягає в тому що:

- *вдосконалено* метод SVD, що передбачає розпаралелювання процесу його обчислення між різними вузлами, тим самим скорочуючи тривалість обчислень та зменшуючи навантаження на компоненти, в порівнянні з існуючим методом, а також дозволяє створення моделі автоматизації завдань керування, аналізу й оцінювання ефективності систем розподіленої обробки даних;
- *вдосконалено* метод розподіленого оброблення користувацьких даних Fed SVD, що використовує генерування даних для захисту даних користувачів безпосередньо на локальному пристрої, тим самим забезпечуючи їх від стороннього втручання при пересиланні в рекомендаційних системах, скорочення тривалості обчислень в порівнянні з існуючим методом;
- *вдосконалено* метод Funk SVD, що передбачає використання не всіх вихідних даних, а їх адаптивну кількість для прийняття швидших рішень рекомендаційною системою в порівнянні з існуючим методом, при цьому зберігаючи досить високу точність обчислень;
- *розроблено* метод розподіленого формування рекомендацій ModFedSVD, що дозволяє обчислення модифікованого Funk SVD кінцевими пристроями на основі власних даних, а також виокремлення приватної та публічної частини результату; таким чином передбачається можливість обміну публічними даними між різними пристроями та оновлення глобальної моделі рекомендацій, забезпечуючи кращу швидкість і

точність обчислень, а також безпеку приватних даних користувачів в порівнянні з існуючими методами.

Практичне значення одержаних результатів. Практична цінність роботи полягає у доведенні отриманих наукових результатів технологій, методик, алгоритмів та програмних продуктів оброблення великих даних в розподілених рекомендаційних системах. На основі запропонованих методів було розроблено модель розподіленої інформаційної системи промислового інтернету речей для оброблення даних про її користувачів. Для роботи пов'язаних із цією системою рекомендаційних систем розроблено вдосконалений розподілений алгоритм сингулярної декомпозиції даних, що дає змогу на 10% швидше і на 5% точніше опрацьовувати інформацію в порівнянні з існуючими методами. Розроблено програмну модель, у якій реалізовані усі наукові результати.

Особистий внесок здобувача. Основні положення та результати дисертаційної роботи одержані автором самостійно. Особисто здобувачеві належать наступні наукові результати: розроблено алгоритм збору користувацьких даних в розподілених системах [1, 2], розроблено метод оптимізації великих даних [3, 10], розроблено вдосконалений метод надання рекомендацій на основі адаптивної кількості даних про користувачів [4, 7, 8], розроблено модель розподіленого опрацювання інформації [5, 6, 9, 10, 11].

1. Апробація результатів дисертації. Результати дисертаційної роботи доповідались на конференціях: TCSET-2020,2022, PCSET-2020, PROBLEMS OF INFOCOMMUNICATIONS, SCIENCE AND TECHNOLOGY (PIC S&T) 2020, International Conference on the Experience of Designing and Application of CAD Systems (CADSM), 2021, 2023, International Conference on Advanced Information and Communication Technologies (AICT) 2021, 18th IMEKO TC10 Conference “Measurement for Diagnostics, Optimisation and Control to Support Sustainability and Resilience” 2022. Також результати доповідались на семінарах кафедри телекомунікацій «Національного університету «Львівська політехніка».

Публікації. За результатами виконаних досліджень опубліковано 25 наукових праць, із них 11 статей – у фахових виданнях України, 5 публікації – у міжнародних виданнях, 9 – у збірниках наукових праць конференцій.

Структура та обсяг дисертації. Дисертація складається із вступу, чотирьох розділів, висновків, списку використаних джерел із 162 найменувань та додатків. Повний обсяг дисертації складає 144 сторінки, основний зміст викладено на 125 сторінках, де наведено 135 рисунків.

РОЗДІЛ 1. АНАЛІЗ ФУНКЦІОНУВАННЯ ОСНОВНИХ ТЕХНОЛОГІЙ ТА МЕТОДІВ ОБРОБКИ ВЕЛИКИХ ОБСЯГІВ ДАНИХ В РОЗПОДІЛЕНИХ СИСТЕМАХ.

У розділі проаналізовано проблеми обробки великих обсягів даних в розподілених системах. Розглянуто підходи до покращення ефективності оброблення даних користувачів. Наведено актуальні дослідження в даній галузі та визначено основні задачі, які потребують вирішення.

Проаналізовано методи оптимізації великих даних для більш ефективної подальшої обробки. Визначено доцільність цифровізації існуючих промислових систем для ефективнішого виконання завдань. Встановлено актуальність дослідження даної тематики та перспективи переходу до більш інтелектуальних методів в умовах стрімкого збільшення кількості даних, що потребують обробки. Результати розділу опубліковано в працях автора [1, 2, 3, 4, 5].

1.1. Тенденції розвитку обробки великих обсягів даних.

Використання хмарних технологій сприяє зберіганню значного обсягу даних. Пошук та аналіз інформації дозволяє контролювати стан промислових систем Інтернету речей (IIoT) і підвищує їх ефективність. Збір даних із різних підсистем дозволяє завчасно виявляти та запобігати проблемам. Системи рекомендацій в епоху IoT надають персоналізовані рекомендації на основі історичних наборів даних користувачів, зібраних з пристроїв IoT. Рекомендації забезпечують ефективний процес прийняття рішень, пропонуючи відповідні продукти, ресурси та інформацію. Розширення переліку послуг, які надають сучасні промислові системи, змінило підхід до їх проектування та використання. Замість централізованих систем, якими керують і обслуговують виключно люди, тепер використовуються гнучкі та децентралізовані системи. Промисловий Інтернет речей (IIoT), хмарне виробництво (CMfg) і машинне навчання дозволяють створювати більш ефективні промислові системи та надавати користувачам якісні послуги. Детальніше розглянуто в працях [1,2,3].

У промисловому Інтернеті речей дані збираються зі значної кількості термінальних і периферійних пристроїв. Розумне виробництво вважається

єдиною системою, що включає етапи виробництва та продажу клієнтам. У промисловому Інтернеті речей інформація, зібрана з різних джерел, допомагає виявити слабкі місця у виробничому процесі та підвищити якість обслуговування клієнтів [4,5,6,7]. Для трансформації існуючих або створення нових розумних виробництв необхідно використовувати сучасні програмно-технічні засоби. Велика кількість кінцевих пристроїв, які виконують функції локального збору та аналізу даних, дозволяють зменшити навантаження на центральні пристрої керування та краще врахувати специфіку конкретних підсистем. Завдяки методам машинного навчання та штучного інтелекту промислові системи здатні навчатися на власних наборах даних і виправляти аварійні ситуації ще до їх виникнення. Такий підхід спрощує процес усунення неполадок, організацію виробничого процесу та доставку товарів безпосередньо споживачам. Зміна параметрів системи може розглядатися як сигнал про необхідність модифікації її роботи. Проблема розумного промислового виробництва полягає в необхідності постійно обробляти значну кількість даних, що надходять від різних пристроїв. Для швидкого результату слід визначити найважливішу інформацію, а непотрібну для результату відкинути [8,9,10,11].

Розподіл виробничих завдань між кількома пристроями знижує навантаження і прискорює процес розрахунку. Розподілена архітектура промислових систем сприяє більш ефективній роботі інтелектуального виробництва, швидшому отриманню результатів. Децентралізація функцій управління дозволяє більш гнучко і ефективно приймати рішення для вирішення локальних проблем і враховувати особливості окремих користувачів. Обмін досвідом між кінцевими пристроями прискорює навчання локальних систем машинного навчання. Для роботи з даними про взаємодію користувачів і продуктів використовуються рекомендаційні системи (Recommender Systems, RS). RS не тільки встановлює статистичні ознаки та взаємозв'язки між товарами, послугами та поведінкою користувачів, але й прогнозує переваги окремих користувачів на основі їхніх попередніх дій. Завдяки такому підходу можна надавати більш персоналізовані рекомендації

користувачам промислових систем, враховувати попит на існуючу продукцію та коригувати параметри виробництва для досягнення найбільшої ефективності.

Залучення кінцевих пристроїв для збору інформації покращує якість надання послуг і дозволяє створювати локальні моделі навчання, які враховують специфічні характеристики користувачів. Для аналізу даних у рекомендаційних системах використовуються спеціальні математичні алгоритми. Один із них, SVD, розглянуто в роботі та наведено його переваги для обробки розріджених матриць великого об'єму. Проблема обробки великих даних і надання відповідних рекомендацій користувачам в промислових системах залишається актуальною. В роботі проаналізовано алгоритм Funk SVD і виявили, що він ефективно використовується для обробки розріджених матриць великих даних. Однак жодні дослідження не показують його ефективності при роботі в умовах, коли навантаження на систему може динамічно змінюватися, тому оптимальні параметри обробки великих даних слід вибирати відповідно. Отже, на основі аналізу існуючих рішень запропоновано модифікації SVD, які дозволили прискорити процес обчислення, відкидаючи зайві дані про користувачів. Також, враховуючи високі вимоги до точності та швидкості обробки даних, запропоновано архітектуру рекомендаційної системи з обміном оптимальними параметрами розрахунку, що дозволяє швидше формувати рекомендації. На основі дослідження запропоновано архітектуру систем рекомендацій в IoT, що працюють зі змінними робочими вимогами. Таким чином, можна гнучко застосовувати запропоновані модифікації для досягнення максимальної ефективності обробки даних користувача.

Проаналізувавши останні роботи, визначено, що проблема гнучкої оптимізації великих даних у промислових системах та вибору найкращих параметрів роботи рекомендованих систем залишається актуальною. У роботі запропоновано поглибити дослідження, використовуючи при наданні рекомендацій дані про різну кількість користувачів. На основі досліджень встановлено, що результати можуть змінюватись залежно від обсягу та характеристик даних; тому доцільно вибирати найбільш оптимальні параметри

алгоритму Funk SVD для конкретних промислових систем. Також пропонується обмінюватися ефективними параметрами роботи системи рекомендацій між кількома користувачами для більш швидкої та ефективної обробки даних [12,13,14,15].

1.2. Ключові технології для оптимізації даних у розподілених системах

1.2.1. Особливості систем інтелектуального виробництва

Розвиток цифрових технологій дозволив трансформувати різні галузі та оцифрувати процеси, які раніше виконувалися лише вручну. Четверта промислова революція (Industry 4.0) передбачає функціонування виробництва у формі гнучкої та масштабованої інтелектуальної системи, яка може автоматично реагувати на події та навчатися самостійно. Розумне виробництво дозволяє інтегрувати нові функції в роботу промислової системи. Завдяки децентралізованій архітектурі використовується значна кількість розумних кінцевих і граничних пристроїв. Розподіл навантаження між ними підвищує надійність і конфіденційність даних. Ієрархія інтелектуальної виробничої системи показана на Рис.1.1.

Згідно з Рис.1.1, пристрої індустріальної системи виконують безпосередні функції виробництва товарів. За допомогою зв'язку між вузлами, обміну даними та аналізу система в цілому навчається надавати послуги користувачам. Постійний моніторинг стану системи визначає найбільш проблемні ланки виробничого процесу. Аналіз поведінки користувачів щодо продукту дозволяє встановити найбільш актуальні з них і модифікувати або припинити виробництво нерентабельних. Концепція Industry 4.0 передбачає перетворення централізованих промислових систем на гнучкі та незалежні. Для досягнення більш високої ефективності всі етапи виробництва розглядаються як сукупність взаємопов'язаних процесів, починаючи від виробництва товарів і закінчуючи взаємодією з споживачами. Інтеграція новітніх технологій, таких як машинне навчання, штучний інтелект, хмарні обчислення та Інтернет речей, трансформує промислові системи та дозволяє надавати нові послуги. Оскільки для аналізу стану системи використовується значна кількість кінцевих

пристроїв, великі дані постійно обробляються, тому їх якісна аналітика має вирішальне значення для ефективності виробництва.

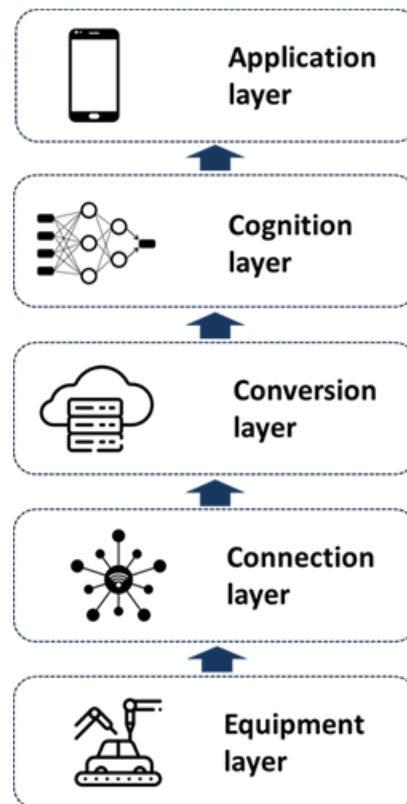


Рис.1.1.Ієрархія системи розумного виробництва.

Промислові системи тепер спрямовані не тільки на виготовлення певної продукції, але й на доставку до місця продажу, взаємодію з користувачем, аналіз його оцінок тощо. Значною мірою такий підхід дозволяє регулювати обсяги виробництва. Для певних продуктів, розробляє нові технології та пропонує послуги, які найбільш ймовірно будуть користуватися попитом. В умовах постійної конкуренції на ринку та змін у потребах користувачів обробка статистики про виробничу операцію знижує ймовірність виробництва неактуального товару та інвестування в проекти, які не знаходять відгуку у користувачів. Зворотній зв'язок може бути явним або прихованим:

- явні відгуки – оцінки, які користувачі залишають самостійно;
- Збір даних із різних джерел, таких як веб-сайти, соціальні мережі та пошукові системи, дозволяє нам отримувати неявні відгуки, які вказують на те, наскільки користувачі зацікавлені в певних продуктах або послугах [16,17,18].

Оскільки кількість даних, що підлягають обробці, постійно збільшується, використовуються системи рекомендацій, які визначають відносини між

користувачами та виробничими службами. Прикладами роботи є рекомендації відеороликів на YouTube, фільмів на Netflix, додатків у Play Market тощо. Таким чином користувачі отримують рекомендації щодо нових продуктів чи послуг, які найбільше їх зацікавлять. Згідно зі статистикою, значна частина завантажень в Play Market була зроблена завдяки рекомендаціям. Системи рекомендацій полегшують процес вибору товарів і пошуку найбільш підходящих для окремих категорій користувачів. Це також можливість для виробників RS ефективніше пропонувати свою продукцію [19,20,21,22].

Процес пошуку найбільш підходящих товарів серед користувачів проходить у кілька етапів:

- вибір серед набору найбільш цікавих для користувача товарів;
- оцінка обраних товарів;
- сортування за рейтингом;
- надання рекомендацій користувачам;
- корекція результатів.

Пошук найбільш підходящих продуктів для користувачів може здійснюватися декількома способами, найпоширенішими є фільтрація на основі продуктів і спільна фільтрація. Фільтрування на основі продукту знаходить нові продукти, найбільш схожі на ті, що раніше подобалися окремому користувачеві. При цьому дані інших користувачів не враховуються. Наприклад, якщо користувачі віддають перевагу одному жанру фільмів, то вибираються схожі, які можуть їм сподобатися. Переваги фільтрації на основі контенту:

- врахування характеристик користувача при формуванні рекомендацій;

- необхідно обробляти дані тільки одного користувача.

Недоліки цього підходу:

- неможливість надання товарних рекомендацій, що виходять за межі звичних переваг;

- значні витрати ресурсів

Іншим підходом є спільне фільтрування даних, при якому спочатку визначається певний ступінь схожості між користувачами. Потім товари, які вже були позитивно оцінені одним або кількома з них, рекомендуються серед схожих користувачів [23,24,25,26,27].. Завдяки такому підходу ми можемо розширити перелік рекомендованих послуг. Перевагами спільної фільтрації є:

- навіть за відсутності даних про переваги користувача можна надати рекомендації;
- більш різноманітні пропозиції товарів;
- простота реалізації.

Недоліки методу спільної фільтрації:

- Проблема холодного запуску, яка виникає, коли нові елементи в системі не обробляються відразу і не підпадають під рекомендації для відповідної категорії. Вирішується шляхом автоматичного додавання одного елемента до категорії без повторного навчання всієї системи;
- труднощі з урахуванням додаткових факторів, які можуть вплинути на точність рекомендацій. Наприклад, при виборі продукту такими факторами можуть бути вік користувача, країна проживання та ін. Для підвищення продуктивності систем спільної фільтрації вводяться додаткові функції розрахунку параметрів [28,29,30,31]..

Щоб підвищити ефективність рекомендацій, обмінюються даними про результати різних користувачів. Загальна архітектура системи рекомендацій показана на Рис.1.2. Визначення найбільш цікавих для користувачів продуктів допомагає скоригувати параметри виготовлення. Відомо, що персоналізовані рекомендації, які користувачі можуть отримувати в соціальних мережах, електронною поштою та на веб-сайтах на основі попередніх дій, полегшують пошук найбільш підходящої служби серед багатьох послуг. Важливою проблемою для систем рекомендацій є безпека та конфіденційність інформації користувача. Використання даних з персональних пристроїв має бути захищене від стороннього втручання та модифікації [31,32,33,34].

Слід також враховувати можливість неправильних оцінок від деяких користувачів в результаті помилки або втручання зловмисників. Для вирішення

проблеми необхідно постійно контролювати аутентифікацію користувачів і перевірку даних, що надходять від них.

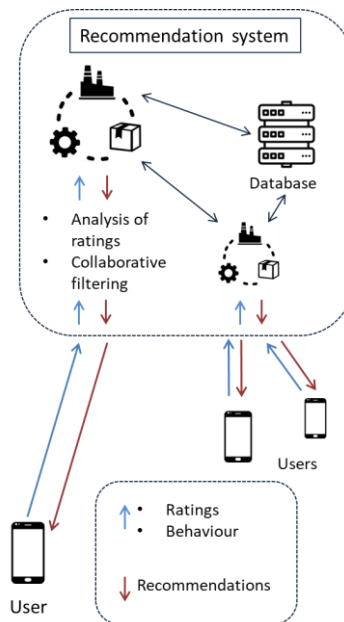


Рис.1.2. Архітектура рекомендаційної системи.

На основі оцінок користувачів промислової системи можна встановити перспективи виробництва певного виду продукції або надання певних послуг. Якщо спостерігається зниження інтересу до продукту, його можна модифікувати або замінити на новий, який більше відповідає потребам споживачів [35,36,37,38,39].

1.3. Рекомендаційні системи як засіб для обробки даних

Обробка даних в інформаційних системах є актуальним завданням сучасності. Оскільки впроваджується величезна кількість технологій, які спрощують виконання складних функцій замість людини, інформація з різних джерел постійно збирається та обробляється. Системи електронної комерції, розумні міста та виробництво давно стали звичним явищем. Такі системи є складними за структурою і містять велику кількість компонентів, які взаємодіють між собою та обмінюються даними. На основі алгоритмів аналізу цих даних і прийняття рішень інформаційна система може самостійно вирішувати багато завдань, підвищуючи тим самим свою продуктивність.

Для ефективної обробки великих даних використовуються методи машинного навчання, які дозволяють значно прискорити та спростити обчислення. Прикладами машинного навчання є класифікація зображень, аналіз

тексту, розпізнавання мовлення тощо. Такі проблеми набагато ефективніше вирішуються за допомогою програмних алгоритмів, особливо з великими обсягами даних. Еволюція технологій машинного навчання змінила багато галузей. У промисловості програмне керування величезною кількістю підсистем дозволяє швидко виявляти, усувати та прогнозувати нові потенційні проблеми [40,41,42,43].

Завдяки розробці нових і вдосконаленню існуючих алгоритмів машинного навчання змінилися й комерційні інформаційні системи. Оскільки користувачі замовляють багато товарів і послуг онлайн, вони проводять багато часу на веб-сторінках і в соціальних мережах, виконуючи певні дії. Їх можна придбати, поставити лайки, відгуки та оцінити. Тому виникає необхідність обробити та проаналізувати всі ці дані, знайти закономірності в поведінці користувачів. Рекомендаційні системи (RS), які є прикладом машинного навчання, визначають відносини між користувачами та продуктами та дозволяють передбачити їхні майбутні можливі переваги. Таким чином, підвищується рівень якості надання послуг.

Алгоритми машинного навчання допомагають покращити швидкість і релевантність рекомендацій. Оскільки дані в рекомендаційних системах надходять з різних джерел і часто різного типу, їх попередня обробка необхідна для швидкого подальшого аналізу. Крім того, неважливу інформацію, яка не впливає на точність алгоритмів рекомендацій, слід ідентифікувати та відкинути. В іншому випадку надлишок даних спричиняє перевантаження обчислювальних пристроїв. Для роботи рекомендаційних систем формуються спеціальні рекомендаційні матриці, елементами яких є рейтинги та оцінки різних товарів користувачами. Такі матриці дуже зручні, оскільки демонструють зв'язки між елементами системи [44,45,46].

У роботі розглянуто особливості алгоритму Fed SVD, який дозволяє обробляти великі дані в рекомендаційних системах. Модифікований Funk SVD для створення рекомендацій на основі меншої кількості даних користувачів і продуктів. Також додатково використовувався метод спільної фільтрації на основі сусідства, який визначає найважливіші дані для подальших досліджень,

що підвищує точність обчислень. Запропоноване спільне використання Funk SVD і розподіленого алгоритму Fed SVD виявилось особливо ефективним. Такий підхід дозволяє організувати обробку даних користувача безпосередньо на кінцевих пристроях, а потім використовувати результати для навчання глобальної моделі. Таким чином забезпечується висока точність і надійність обчислень, зменшується їх тривалість і навантаження на обчислювальні пристрої.

1.3.1. Методи машинного навчання

У сучасному світі ми є свідками стрімкого впровадження інформаційних технологій у різні сфери. Повсякденне життя змінюється, оскільки багато функцій, які раніше виконувалися людьми, тепер автоматизовані. Різноманітні соціальні мережі, додатки та гаджети становлять значну частину існування не лише окремих людей, а й цілих промислових корпорацій. У результаті багато взаємопов'язаних процесів часто мають глобальний вплив. Інтеграція новітніх технологій у соціальні сфери вимагає розробки нових інструментів для регуляризації взаємодії між людиною та машиною. Забезпечення безпеки та конфіденційності інформації користувача є надзвичайно важливим, оскільки вона може бути використана для покращення продуктивності системи або потрапити в руки зловмисників. Для вирішення таких проблем використовуються спеціальні методи захисту важливих даних, захисту каналів зв'язку від несанкціонованого зовнішнього втручання та аутентифікації користувача. Оскільки більшість сучасних інформаційних систем мають розподілену архітектуру, важливо забезпечити доступ до всіх сервісів користувачам з максимальною швидкістю та якістю [47,48,49].

Машинне навчання дозволяє нам швидко й з високою точністю обробляти великі обсяги даних. Таким чином, вирішуються завдання пошуку необхідних елементів, визначення статистичних особливостей інформації та прогнозується наступні ймовірні події. Машинне навчання — це підрозділ штучного інтелекту, який використовує різні методи та засоби аналізу даних для самостійного виконання певних операцій над ними. На відміну від звичайних програмних алгоритмів, машинне навчання працює більш незалежно

від людини. Різноманітність методів машинного навчання дозволяє визначити найкращий підхід до вирішення проблем. Алгоритми навчання під наглядом отримують спеціальні мітки про те, яким має бути результат. Таким чином, можна навчити програму швидше приймати рішення про наявність певних ознак у об'єкта, за яким вона класифікується [51,52,53,54].

Основні завдання навчання під керівництвом:

- регресія, тобто знаходження рівняння, яке описує зв'язок між змінними і константами в моделі;
- класифікація — це поділ об'єктів на різні групи за певними ознаками.

Ще одним надзвичайно популярним методом машинного навчання є навчання без нагляду. Цей підхід передбачає незалежну обробку даних для алгоритму навчання. Результат може бути повільнішим, але для певних завдань навчання без нагляду показує хороший результат. Приклади навчання без контролю:

- кластеризація, коли дані розбиваються на групи за певною ознакою. Кластеризація відрізняється від класифікації тим, що алгоритм не знає ознак заздалегідь, а визначає їх у процесі навчання;
- зменшення розмірності даних, тобто групування набору ознак в одну загальну, таким чином зменшуючи їх кількість;
- асоціація, пошук відповідностей між даними.

Контрольовані та неконтрольовані алгоритми навчання належать до так званих класичних методів машинного навчання. Їх перевагою є відносна простота й ефективність обробки даних. Більше методів машинного навчання зображено на Рис.1.3.

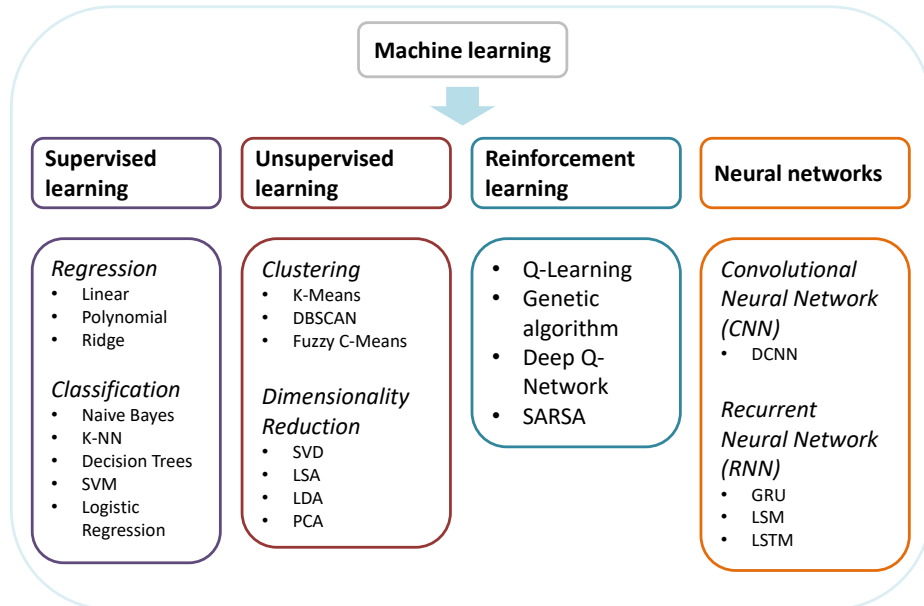


Рис.1.3. Методи машинного навчання

Навчання з підкріпленням використовується для навчання в певному середовищі, якщо у нас недостатньо даних для використання класичних алгоритмів. Тоді методом проб і помилок при виконанні певного завдання можна визначити оптимальний шлях його вирішення. Іншим методом машинного навчання є нейронні мережі (Neural Networks, NN), де створюється багатошарова модель зв'язків між простими елементами (нейронами). NN широко використовуються для розпізнавання зображень, розпізнавання мови тощо.

Проблеми обробки великих даних зараз дуже актуальні для багатьох галузей. Наприклад, системи промислового Інтернету речей складаються з великої кількості кінцевих пристроїв, які виконують різні функції для отримання загального результату. Необхідно постійно збирати дані про стан цих пристроїв, навколишнє середовище, особливості поведінки користувачів, попит на товари. Усі ці показники потрібно швидко та безпечно зібрати, а також попередньо обробити, тобто виявити неважливі та повторювані дані та видалити їх. Такий підхід дозволяє знизити навантаження на обчислювальні пристрої та отримати більш швидкий результат. Потім дані аналізуються за допомогою методів штучного інтелекту та машинного навчання для встановлення закономірностей і статистичних особливостей.

На основі отриманих залежностей можна швидко виявити можливі несправності та аварійні ситуації, нераціональний розподіл ресурсів, слабкі ланки виробництва. Прикладом інтелектуальної промисловості є розумні електромережі, охорона здоров'я та комерційні системи. Гнучкість розподілу електроенергії в системах smart grid сприяє її раціональному використанню в залежності від індивідуальних особливостей кожного користувача. Визначення годин пікового навантаження дозволяє розрахувати системні ресурси для ефективного обслуговування. У комерційних інформаційних системах аналітика великих даних дозволяє нам надавати клієнтам персоналізовані пропозиції та автоматично обробляти відгуки, таким чином підвищуючи рівень надання послуг [55,56,57].

Проблема систем електронної комерції зараз полягає в наступному:

- велика кількість даних, отриманих від користувачів (статистика покупок, відвідування сайтів і соціальних мереж);
- необхідність формування швидкої рекомендації найбільш підходящих товарів у короткий термін;
- обмеженість обчислювальних ресурсів системи;
- розподілена архітектура системи та велика кількість кінцевих пристроїв з різною функціональністю та правами, що вимагає надійності та конфіденційності при обробці даних.

Системи рекомендацій працюють у кілька етапів:

- збір даних, що означає збір оцінок продуктів від різних користувачів та їх взаємодію з контентом на різних платформах (сайти, програми, соціальні мережі). На цьому етапі важливо забезпечити збір даних, які найбільш правильно відображають поведінку різних користувачів щодо різних предметів;
- оптимізація даних, тобто визначення найбільш важливих для подальшої обробки. Видалення неважливої та зайвої інформації дозволяє нам прискорити обчислення, часто зберігаючи досить високу точність;
- збереження інформації про користувачів у базах даних є важливим для її подальшого використання при наданні рекомендацій;

- аналіз даних і надання рекомендацій за допомогою методів машинного навчання та надання їх відповідним користувачам.

Існує багато способів визначити рекомендації продукту чи послуги для конкретного користувача. в тому числі:

- аналіз попередніх дій користувача;
- визначення схожості між товарами;
- виявлення схожості між користувачами та рекомендація їм схожих послуг.

Одним із найпопулярніших застосувань машинного навчання для обробки даних користувачів є RS (системи рекомендацій або системи рекомендацій), які призначені для пошуку відповідних продуктів або послуг. RS використовується Netflix, Youtube і Amazon для надання персоналізованих рекомендацій користувачам системи щодо нових продуктів. На основі аналізу попередніх дій користувачів можна встановити відповідність між цікавими сервісами (Рис.1.4).

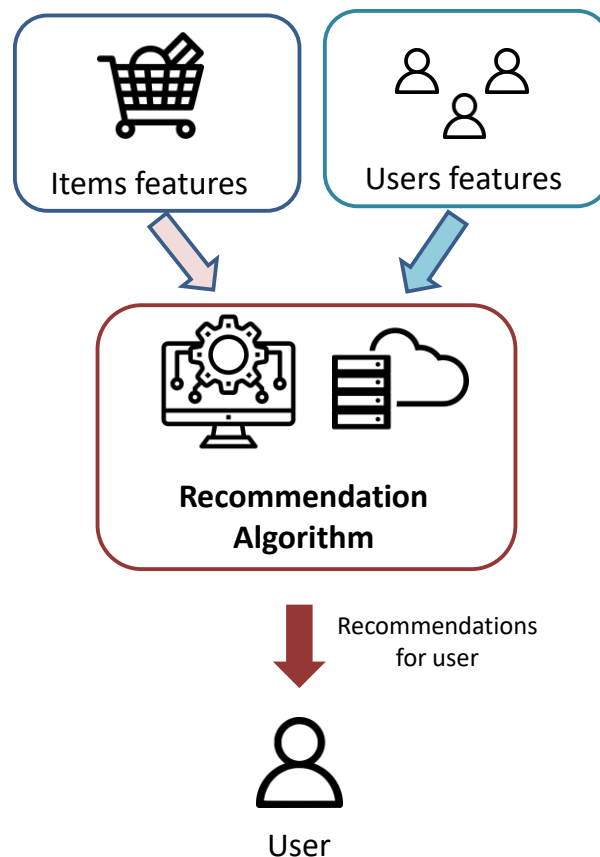


Рис.1.4. Процес визначення рекомендацій

На основі цієї класифікації виділяють різні технології пошуку рекомендацій. Фільтрування на основі вмісту передбачає ідентифікацію

потенційно цікавих продуктів лише на основі попередніх дій користувача. Таким чином, алгоритм розраховує відмінності за певними параметрами для попередніх і нових продуктів і пропонує найбільш схожі. Контентний підхід досить простий і ефективний, якщо система рекомендацій має інформацію про дії користувача. Однак новим користувачам важко вибрати продукти в системі. Нові продукти, які відрізняються від інших продуктів, також враховуються, тому важко забезпечити різноманітність рекомендацій. Користувач, швидше за все, отримає інформацію, дуже схожу на його попередні запити, і не матиме пропозицій щодо інших. У цьому випадку є ризик, що рекомендації з часом втратять актуальність,

Спільна фільтрація (Collaborative Filtering, CF) використовується для вирішення проблем фільтрації на основі вмісту. У цьому випадку розрізняють CF на основі пам'яті та на основі моделі. CF на основі пам'яті підрозділяється на:

- CF на основі користувача, коли ідентифікуються користувачі зі схожою поведінкою, а потім їм рекомендуються схожі продукти;
- на основі предметів, пошук продуктів, які отримали подібні оцінки від різних користувачів.

Переваги підходу на основі пам'яті полягають у тому, що рішення для "холодного запуску" проблема, коли система рекомендацій ще не має достатньо інформації про користувачів і продукти. Для покращеного розрахунку рекомендацій використовується CF на основі моделі, яка формує матрицю взаємодії між користувачами та продуктами. За допомогою алгоритмів машинного навчання ми можемо знайти закономірності та подібності між даними та визначити, які продукти зацікавлять користувачів у майбутньому. Важливим прикладом модельного підходу є матрична факторизація.

Матрична факторизація (MF) передбачає перетворення (декомпозицію) матриці взаємодії користувачів і товарів або послуг (елементів загальної назви) на помножені матриці менших розмірів. Кожна така матриця відповідає за індивідуальні характеристики (користувачів або продуктів). Таким чином, зручніше виявити закономірності в уподобаннях різних користувачів,

розрахувати незаповнені клітинки рейтингової матриці та задати найважливіші дані для подальших розрахунків.

1.4. Перспективи переходу до методів інтелектуальної обробки даних великих обсягів

Різноманітність технологій формування рекомендацій дозволяє використовувати найбільш оптимальну з них при вирішенні конкретних завдань. Часто використовуються гібридні рекомендаційні системи, які поєднують різні методи аналізу даних, розглянуті вище. Залежно від кількості вихідних даних про користувачів і продукти, масштабу системи та вимог до її функціонування змінюються підходи до здійснення розрахунків. З розвитком технологій штучного інтелекту (ШІ) використання систем рекомендацій тепер дозволяє покращувати звичайні послуги. Наприклад, пошукова система Google на основі штучного інтелекту розробляє алгоритми рекомендацій, які дозволяють користувачам швидше знаходити потрібну інформацію, а також передбачити потенційно цікавий контент. Чим точніше і швидше система рекомендацій рекомендує відповідні товари, тим більша ймовірність того, що покупець з нею взаємодіє. Якщо рекомендація не була зроблена вчасно, клієнт може просто покинути сайт або додаток, не зробивши покупки [57,58].

Як згадувалося раніше, одним із прикладів неконтрольованого навчання є зменшення розмірності даних. Це означає групування ознак за певною властивістю та призначення їм нової ознаки. Наприклад, ми не можемо розглянути всі ознаки та параметри автомобіля, але встановити для нього певну характеристику, за якою будемо формувати рекомендації (марка автомобіля, об'єм двигуна, рік випуску тощо). Звичайно, такий підхід передбачає видалення частини даних під час обчислень, що може негативно вплинути на їх точність. Однак досвід різних досліджень показує, що надлишок даних не завжди критичний для надання успішних рекомендацій. З іншого боку, велика кількість ознак, якими оперує алгоритм навчання, ускладнює визначення основних, використовує багато ресурсів і уповільнює обчислення. Важливим показником якості рекомендаційних систем є швидкість надання результатів. Якщо користувачі не отримують рекомендації щодо нового продукту чи послуги,

поки вони ще користуються програмою чи веб-сайтом, через певний період вони втратять інтерес. Компроміс між точністю та швидкістю обчислення даних користувача в системах рекомендацій є важливим питанням. Тому методи машинного навчання використовуються для виявлення найбільш цінних даних для подальшої обробки та швидкого надання результату.

Популярними алгоритмами для зменшення розмірності великих даних є SVD (Singular value decomposition), PCA (Principal component analysis) і LSA (Latent semantic analysis). Усі ці алгоритми призначені для виявлення прихованих ознак у масивах даних та їх узагальнення. Декомпозиція, тобто перетворення вихідної матриці даних у набір підматриць, дозволяє розділяти характеристики різних об'єктів і ефективно їх обробляти. Тому, визначивши, які ознаки є важливими при вирішенні конкретного завдання, можна використати менше даних, але зберегти відносно високу точність обчислень. Принцип роботи алгоритму SVD показаний на Рис.1.5.

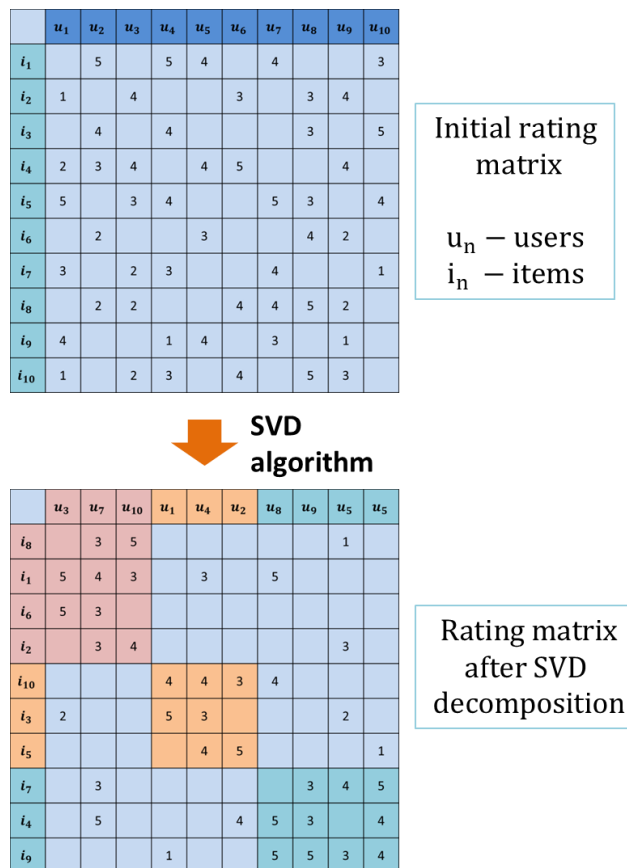


Рис.1.5. Алгоритм сингулярної декомпозиції (SVD)

Алгоритм SVD використовується найчастіше, оскільки він демонструє високу ефективність у виявленні прихованих зв'язків між даними та дозволяє зменшити кількість даних, необхідних для розрахунку рекомендацій. SVD знаходить власні вектори і власні значення для початкового набору даних і формує з нього добуток трьох матриць. За характеристики користувача і продукту відповідають дві матриці. Елементи третьої матриці містять зв'язки між товарами та користувачами. Отже, простіше і зрозуміліше визначити важливість тих чи інших елементів вихідної матриці [59,60].

Висновки до першого розділу.

У розділі було проаналізовано тенденції розвитку оброблення великих даних, визначено методи та засоби для їх оптимізації в розподілених системах. Досліджено особливості роботи рекомендаційних систем для знаходження закономірностей в масивах інформації. Визначено особливості оброблення даних в промислових та комерційних інформаційних системах. Розглянуто методи машинного навчання для прискорення та автоматизації аналізу інформації. Результатом розділу є розв'язання задачі № 1 дослідження.

РОЗДІЛ 2. ДОСЛІДЖЕННЯ МЕТОДІВ ТА АЛГОРИТМІВ АВТОМАТИЧНОЇ ОБРОБКИ ВЕЛИКИХ ДАНИХ

У розділі розглянуто підходи до покращення ефективності оброблення даних користувачів. Досліджено актуальні методи в даній галузі та визначено основні задачі, які потребують вирішення.

Проаналізовано ефективність використання алгоритму сингулярної декомпозиції та особливості його роботи в розподілених системах. Досліджено алгоритм федеративного сингулярного розкладу даних. Визначено методи машинного навчання для оброблення даних в розподілених системах.

Результати розділу опубліковано в працях автора [2,4,6,7,8,9,10,11].

2.1. Ефективність використання алгоритму сингулярної декомпозиції

Для потреб великих компаній, які займаються торгівлею, послугами та іншою суміжною діяльністю, важливою проблемою є постійний збір інформації про характеристики покупців та їх уподобання щодо різних товарів чи послуг.

Невеликі компанії можуть збирати інформацію без використання складних обчислювальних ресурсів. Це можна зробити за допомогою простих обчислювальних пристроїв, які не потребують особливих складнощів в обслуговуванні. Однак для великих компаній існує необхідність аналізувати дані багатьох користувачів. Регіональні, національні чи міжнародні компанії працюють з даними мільйонів людей [61,62,63].

У цьому випадку доцільно застосувати інші підходи до організації комп'ютерної системи. Навіть із збільшенням потужності обчислювального пристрою, що обробляє дані користувача, зростає ризик збою через велике навантаження.

Для ефективного аналізу даних необхідно використовувати досить складні алгоритми, здатні обробляти інформацію про користувача і відкидати несуттєве. Наприклад, якщо користувач інтернет-магазину замовив усі речі з осінньої колекції, то немає необхідності зберігати дані про кожну вподобану річ окремо, щоб потім зробити висновок, що користувачеві магазину подобається

Таким чином, системи рекомендацій можуть вивчати отриману інформацію та заповнювати пропуски в таблиці переваг. Однак для систем з мільйонами користувачів необхідно використовувати спеціальні методи аналізу інформації. Одним із них є метод декомпозиції сингулярного значення (SVD), який, на відміну від інших методів, простий у використанні та забезпечує менший час обробки даних.

Проте зараз все більше комерційних компаній переходять на онлайн-режим. Це створює ще більше навантаження на реферальні системи. Якщо робота алгоритму обробки даних підтримується тільки сервером вузла, то рано чи пізно він не зможе обробляти дані в реальному часі, і можуть виникнути затримки в наданні рекомендацій. Також можливий збій обчислювального пристрою, і якщо він один обробляє всі дані, система повністю виходить з ладу.

Для підвищення ефективності пропонується використовувати розподілені системи обробки даних. У них усі дані, які потребують обробки, поділяються на менші частини й обробляються одночасно на різних обчислювальних пристроях. У цьому випадку скорочується час обробки даних і знижується ризик збоїв системи.

Для функціонування розподілених систем доцільно використовувати різні технології розподіленої обробки даних. Найпростішою є технологія MPI (Message Passing Interface). У ньому дані просто розподіляються центральним вузлом на рівні частин і відправляються на робочі вузли для обробки. Це передається між вузлами системи за допомогою спеціального протоколу обміну повідомленнями. Зі збільшенням обсягу даних у системі службові повідомлення набувають значних обсягів, що негативно позначається на продуктивності обчислень. Більш ефективною є технологія Nadoop, в якій немає таких службових повідомлень, дані просто надсилаються на робочі пристрої, записуються в його енергонезалежну пам'ять, обробляються і відправляються на центральний вузол. Після обробки дані видаляються з пам'яті пристрою [69,70,71]. Саме процес запису та видалення даних з енергонезалежної пам'яті негативно впливає на час обчислень, особливо з великими обсягами даних. Технологія Spark вирішує цю проблему, оскільки

дані записуються не в енергонезалежну пам'ять, а в оперативну пам'ять (RAM), що прискорює процес обчислень. Також варто відзначити таке поняття, як розподілені агенти. Це інтелектуальні пристрої в розподіленій системі, які здатні обмінюватися повідомленнями один з одним і визначати оптимальне навантаження на кожен вузол.

Зі збільшенням обсягу даних швидкість надання користувачам якісних рекомендацій серед великих обсягів інформації стала серйозною проблемою. З поширенням використання послуг мобільного зв'язку швидкість відповіді на запит також стала важливим показником якості обслуговування користувачів. На сьогоднішній день в системах рекомендацій, які використовують Інтернет як шину зв'язку, існує технічна проблема з гарантованою якістю послуг за критерієм часу відгуку послуги. Оскільки загальний час відповіді для наданої послуги складається з часу обробки запитів на серверах додатків і затримки мережі, створеної комутаторами та маршрутизаторами, мобільними шлюзами, завдання надання запитаного часу відповіді послуги вимагає належного управління як комунікаційними, так і обчислювальними ресурсами [72,73,74].

2.2. Аналіз роботи алгоритму сингулярної декомпозиції в розподілених системах

2.2.1. Дослідження та моделювання методу спільної фільтрації даних

Щоб ефективніше обробляти дані та давати рекомендації користувачам, недостатньо використовувати модель, яка просто враховує, що певному користувачеві сподобався певний предмет. Слід переходити до більш точного опису, в якому враховуються загальні тенденції для певних категорій користувачів цікавитися будь-якими предметами.

Це також усуває проблему занадто великої кількості даних. Замість використання великої кількості векторів оцінки продукту для оцінки переваг користувачів використовується набагато компактніший вектор переваг.

У цьому розділі буде обговорено метод сингулярної декомпозиції даних, який використовується для зменшення розмірності даних. Даний метод також використовується в задачах пошуку прихованих характеристик користувачів системи та об'єктів інтересу [75,76,77].

При сингулярному розкладі рейтингова матриця R ($m \times n$) може бути розкладена на добуток трьох матриць: U ($m \times k$), Σ ($k \times k$) і V ($k \times n$) (k змінюється залежно від ступінь деталізації розкладу). Матриці U і V відповідають за приховані характеристики користувачів і об'єктів (вік, регіон проживання, ціна і т.д.).

$$A = U \cdot D \cdot V^T \tag{2.1}$$

Згідно формули 1, матриці U та V містять ортогональні стовпці, які є власними векторами (для ненульових значень) для $A \cdot A^T$ і $A^T \cdot A$ матриць.

Слід також мати на увазі, що:

$$U^T \cdot U = V^T \cdot V = I \tag{2.2}$$

Де I — одинична матриця, в якій елементи, розташовані на головній діагоналі, рівні один одному, а всі інші — нулю.

D є матрицею розміру (d, d) :

$$D = \begin{pmatrix} a_1 & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & a_d \end{pmatrix} \tag{2.3}$$

є діагональною матрицею, що містить позитивні елементи вздовж головної діагоналі. Ці елементи впорядковані у порядку спадання:

$$a_1 \geq a_2 \geq \dots \geq a_d \geq \tag{2.4}$$

SVD є розширенням вихідних даних в системі координат, де матриця коваріації є діагональною. Операція методу декомпозиції сингулярних значень показана на Рис. 2.2.

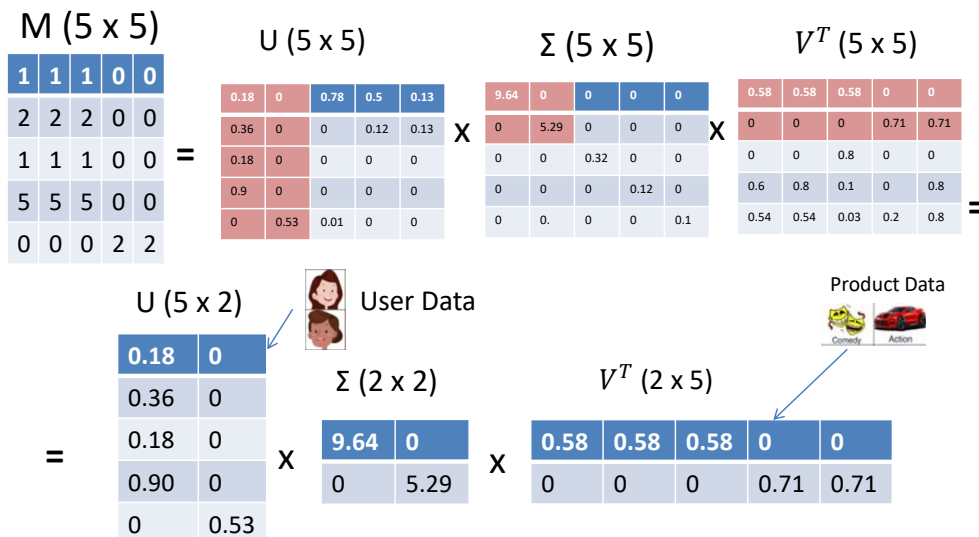


Рис.2.2. Метод SVD

Початкова матриця M ділиться на добуток ортогональних матриць U і V і діагональної матриці Σ .

Можна виділити певну k максимальну кількість елементів діагоналі матриці Σ і відкинути інші рядки і стовпці. У матрицями U і V , відповідно, ми також залишимо лише k рядків. Після множення нових підматриць отримуємо матрицю даних, яка з високою точністю повторює характеристики початкової M , але має меншу розмірність, оскільки не враховує зайву і непотрібну інформацію.

Робота методу SVD складається з знаходження власних значень і власних векторів (позначається $A \cdot A^T$ і $A^T \cdot A$) [78,79]. Власні вектори $A^T \cdot A$ відповідають стовпчикам матриці V . Вектори $A \cdot A^T$ відповідають стовпчикам матриці U .

Елементи матриці S відповідають квадратним кореням власних значень $A \cdot A^T$ або векторів або $A^T \cdot A$.

Елементи матриці S впорядковані в порядку спадання. Вони завжди належать до дійсних чисел.

2.2.2. Застосування сингулярної декомпозиції значень в рекомендаційних системах

Щоб застосувати сингулярну декомпозицію в рекомендаційних системах, ми спочатку шукаємо користувача, який найбільше нагадує поточний, використовуючи матрицю U , що має зменшену розмірність. Для такого пошуку застосовуються спеціальні алгоритми знаходження подібності між елементами.

Потім надається необхідна оцінка. Для цього ми використовуємо матрицю елементів. Він шукає елементи, найбільш схожі на ті, які раніше оцінювали цей користувач. Оцінка цього елемента може бути використана для передбачення [80,81,82].

Рядки і стовпчики матриць називаються векторами. Рядок матриці U відповідає користувачам системи, стовпчик матриці V містить елементи, які оцінюються. Якщо вектори користувача схожі, то можна припустити, що користувачі мають подібні вподобання. У методі декомпозиції сингулярних значень найбільше уваги приділяється пошуку подібних користувачів або

елементів. Для обчислення подібності використовуються такі методи як: Знаходження евклідової відстані між елементами, обчислення коефіцієнта кореляції Пірсона тощо

2.2.3. Стандартний метод сингулярної декомпозиції

Для реалізації стандартного методу сингулярного розкладання матриць слід виконати такі кроки:

1. Для початкової матриці A шукається її QR-розклад;
2. Представляємо матрицю рекомендацій RT у вигляді добутку трьох матриць: $L^T \cdot D_2 \cdot R^T$. Варто відзначити, що матриця D_2 є дводіагональною матрицею;
3. Представляємо матрицю D_2 як добуток трьох матриць: $L^T \cdot D_1 \cdot R^T$ (Матриця D_1 має діагональний вигляд).

(1) Пошук QR-розкладу

Для знаходження розкладання вхідної матриці її представлення виконується як добуток матриці Q , що ортогональна, і матриці R (яка є верхньою трикутною).

Ці операції можуть бути виконані за допомогою методу обертання Гівенса.

Здійснюється послідовне перетворення всіх елементів, розташованих під головною діагоналлю матриці до нуля. Для цього ми множимо матрицю на певну матрицю обертання (позначається як G):

$$A = G_k \cdot G_{k-1} \cdot \dots \cdot G_1 \cdot R, \quad (2.5)$$

$$Q = G_k \cdot G_{k-1} \cdot \dots \cdot G_1, \quad (2.6)$$

$$A = Q \cdot R. \quad (2.7)$$

(2) Приведення матриці RT до дводіагональної форми

Матриця R є верхньою трикутною, відповідно R^T , є нижньою трикутною. Відбувається поступове обнулення елементів під головною діагоналлю (спочатку, нижні стовпці матриці R^T). Це можна зробити множенням R^T на ліву матрицю поворотів Гівенса [37].

Слід зазначити, що в кожній ітерації методу елемент матриці $r(i, j)$ може бути ненульовим. У випадках справедливості умови $j \neq (i + 1)$ ми повинні

помножити матрицю R^T на праву матрицю обертання Гівенса праворуч, щоб даний елемент дорівнював нулю.

Тому тільки елементи матриці, які розташовані на головній діагоналі і безпосередньо над нею не є нульовими:

$$\begin{array}{cccccc} x & 0 & 0 & x & Z & 0 \\ x & x & 0 & \Rightarrow 0 & x & Z. \\ x & x & 0 & 0 & 0 & x \end{array} \quad (2.8)$$

Результатом таких операцій є матриця D_2 (яка характеризується подвійною діагоналізністю):

$$D_2 = L_1 \cdot R^T \cdot R_1. \quad (2.9)$$

(3) Приведення матриці D_2 до діагональної форми

Щоб зробити його діагональним, ми поступово будемо присвоювати нульове значення елементам, які не знаходяться на головній діагоналі матриці (допоки хоча б один елемент більший за встановлене значення) [83,84,85].

При виконанні операцій з елементами нижче головної діагоналі матриця множиться на ліву матрицю обертання. При обробці елементів, розташованих над головною діагоналлю, виконується множення на праву матрицю обертання.

Після ітеративної обробки матриці D_2 вона поступово зводиться до діагональної форми. Тоді отримана діагональна матриця D_1 :

$$D_1 = L_2 \cdot D_2 \cdot R_2. \quad (2.10)$$

В результаті отримуємо:

$$R^T = L_1^T \cdot D_2 \cdot R_1^T, \quad (2.11)$$

$$D_2 = L_2^T \cdot D_1 \cdot R_2^T, \quad (2.12)$$

$$R_2^T = L_1^T \cdot L_2^T \cdot D_1 \cdot R_2^T \cdot R_1^T, \quad (2.13)$$

$$R = R_1 \cdot R_2 \cdot D_1^T \cdot L_1, \quad (2.14)$$

$$R = R_1 \cdot R_2 \cdot D_1 \cdot L_1 \cdot L_2, \quad (2.15)$$

$$A = Q \cdot R_1 \cdot R_2 \cdot D_1 \cdot L_2 \cdot L_1, \quad (2.16)$$

$$U = Q \cdot R_1 \cdot R_2, \quad (2.17)$$

$$\Sigma = D_1, \quad (2.18)$$

$$V = L_2 \cdot L_1, \quad (2.19)$$

$$A = U \cdot \Sigma V^T. \quad (2.20)$$

A. Розклад матриці QR

QR -розклад — це перетворення вхідної матриці з розмірністю $(n \times m)$ в добуток матриць: Q , розмірність $(n \times n)$, яка ортогональна, і матрицю $(n \times m)$, яка є верхньою трикутною.

Існує багато методів отримання QR -розкладу. Серед найпоширеніших є метод обертання (його ще називають методом Гівенса).

Метод обертання поступово приводить матрицю, що знаходилася на вході, до верхньої трикутної форми. Метод працює так: На кожному етапі для матриці A_h обчислюється певна матриця повороту Q_h , щоб зробити елемент a_{ij} з матриці A_{h+1} ($A_{h+1} = Q_h \cdot A_h$) рівним нулю.

Обертання Гівенса називається перетворенням (лінійністю) векторної площини. Таке перетворення відповідає обертанню двох координатних осей у площині на певний кут.

Щоб скинути певний елемент матриці, необхідно обчислити матрицю обертання, відповідну початковій матриці, тоді ці дві матриці множаться.

Матриця повороту:

$$G(i, k, \varphi) = \begin{pmatrix} 1 & & & & & & & \\ & \ddots & & & & & & \\ & & c & & s & & & \\ & & \vdots & \ddots & \vdots & \vdots & \vdots & \\ & & -s & & c & & & \\ & & & & & \ddots & & \\ & & & & & & 0 & \\ & & & & & & & \ddots \\ & & & & & & & & 1 \end{pmatrix}, \quad (2.21)$$

Де $s = \sin(\varphi)$, $c = \cos(\varphi)$, φ — кут повороту матриці.

Наприклад, слід знайти QR -розклад матриці такого роду:

$$A = \begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix}. \quad (2.22)$$

Для цього ми використовуємо метод обертання.

Процесуватимемо елементи матриці поступово. Спочатку вибираємо лівий стовпчик, а елементи в ньому обчислюються зверху вниз. Далі ми працюємо з другою колоною ліворуч і т.

На першому етапі обчислюємо матрицю обертання елемента c_1 .

Матриця обертання G_{c_1} :

$$G_{c_1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c & s \\ 0 & -s & c \end{pmatrix} \quad (2.23)$$

Не будемо явно обчислювати кут φ , а просто визначимо його за формулами:

$$\gamma = \sqrt{b_1^2 + c_2^2}, \quad (2.24)$$

$$c = \frac{b_1}{\gamma}, \quad (2.25)$$

$$s = \frac{c_1}{\gamma}. \quad (2.26)$$

Після множення матриць G_{c_1} змінюються тільки рядки b_1 і c_1 , що відповідають індексам рядків, де вони розташовані:

$$A_{c_1} = G_{c_1} \cdot A = \begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ 0 & c_2 & c_3 \end{pmatrix}. \quad (2.27)$$

Виконаємо ті ж перетворення для елементів b_1 і c_2 за аналогією:

$$A_{c_2} = G_{c_2} \cdot (G_{c_1} \cdot A), \quad (2.28)$$

$$R = A_{c_2} = \begin{pmatrix} a_1 & a_2 & a_3 \\ 0 & b_2 & b_3 \\ 0 & 0 & c_3 \end{pmatrix}, \quad (2.29)$$

$$Q = G_{c_2} \cdot (G_{b_1} \cdot G_{c_1}). \quad (2.30)$$

Після розрахунків отримують матриці Q і R . Матриця Q ортогональна, а R — трикутна. Тоді:

$$A = Q \cdot R. \quad (2.31)$$

В. Застосування значень власної матриці для сингулярного розкладання

Для обчислення сингулярної декомпозиції значень необхідно знайти власні значення і власні вектори матриць, які можна розкласти [86,87,99].

За визначенням, власний вектор матриці A називається вектором k таким, що:

$$A \cdot k = l \cdot k \quad (2.32)$$

Де l - власне значення для a .

Згідно з теорією методу декомпозиції сингулярних значень, $A = U \cdot S \cdot V^T$. Зауважимо, що U і V мають властивості ортогональності:

$$U^T \cdot U = V^T \cdot V = I \quad (2.33)$$

Де I — одинична матриця, в якій елементи, розташовані на головній діагоналі, рівні один одному, а всі інші — нулю.

Отже:

$$A^T = U^T \cdot S \cdot V. \quad (2.34)$$

Звідси:

$$A \cdot A^T = U \cdot S \cdot V^T \cdot V \cdot S \cdot U^T = U \cdot S \cdot S \cdot U^T = U \cdot S^2 \cdot U^T, \quad (2.35)$$

$$A \cdot A^T = V \cdot S \cdot U^T \cdot U \cdot S \cdot V^T = V \cdot S \cdot S \cdot V^T = V \cdot S^2 \cdot V^T, \quad (2.36)$$

$$A \cdot A^T \cdot U = U \cdot S^2 \cdot U^T \cdot U = U \cdot S^2, \quad (2.37)$$

$$A \cdot A^T \cdot V = V \cdot S^2 \cdot V^T \cdot V = V \cdot S^2. \quad (2.38)$$

Після аналізу двох останніх формул стає очевидним, що власними векторами для матриці $A \cdot A^T$ є стовпці з матриці U . Зауважмо, що власні значення для матриць $A \cdot A^T$ і $A^T \cdot A$ є числа $S = \text{diag}(l_1, l_2, \dots, l_n)$.

Щоб знайти матриці V і U , обчислюємо матриці $A \cdot A^T$ або $A^T \cdot A$. Тоді обчислюємо другу матрицю, використовуючи наступний вираз:

$$\begin{cases} V = (S^{-1} \cdot (U^T) \cdot A^T), \\ U = A \cdot V \cdot S^{-1}. \end{cases} \quad (2.39)$$

Матриця S діагональна:

$$S^{-1} = (l_1^{-1}, l_2^{-1}, \dots, l_n^{-1}). \quad (2.40)$$

Наприклад, ми описуємо операцію методу для знаходження власних значень і векторів початкової матриці.

Основний принцип методу QR полягає в тому, щоб поступово довести до максимальної подібності початкової матриці (A) з аналогічною матрицею A_n . Цей метод використовується з цією метою. Оскільки початкові та приблизні матриці дуже схожі, вони мають однакові власні значення.

За умови, що початкова матриця має реальні і різні власні значення, метод QR призводить до отриманої матриці, на діагоналі є власні значення, а сама матриця є верхньою трикутною.

Однак власні значення початкової матриці можуть бути також комплексними числами. Тоді результатом методу буде блочна верхня трикутна матриця результату. По діагоналі такої матриці розміщуються блоки (першого і другого порядків). Блоки першого порядку мають власні дійсні числа. У блоках другого — комплексні числа.

Слід зазначити, що комплексних власних значень не існує в симетричних матрицях, тому їх зручно використовувати в таких методах.

Метод QR виконується в таких стадіях:

1. На вхід методу отримують матрицю A ;
2. Обчислюються матриці R_n (верхня трикутна) і Q_n (діагональна).

Добуток цих матриць відповідає початковій матриці $A_n = Q_n \cdot R_n$;

3. Розраховуємо $A_{n+1} = Q_n \cdot R_n$;

4. У випадку, якщо A_i не є верхньою трикутною, метод повертається до другого ступеня. Якщо матриця трикутна припустимо, що діагоналі матриці є її власними значеннями. Також матриця, утворена множенням матриць Q кожної ітерації методу містить власні вектори початкової матриці.

Застосування модифікованого методу для проведення сингулярної декомпозиції значень

Зі збільшенням розмірності даних виникає необхідність обробки даних не на одному обчислювальній пристрої, а в розподіленій обчислювальній системі. У таких системах можна модифікувати метод сингулярної декомпозиції значень, в якому один і той же тип операцій, наприклад додавання даних або матриці множення, діляться на менші частини і обробляються одночасно різними пристроями. Завдяки такому підходу можна прискорити час роботи методу і, відповідно, дати рекомендації користувачам швидше. Більш детальна реалізація моделі обробки даних з використанням розподілених систем обговорюється в наступному розділі.

2.3. Дослідження алгоритму федеративного сингулярного розкладу даних

Експериментальне дослідження обробки масивів даних у розподіленій системі

Для проведення експериментального дослідження ефективності запропонованого методу розподілу Singular Value Decomposition було запропоновано створити програмну модель на мові Java.

Для реалізації розподілених обчислень використовувалися три технології:

- MPI (Message Passing Interface) — між вузлами, які виконують обчислення даних, і основним вузлом обмінюються великою кількістю службових повідомлень. Зі збільшенням обсягу даних, які необхідно обробити, кількість службових повідомлень може займати значну частину обчислювальних ресурсів, що призводить до погіршення продуктивності системи;
- Hadoop не використовує значної кількості службових повідомлень. Всі отримані для розрахунку дані записуються в постійну пам'ять пристрою і після обробки видаляються. Час запису та видалення даних у цій технології сповільнюється;
- Технологія Spark схожа на Hadoop, але використовує RAM (оперативну пам'ять) замість постійної пам'яті. Особливо доцільно використовувати цю технологію при обробці великих обсягів даних.

Розрахунок декомпозиції сингулярного значення в системах, що використовують MPI

Реалізовано стандартний метод пошуку сингулярного матричного розкладу, описаний у попередньому розділі. Під час експерименту досліджувалися залежності часу виконання програми від способу обчислення та розмірності даних.

Для створення середовища тестування ми використовували програмні засоби та вбудовані методи для моделювання роботи розподілених систем на основі технологій MPI, Hadoop та Spark.

Кількість обчислювальних процесів у розподіленій системі змінювалася від 1 до 16 для визначення переваг використання запропонованого модифікованого алгоритму. Розподіл навантаження в змодельованій системі здійснювався процесом програмного керування, дані розподілялися для обробки рівномірно між усіма доступними процесами. При цьому враховувалися можливі незначні затримки в обробці окремих обчислювальних процесів своїх завдань. Тому, коли окремі процесори були зайняті, дані пересилалися на інший вільний.

Такий підхід значно підвищив ефективність алгоритму, особливо коли нам потрібно було обробляти великі дані. Надійність обробки інформації була підвищена також завдяки тому, що навантаження розподілялося між декількома процесорами, і проблеми або неготовність одного з них не зупиняли весь алгоритм.

Розподілений алгоритм рекомендацій. Алгоритм SVD широко використовується в інформаційних системах обробки даних. Враховуючи те, що в нашому дослідженні ми запропонували використання розподілених систем обробки даних, було вирішено модифікувати існуючий алгоритм СВД і розділити його на дві частини: операції, які могли б виконуватися паралельно різними обчислювальними пристроями; і операції, які було надто важко виконати паралельно, тому було вирішено використовувати традиційну послідовну обробку даних.

Таким чином, деякі однотипні операції, такі як множення матриць і піднесення до степеня, були розподілені між декількома обчислювальними процесами, що прискорило роботу алгоритму. Для досягнення максимальної ефективності модифікованого алгоритму було проведено експериментальне визначення кількості операцій, що розпаралелюються [89,90].

На Рис.2.3 показаний алгоритм модифікації методу SVD в розподілених системах.

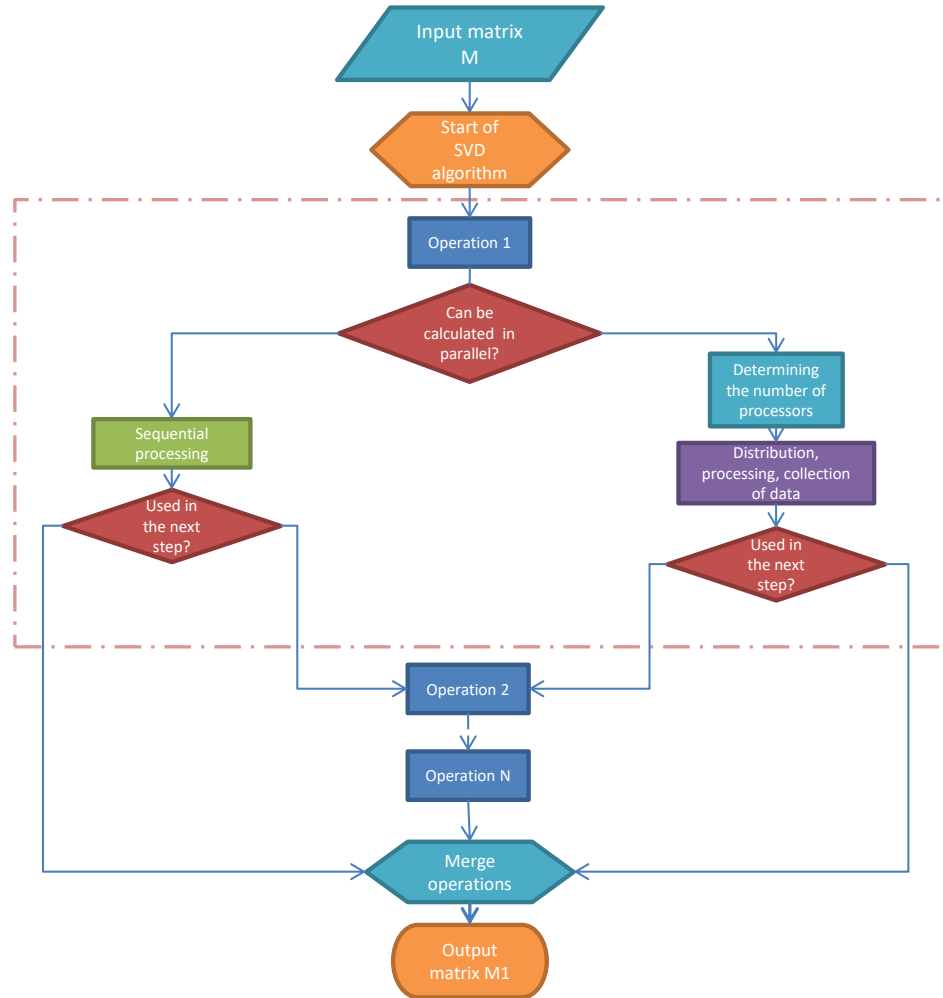


Рис.2.3. Алгоритм визначення максимальної кількості розподілених операцій в алгоритмі СВД.

Згідно з Рис.2.3, алгоритм виконував послідовне виконання класичного алгоритму SVD. Однак на кожному кроці, де необхідно було виконати певну дію над даними, перевірялася можливість їх підпорядкованого поділу для обробки різними процесорами. При моделюванні роботи даного алгоритму використовувалася 4-процесорна архітектура розподіленої системи. Результати моделювання показані на Рис.2.4.

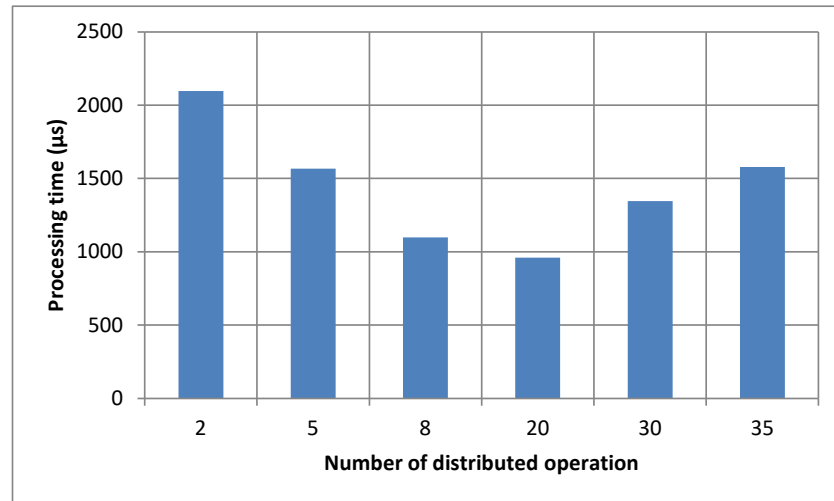


Рис.2.4. Залежність часу обробки SVD від кількості розподілених операцій.

За результатами, отриманими на Рис.2.4, ми зробили висновок, що зі збільшенням розподіленої кількості виконуваних операцій тривалість обчислень спочатку зменшувалася, а потім зростала. Це було тому, що розподілені обчислення невеликих операцій займають більше часу, ніж їх просте послідовне виконання. Тому для подальших досліджень ми використовували розподілене виконання не більше 20 операцій. Запропонований алгоритм дозволив визначити необхідний ступінь модифікації алгоритму SVD для різних архітектур розподілених систем з найбільшою ефективністю. Реалізація методу декомпозиції сингулярного значення в розподіленій системі показана на Рис.2.5.

На Рис.2.5 показана робота немодифікованого методу Singular Value Decomposition, в якому всі дані обробляються одним процесором, і модифікованого, в якому завдання розподіляється на кілька процесорів. Розподілена обробка може підвищити ефективність обчислювальної системи. Елементи модифікованого методу виділені червоним кольором.

На Рис.2.6 і в Таблиці 2.1 показані результати експерименту. Наведено залежність часу виконання сингулярного розкладу матриці розмірністю 120 рядків і 120 стовпців залежно від кількості обчислювальних процесів.

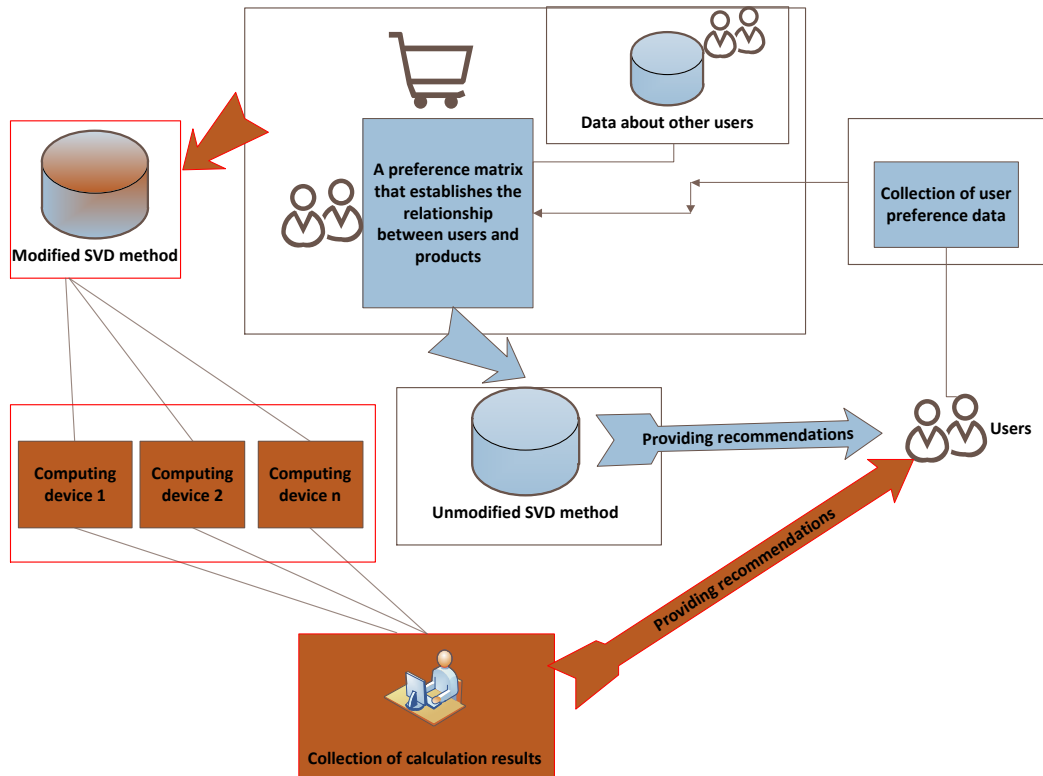


Рис.2.5. Реалізація методу СВД в розподіленій системі.

Таблиця 2.1. Залежність часу виконання розподілених обчислень сингулярної декомпозиції даних від кількості обчислювальних процесів.

Кількість процесорів	Час обробки (мкс)
1	1132
2	1060
4	1006
8	910
16	720
32	620

З Таблиці 2.1 і Рис.2.6 зрозуміло, що збільшення кількості процесів зменшує обчислення часу обробки, оскільки дані були розділені на менші частини, які швидше оброблялися процесором.

На Рис.2.7 і в Таблиці 2.2 представлено залежність часу обробки обчислень від розміру масиву обчислених даних під час обробки в двопроцесорній системі.

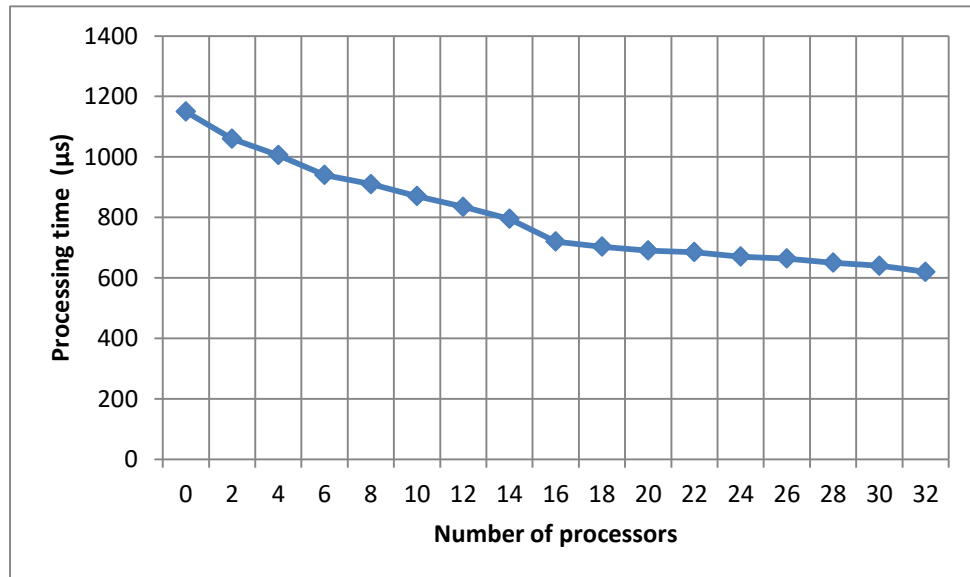


Рис.2.6. Залежність часу обробки SVD від кількості обчислювальних процесорів.

Таблиця 2.2. Залежність часу обробки від розміру масиву обчислених даних

Розмір матриці	Час обробки (мкс)
8	9
15	14
30	20
45	89
60	121
80	343
95	503
105	824
120	1033

З Таблиці 2.2 і Рис.2.7 можна зробити висновок, що зі збільшенням розмірності даних час обробки збільшився. Для більш ефективного виконання обчислень необхідно було збільшити кількість процесорів, як показано на Рис.2.6.

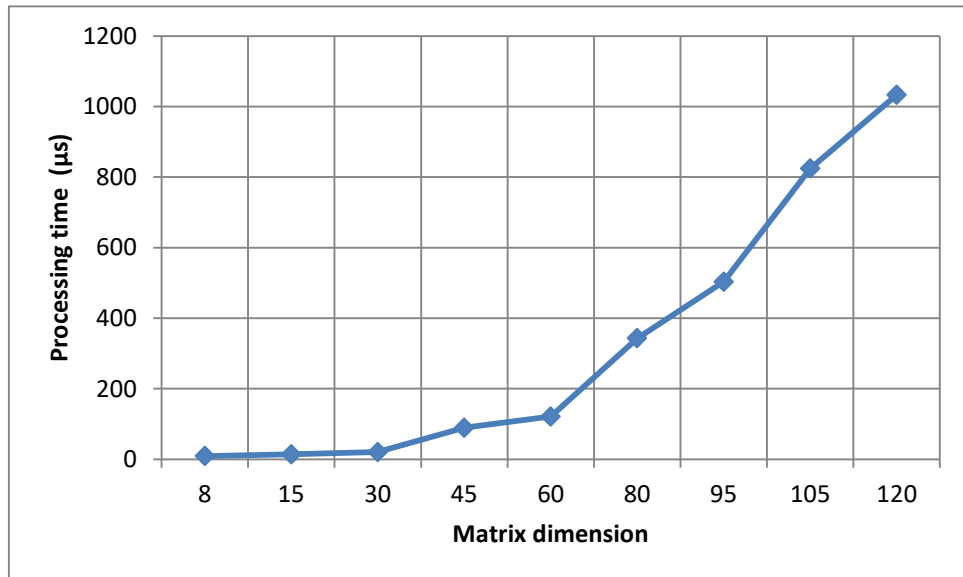


Рис.2.7.Залежність часу обробки розрахунків кількості процесорів від розмірності даних у двопроцесорній системі.

За результатами експериментів ми зробили висновок, що час виконання програмного алгоритму зростає зі зменшенням кількості обчислювальних пристроїв. Для однопроцесорної системи, яка, отже, не є розподіленою, час обчислення був найдовшим.

Порівняння ефективності програми MPI для розподілених обчислень для різних обсягів даних

Були проведені експерименти з розрахунку QR-розкладу матриці для матриць різного розміру з використанням різної кількості технологічних пристроїв у розподіленій системі.

1. Згенерували матрицю розміром 20 рядків на 20 стовпців.

Цей обсяг даних вважався невеликим.

Кожен елемент матриці був типу double (8 байт). Тоді загальна сума складала:

$$C = 8 \cdot 20 \cdot 20 = 3200 \text{ Byte} = 3.2 \text{ Kbyte}$$

1)

Кількість процесорів змінювалася послідовно від 1 до 32.

Результати експерименту показані на Рис.2.8.

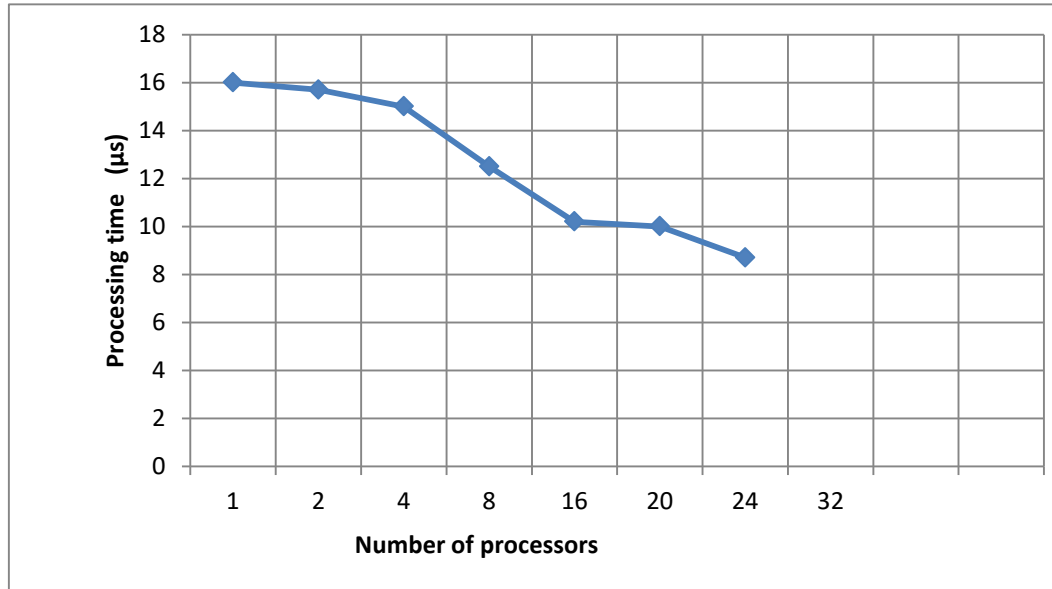


Рис.2.8.Залежність часу обробки СВД від кількості обчислювальних процесорів при розмірі матриці 20×20.

З графіка можна зробити висновок, що зі збільшенням кількості процесорів час обробки зменшувався.

2. Збільшили обсяг даних. Для розрахунку ми взяли матрицю рангу 200 x 200. Кількість інформації:

$$8 \cdot 200 \cdot 200 = 320,000 \text{ Byte} = 320 \text{ КБайт} \quad (2.42)$$

Результат показано на Рис.2.9. Ми дійшли висновку, що час обробки значно збільшився. Однак із збільшенням кількості процесорів час скорочувався.

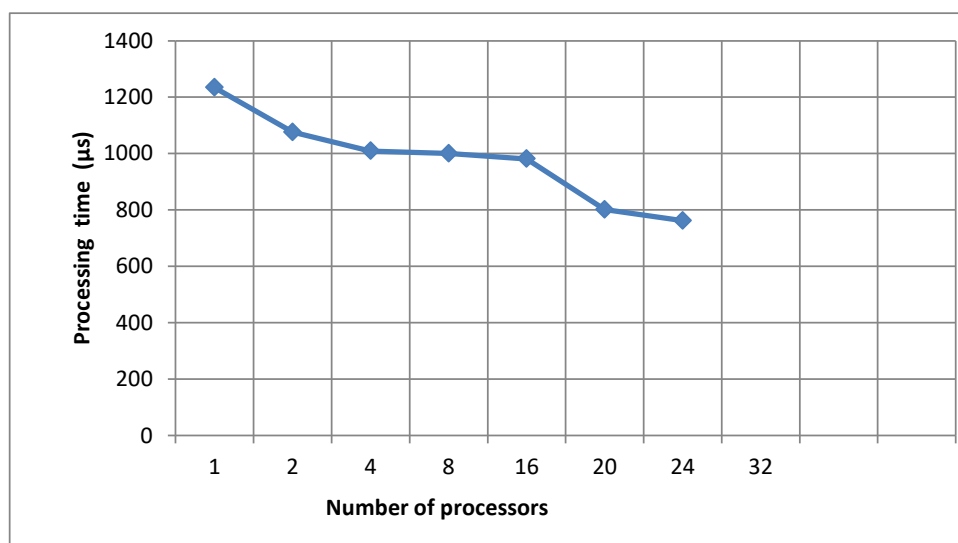


Рис.2.9.Залежність часу обробки SVD від кількості обчислювальних процесорів при розмірі матриці 200 x 200.

3. Для розрахунку ми взяли матрицю рангу 1500 x 1500. Обсяг інформації: $8 \times 1500 \times 1500 = 18\,000\,000$ байт = 18 МБ (Рис.2.10).

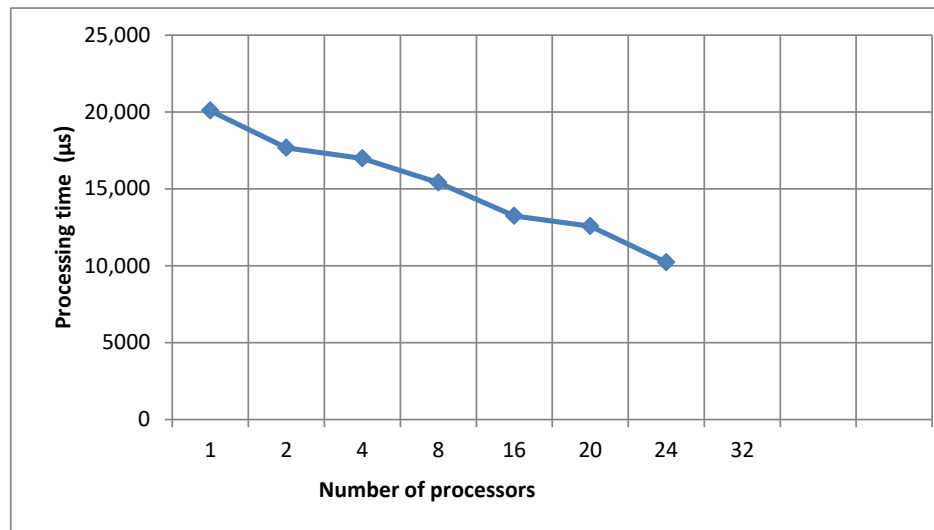


Рис.2.10.Залежність часу обробки СВД від кількості обчислювальних процесорів при розмірі матриці 1500×1500.

На графіку видно, що час обчислень зменшувався зі збільшенням кількості процесорів.

Загальні результати експериментів наведені в Таблиці 2.3.

Таблиця 2.3. Залежність часу обробки від кількості обчислювальних процесів і розміру матриці.

Кількість процесорів	Розмір матриці ($n \times n$), n		
	1500	200	20
Час обробки (мкс)			
1	20,286	1564	18.1
2	20 076	1234	16
4	17 654	1076	15.7
8	16 976	1009	15
16	15 400	1000	12.5
20	13,234	981	10.2
24	12 567	801	10
32	10,213	762	8.7

За результатами експериментів зроблено висновок, що зі збільшенням обсягів даних час обчислень постійно зростає. Для підвищення продуктивності системи необхідно було здійснювати розподілену обробку інформації. Наприклад, обчислення двовимірною масиву даних, представленого у вигляді матриці рангу 20, на звичайному нерозподіленому процесорі зайняло більше 20 мс. При використанні розподіленої системи, що складається з 32 процесорів,

час обробки скоротився вдвічі. Подібні показники можна спостерігати і для інших обсягів вхідних даних. Таким чином, для розміру матриці 200×200 час обробки було зменшено з 1564 мкс на одному процесорі до 762 мкс на 32. Для найбільшого масиву даних рангу 1500 також можна було вдвічі скоротити час обробки за допомогою розподіленої системи з багатьма обчислювальними пристроями. З проведених досліджень ми дійшли висновку, що для обробки великих масивів даних необхідно використовувати розподілені обчислювальні системи, зі значною кількістю процесорів, які забезпечували необхідну продуктивність обчислювальної системи.

Залежність ефективності обробки даних від прихованих факторів

Щоб сформулювати рекомендації, часто недостатньо обчислити сингулярну декомпозицію матриці переваг. Необхідно враховувати певні початкові умови, якщо система ще не має достатньо інформації для формування результату. До результату роботи також додаються коригувальні коефіцієнти з метою врахування можливих змін уподобань та забезпечення можливості автоматичного навчання системи формування рекомендацій [91,92,93].

Для модифікованого розподіленого алгоритму було обрано різну кількість сингулярних чисел. У результаті ми виявили залежність між швидкістю обчислень і точністю алгоритму.

Проведено моделювання модифікованого алгоритму для додаткового скорочення часу видачі рекомендацій. Щоб досягти цього, ми змінили кількість сингулярних чисел. Результати досліджень показали можливість підвищення ефективності алгоритму SVD, якщо не потрібно враховувати такі показники, як, наприклад, приналежність користувача до певної професії, місце проживання або інший фактор, що підвищує точність персональні рекомендації.

Залежність часу обробки обчислень у системі (з використанням MPI) матриці рангу 120 від керуючих коефіцієнтів наведено на Рис.2.11.

Коефіцієнт прогнозної регуляризації використовував додаткову інформацію про користувача для коригування надання рекомендацій через певний час; фактор початкового навчання ввів певні початкові значення для надання рекомендацій; співвідношення швидкості навчання дозволило нам

обчислювати дані швидше, враховуючи більше параметрів, які характеризують користувача. З Рис.2.11 видно, що зі збільшенням значень керуючих коефіцієнтів формування рекомендацій, що має підвищити достовірність результатів, збільшувався час обробки.

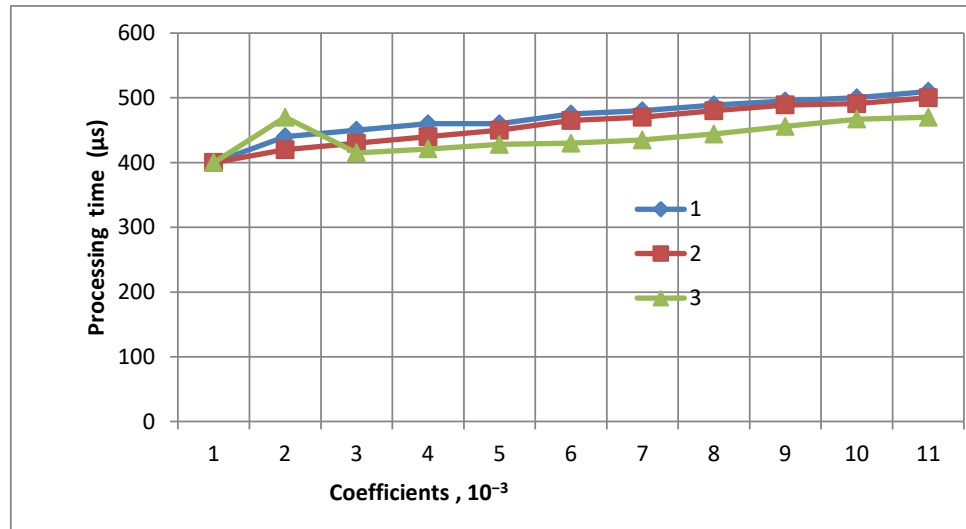


Рис.2.11. Залежність часу обробки обчислень матриці рангу 120 від коефіцієнта регуляризації прогнозів (крива 1), коефіцієнта початкового навчання (крива 2) та коефіцієнта швидкості навчання (крива 3).

Хоча значення коефіцієнтів були незначними (від 0,001 до 0,01), тривалість обчислень зростає з 0,4 до 0,5 мс. Цей результат може зростати з більшою кількістю даних, тому для підвищення швидкості обчислень потрібно враховувати менше факторів [94,95].

Далі ми розглянули обробку сингулярних чисел матриць даних. У матриці Σ , яка є невід'ємною частиною Сингулярного розкладу матриці M , на головній діагоналі розташовані сингулярні числа останньої в порядку спадання значень. Щоб зменшити кількість обчислювальних даних, достатньо було розглянути лише k найбільших сингулярних чисел, а решту відкинути. Також можна врахувати k перших рядків матриць U і V . При перемноженні ми отримали матрицю $M1$, яка повторювала параметри початкової M , але меншого обсягу. Залежність часу обробки обчислень матриці рангу 200 від кількості сингулярних чисел наведено на Рис.2.12.

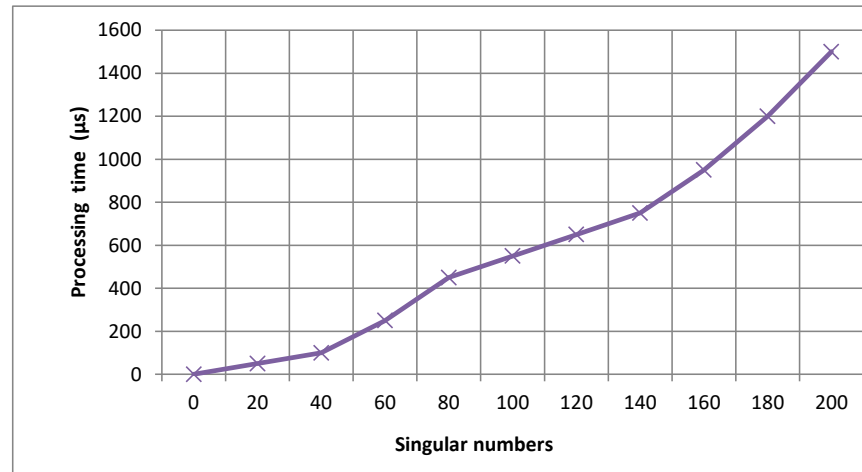


Рис.2.12.Залежність часу обробки обчислень матриці даних рангу 200 від кількості сингулярних чисел.

З графіка ми зробили висновок, що час обробки даних обчислень зменшувався зі збільшенням кількості відкинутих одиничних чисел. При кількості одиничних чисел 20 тривалість обчислень становила 0,05 мс. Коли число збільшилося до 200, час обчислення збільшився до 1,5 мс. Результати показали, що при використанні багатьох сингулярних чисел тривалість обчислень збільшується, але це призводить до підвищення точності алгоритму. Однак для ефективної роботи методу формування рекомендацій доцільно позбутися якомога меншої кількості зі значеннями, близькими до нуля. Слід зазначити, що при використанні розподіленого алгоритму SVD ми могли б використовувати більшу кількість сингулярних чисел за тієї ж тривалості обчислень (Рис.2.13).

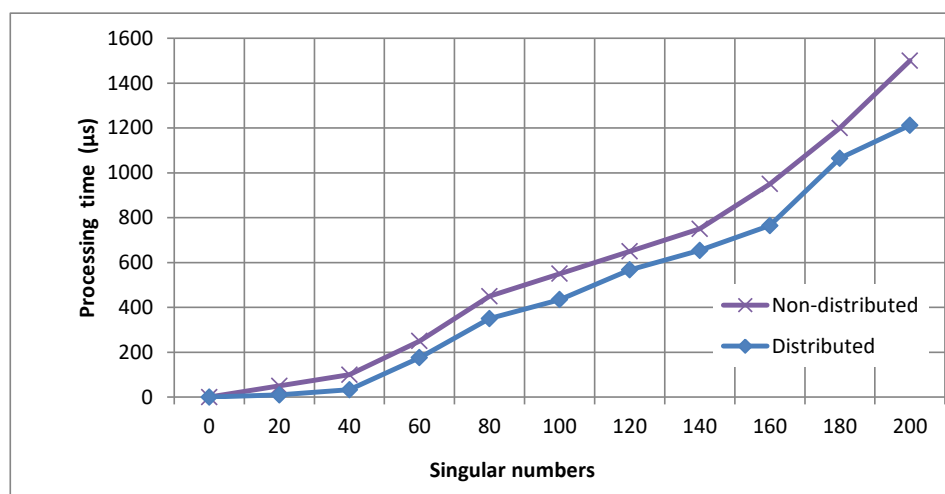


Рис.2.13.Залежність часу обробки обчислень матриці даних рангу 200 від кількості сингулярних чисел у розподілених і нерозподілених системах.

На Рис.2.13 показано тривалість обчислення однакової кількості сингулярних чисел для немодифікованого алгоритму SVD і модифікованого за технологією MPI. Немодифікований алгоритм виконував роботу з урахуванням максимальної кількості додаткових параметрів за 1,5 мс. Модифікований алгоритм обробив цей обсяг даних приблизно за 1,2 мс. Отже, запропонована модифікація алгоритму SVD, що працює в розподілених системах, може бути використана як ефективний спосіб підвищення швидкості та точності обчислень.

Обробка даних за допомогою технології Hadoop і Spark

Дана модель розподілених обчислень модифікованого методу SVD може бути використана для визначення доцільності використання складної обчислювальної системи для вирішення завдань. Вхід моделі надає параметри кількості даних, що підлягають обробці, а вихід – час обробки для різної кількості обчислювальних пристроїв. Для компаній, які залежать від швидкої реакції на зміну уподобань користувачів, кілька секунд мають вирішальне значення для надання рекомендацій. Наприклад, користувач сайту шукає в інтернет-магазині певний товар, вказуючи характеристики та прийнятну ціну. Система рекомендацій, яка за 3 секунди аналізує свої потреби та надає схожу та цікаву пропозицію, підвищує вірогідність вибору потрібного товару. Натомість, якщо на формування рекомендацій йде 20–30 с, користувач, не знайшовши багато цікавих пропозицій, може просто покинути сайт, не зробивши покупки. Такі ситуації трапляються досить часто, і від них залежить значна частина прибутку збутових компаній. Це особливо важливо зараз, коли багато магазинів працюють онлайн.

На Рис.2.14 показано залежність часу обробки розрахунку матриці даних розміром 1500 для різних технологій розподіленої обробки даних в залежності від кількості процесорів.

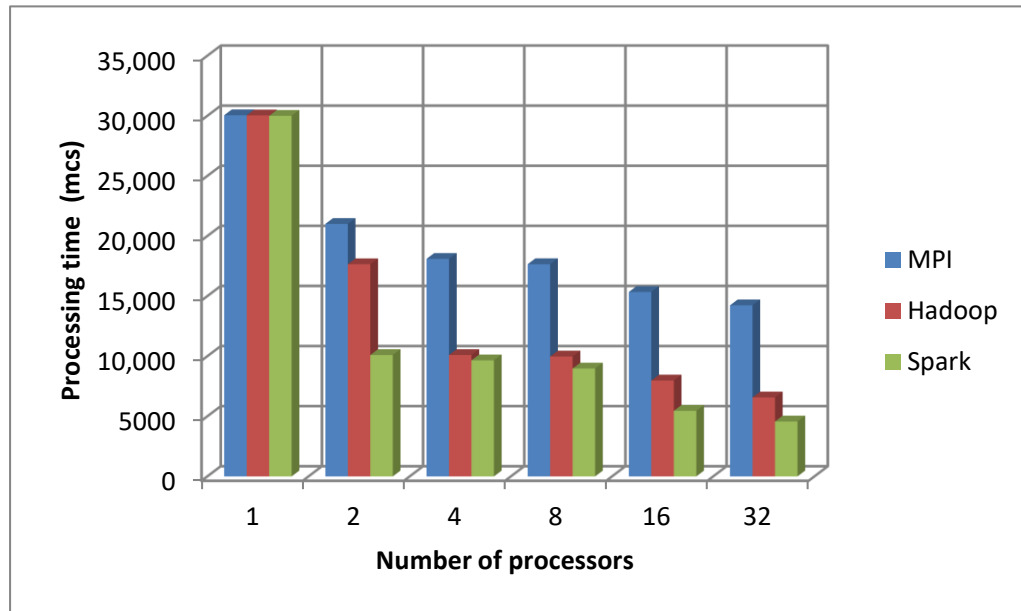


Рис.2.14.Залежність часу обробки SVD розміром 1500 для різних технологій розподіленої обробки даних від кількості процесорів.

З Рис.2.14 можна зробити висновок, що зі збільшенням вимог до обчислювальних ресурсів доцільно використовувати розподілену систему з більшою кількістю процесорів. Технологія Spark була найбільш ефективною за часом. При використанні одного обчислювального процесу час обчислення для всіх технологій становив 30 мс. Однак коли кількість процесорів подвоїлася, технологія MPI обчислила аналогічний обсяг даних за 20 мс, Hadoop приблизно за 17 мс, а Spark за 10 мс.

При використанні 32 процесорів ми побачили, що технології MPI, Hadoop і Spark виконували обчислення за 13, 6 і 4 мс відповідно. Це довело, що технологія Spark виконувала обчислення великих обсягів даних найшвидше з усіх розглянутих у дослідженні технологій. На Рис.2.15 показано залежність часу обробки розрахунку рекомендацій користувачам в залежності від розмірності вхідних даних для розподілених систем і розподілених з використанням різних технологій обробки даних:

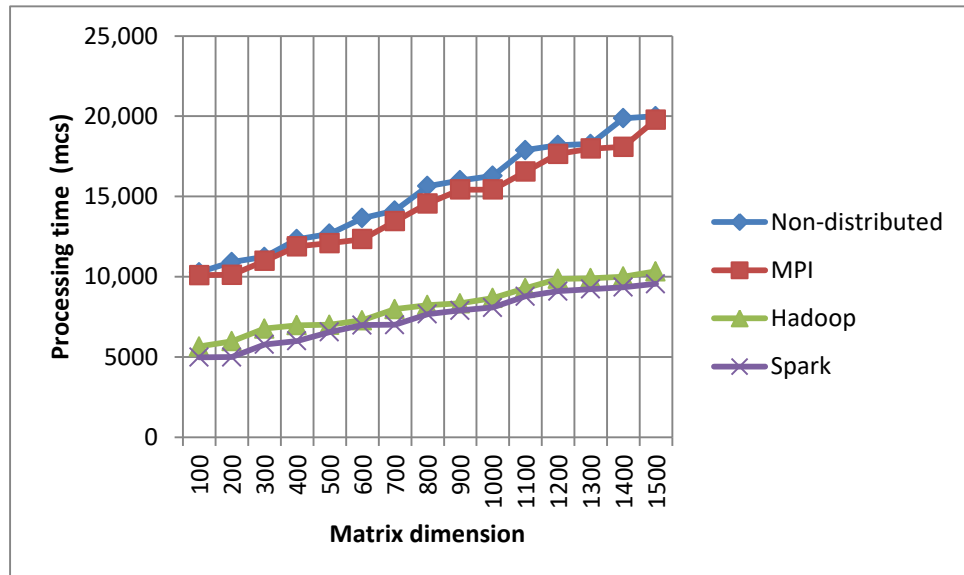


Рис.2.15.Залежність часу обробки рекомендацій користувачам від розмірності вхідних даних для немодифікованого методу SVD (1); модифікований метод на основі технології MPI (Message Passing Interface) (2) модифікований метод на основі технології Hadoop (3); модифікований метод на основі технології Spark (4).

З Рис.2.15 зрозуміло, що метод SVD, який працював без модифікації, показав гіршу продуктивність, порівняно з модифікованим методом, який працював у розподілених системах. Для підвищення ефективності обчислень, залежно від вимог до часу обробки обчислень, доцільно було використовувати технології Hadoop та Spark. Технології Hadoop і Spark були найбільш ефективними при використанні в розподілених системах з великими обсягами даних. Для цих технологій час обчислення матриці даних рангу 1500 не перевищував 10 мс. Натомість для технології MPI час обчислення становив 20 мс.

Результати показали, що Spark продемонстрував найбільшу ефективність серед досліджуваних технологій розподілених обчислень. Перевага над Hadoop в тривалості обчислень склала в середньому 0,5 мс. Для систем рекомендацій це визначає тривалість обробки частини даних користувача. Часто таких фрагментів багато, тому, якщо системі потрібно швидко надати рекомендації для розрахунків, пропонується вибрати технологію Spark.

На Рис.2.16 показано зв'язок, подібний до показаного на малюнку 15, але, аналізуючи дані, отримані на попередніх графіках, тривалість обчислень

стосується більшої кількості даних. Таким чином, зрозуміло, що для обробки великих обсягів даних необхідно використовувати розподілені обчислювальні системи.

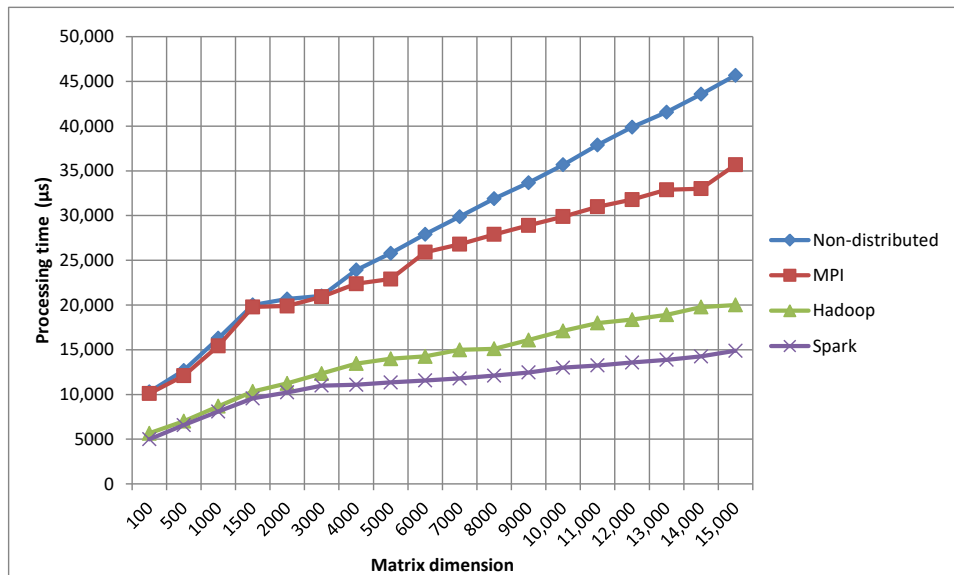


Рис.2.16. Залежність часу обробки рекомендацій користувачам для великих обсягів від розмірності вхідних даних для немодифікованого методу SVD (1); модифікований метод на основі технології MPI (2); модифікований метод на основі технології Hadoop (3); модифікований метод на основі технології Spark (4).

Згідно з Рис.2.16 зроблено висновок, що серед розподілених технологій технологія MPI найдовше обробляла дані (від 10 до 35 мс, залежно від обсягу даних). Матриця даних з максимальним розміром 15 000 була оброблена Hadoop за 20 мс, Spark за 15 мс. На Рис.2.16 показано співвідношення, яке дозволяє зробити висновок, що для обробки великих обсягів даних доцільно було використовувати технологію Spark. Технології MPI і Hadoop також мали переваги перед звичайною нерозподіленою системою. Ця залежність дозволила вибрати оптимальний спосіб обробки даних на основі необхідного часу обробки та обсягу даних. Якщо великій компанії економічно вигідно витратити ресурси на потужну обчислювальну систему і швидко обробляти дані, то найкращий вибір технології – Spark. Порівняно з нерозподіленою системою система, що використовує технологію Spark, може обчислювати дані в 2–3 рази швидше.

У дослідженні використовувався обсяг даних, який не можна було обробити на обладнанні, що використовується в експерименті, без зміни методу обробки даних і використання розподілених обчислень. Тому можна сказати, що для цього дослідження робота проводилася з Big Data. Отримані результати можуть бути використані для продовження досліджень на більш потужному обладнанні та підвищення ефективності обчислень Big Data.

З результатів моделювання також можна зробити висновок не тільки про порівняння нерозподілених і розподілених систем за допомогою алгоритму SVD, але й про аналіз роботи в різних архітектурах розподілених систем. У процесі моделювання змінилася кількість обчислювальних процесів. Результати показали, що модифікований алгоритм дає найбільше переваг у розподілених системах з великою кількістю обчислювальних пристроїв та значним навантаженням на них. Крім зміни кількості процесорів ми використовували різні технології: MPI, Hadoop, Spark. Це дозволило нам створити кілька різних розподілених алгоритмів SVD з різними результатами. Використовуючи отримані результати дослідження, вдалося спрогнозувати роботу алгоритму SVD в різних розподілених системах з різними обсягами оброблюваних даних. Це також дозволило вибрати розподілену реалізацію алгоритму, яка давала найкращий результат для певних вимог і умов у конкретній системі.

2.4. Принцип обробки даних у розподілених системах за допомогою методів машинного навчання

У роботі було проведено дослідження обробки великих обсягів даних у довідкових системах. Для цього було використано модифікований метод Singular Value Decomposition, який запропоновано для використання в системах розподіленої обробки.

Для методу роботи в розподілених системах використовувалися технології MPI, Hadoop і Spark. На основі отриманих результатів побудовано модель, яка дозволила визначити доцільність використання розподілених систем та модифіковано метод однозначної декомпозиції в залежності від задач, що вирішуються.

У майбутньому цей метод можна вдосконалити, застосувавши методи машинного навчання, що дозволить запропонованому методу навчатися самому та давати точніші рекомендації користувачам. Ця тема також є перспективною для подальших досліджень і експериментів з новими технологіями розподіленої обробки даних і обчислювальними платформами з метою пошуку оптимальних шляхів підвищення ефективності обробки великих обсягів даних.

Зростання даних вимагає впровадження нових підходів до організації розрахунків. У зв'язку з цим широко використовуються технології розподіленої та паралельної обробки.

У роботі розглянуто використання розподілених інфокомунікаційних систем при обробці великих обсягів даних. Досліджено роботу рекомендаційних систем аналізу інформації. Було розглянуто огляд підходів до скорочення обчислень за допомогою методів спільної фільтрації даних.

Досліджено впровадження методів оптимізації наборів даних у розподілених системах. Прикладом став метод СВД. Новизна цього підходу полягала в тому, що він дозволив нам упорядкувати інформацію та відкинути зайві дані, що призвело до швидшої та ефективнішої обробки. Інформація про користувачів системи, зібрана з різних джерел, була відсортована та проаналізована. На основі цього були сформовані рекомендації щодо послуг системи, які можуть зацікавити її клієнтів. Оскільки дані, отримані від систем рекомендацій, були великими, для їх обробки використовувалися розподілені обчислення, щоб забезпечити необхідну продуктивність.

Проведено моделювання роботи СВД при обробці великих масивів інформації. Розмір даних і кількість вузлів розподіленої системи змінено за допомогою протоколу MPI для отримання оптимальних значень продуктивності. Було встановлено, що коли кількість обчислювальних пристроїв збільшилася з 1 до 32, час обчислення для різних обсягів даних скоротився вдвічі. Сингулярне розкладання матриці рангу 1500 у розподіленій 32-х процесорній системі було виконано приблизно за 10 мс, а в нерозподіленій – за 20 мс.

Проведено моделювання SVD в системах з використанням протоколу обміну повідомленнями MPI та програмної моделі MapReduce. Результати показали, що для невеликих обчислень ефективніше використовувати MPI, а при обробці більше 10 Мбайт інформації доцільно використовувати MapReduce завдяки простій реалізації та більш швидкій роботі.

Також ми порівняли ефективність технологій Hadoop і Spark, що показало доцільність використання останньої при необхідності швидкої обробки великих обсягів даних. Зокрема, використання модифікованого методу SVD спільно з технологіями Hadoop і Spark дозволило обробляти дані в 2–3 рази швидше, ніж при використанні немодифікованого методу в нерозподіленій системі. У сфері розподілених систем нові дослідження та вдосконалення завжди будуть актуальними. Ця робота може слугувати основою для подальших досліджень в області обробки великих масивів даних за допомогою розподілених або паралельних обчислень [96,97].

Висновки до другого розділу.

У розділі було проаналізовано ефективність використання алгоритму сингулярної декомпозиції. Також було розглянуто роботу федеративного алгоритму сингулярної декомпозиції в розподілених інформаційних системах.

З урахуванням проаналізованого матеріалу було поставлено вимоги до створення та функціонування системи пошуку важливих даних в рекомендаційних системах.

Результатом розділу є розв'язання задач № 2,3 дослідження.

РОЗДІЛ 3. РОЗРОБЛЕННЯ МОДЕЛІ ОБРОБКИ ДАНИХ В РОЗПОДІЛЕНИХ СИСТЕМАХ З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ.

У розділі проаналізовано використання машинного навчання для управління ресурсами інформаційно-комунікаційних систем промислової інфраструктури. Досліджено використання алгоритмів машинного навчання для підвищення ефективності обробки великих даних. Запропоновано модель та методи для обробки даних великих обсягів в промислових IoT системах на основі модифікованих алгоритмів Fed SVD і Funk SVD.

Результати розділу опубліковано в працях автора [1,3,5,10]

3.1. Дослідження використання машинного навчання для управління ресурсами інформаційно-комунікаційних систем промислової інфраструктури

Промислові системи виконують безліч функцій, необхідних для комфортного життя. Виробництво продукції та надання послуг у великих масштабах вимагає чіткої організації системи та постійної обробки великих обсягів даних. Промислові системи пройшли довгий шлях. Від звичайного використання пристроїв, які виконують обмежені функції та потребують постійного нагляду персоналу, до високоінтелектуальних систем, здатних до самообслуговування та навчання. На зміну централізованим мережам прийшли децентралізовані, які складаються з багатьох компонентів. Замість постійного контролю людини промислові системи виконують багато завдань самостійно. Обробка великих даних є важливою особливістю сучасних промислових систем. Оскільки на виробництві використовується багато приладів різного призначення, їх стан необхідно постійно контролювати. Такий підхід дозволяє визначити неефективність окремих компонентів системи, можливі ризики для безпеки обладнання та персоналу. Термін «великі дані» описує інформацію, яка збирається з різних джерел, обробляється та представляється в зручній для користувача формі. Великі дані в промислових системах постійно надходять від

різних датчиків. Тому для виявлення та відкидання зайвої, а також пошуку необхідної інформації використовуються спеціальні алгоритми аналізу інформації. Швидкість обробки даних надзвичайно важлива, оскільки необхідно дуже швидко приймати рішення щодо управління системою та усунення несправностей [89,99,100].

Промисловий Інтернет речей – це сукупність взаємопов'язаних елементів, інтелектуальних пристроїв, алгоритмів обробки даних і пристроїв керування. Системи ПоТ мають багато переваг перед звичайними промисловими системами. Можливість використання багатьох кінцевих пристроїв, які збирають і обробляють дані, дозволяє швидко реагувати на різні події в системі. Алгоритми обробки даних і спілкування між вузлами сприяють швидкому обміну досвідом і навчанню промислової системи. Промисловий Інтернет речей використовується в багатьох галузях: енергетика, виробництво, транспорт, розумне місто тощо.

Під час пандемії доцільно делегувати програмному та апаратному забезпеченню деякі функції, які раніше виконувала велика кількість персоналу. Використання ПоТ передбачає модернізацію виробничого процесу та залучення великої кількості кінцевих користувачів. Виробництво і збут також інтегровані. Аналіз товару показує, наскільки він затребуваний. У розумному виробництві може бути прийнято рішення про модифікацію продукту або вилучення з обігу. Постійний моніторинг ефективності роботи підприємства дозволяє визначати неоптимальне використання ресурсів і автоматично їх перерозподіляти. Таким чином, машинне навчання та штучний інтелект є важливими компонентами промислового Інтернету речей. Виявлення критичних ситуацій на підприємстві може бути використано для уникнення їх повторення. Обмін даними між різними кінцевими пристроями дозволяє швидше навчатися. Оскільки пристрої ІоТ розташовані далеко одне від одного, процес збору даних може спричинити деякі труднощі:

- навантаження на канали зв'язку при постійній передачі великих обсягів інформації від робочих вузлів до центрального пристрою;

- ризики достовірності та конфіденційності даних користувачів. Для захисту даних слід використовувати спеціальні методи шифрування.

Ці особливості призвели до появи федеративного машинного навчання, в якому всі вузли навчаються на локальних даних і надсилають на центральний вузол лише результати, не розповсюджуючи особисту інформацію. У роботі досліджено ефективність обробки даних в системах ІоТ з використанням федеративного навчання. Також пропонується використовувати алгоритм Singular Value Decomposition для попередньої оптимізації обробки великих даних та утилізації надлишкових даних.

3.2. Алгоритми машинного навчання для обробки даних користувачів промислових систем

Зараз у світі відбувається четверта промислова революція. Вертикальна та горизонтальна інтеграція промислових систем забезпечує автоматизацію обробки даних, створення та розповсюдження продуктів. Розумні галузі можуть виконувати свою роботу швидше та ефективніше, споживаючи при цьому менше ресурсів. Причиною такого зростання продуктивності є використання автоматизованих систем управління та покращення виконання виробничих завдань. Замість громіздких пристроїв, які виконують обмежене коло завдань, розумні галузі використовують більш автономні, але менш потужні. Залучення великої кількості датчиків, які контролюють стан промислової системи, дозволяє здійснювати постійний моніторинг можливих аварійних ситуацій, визначати працездатність окремих компонентів і системи в цілому. Програма Industry 4.0 була запроваджена в Німеччині, щоб визначити важливість розробки інтелектуальних промислових систем і дослідити нові способи їх вдосконалення. Визнання на міжнародному рівні перспектив удосконалення індустриальних систем і впливу цього процесу на людство свідчить про новий етап у розвитку обробної промисловості. Виходячи з розвитку промисловості в країні, ми можемо свідчити про її конкурентоспроможність та успіх на міжнародній арені. Тому постійно впроваджуються нові стандарти цифровізації виробництва, методи оптимізації Big Data, аналізу та управління промисловими системами [101,102].

Концепція промислового Інтернету речей передбачає використання новітніх інструментів управління галуззю. Розподілені датчики постійно збирають дані про стан окремих елементів системи. За допомогою аналізу алгоритми є важливою інформацією для прийняття рішень щодо управління системою. Гнучкі методи управління дозволяють нам швидко реагувати на події на робочому місці та знаходити винахідливі та ефективні методи їх вирішення. Промисловий Інтернет речей повністю відповідає принципу Індустрії 4.0, який передбачає оцифрування всіх етапів виробництва. Важливими етапами є як процес безпосереднього виробництва, так і збут, тобто доставка до кінцевого споживача. Якщо в минулому логістикою товарів займалися виключно люди, то тепер вона може бути інтегрована в глобальну промислову систему. Якщо у користувачів певних послуг є зауваження до виробленої продукції або вона не користується попитом, розумна компанія може автоматично налагодити виробничі процеси для оптимізації продуктивності. Зміни торкнулися багатьох сфер нашого життя. Модернізації потребує не лише традиційне виробництво. Енергетика, охорона здоров'я, транспорт і управління цілими містами є прикладами промислового Інтернету речей. Через ситуацію з пандемією багато компаній були змушені скоротити кількість співробітників і обмежити скупчення персоналу. Тому дуже важливо забезпечити функціонування галузі шляхом дистанційного керування системою та автоматизації багатьох виробничих функцій. Враховуючи те, що промислові системи стрімко розвиваються і змінюються, їх необхідно постійно вдосконалювати, впроваджувати нові методи збору, обробки та зберігання даних. Тому управління інтелектуальними системами має враховувати динамічні зміни програмного та апаратного забезпечення [103,104].

Кіберфізичні системи (CPS) використовуються для розумного управління промисловими системами, які можуть виконувати функції збору, обробки та аналізу екологічних даних за певними критеріями. На основі обробленої інформації об'єкт може взаємодіяти з іншими об'єктами в системі для виконання більш глобальних завдань. Оснащені найсучаснішими пристроями збору даних, елементи CPS можуть з високою точністю виявляти та коригувати

відхилення від встановленої норми в поведінці локальних пристроїв. Такий підхід дозволяє більш ретельно визначити особливості промислової системи та забезпечити більш швидке та ефективне виконання завдань. Розумне виробництво відкриває перспективи для реалізації функцій, які раніше були недоступні. Гнучке управління промисловими системами підвищує рівень якості послуг, тому дає можливість модернізувати існуючі галузі та створювати нові.

Розумне виробництво. Для перетворення класичних промислових систем на інтелектуальне виробництво необхідно використовувати спеціальні алгоритми та методи машинного навчання. Враховуючи те, що сучасні великомасштабні промислові системи складаються із значної кількості пристроїв, можна говорити про розподілену мережеву архітектуру. Відповідно, алгоритми обробки даних також повинні бути поширені. Адаптація методів машинного навчання та глибокого навчання до особливостей промислових систем дозволяє досягти кращих результатів у продуктивності. Децентралізація процесів управління промисловою мережею дозволяє приймати важливі рішення безпосередньо на кінцевих пристроях, що скорочує час обробки даних і завдань. Сучасні інтелектуальні виробничі системи зазвичай використовують кілька виділених вузлів, які обробляють дані з усієї мережі та навчають загальний алгоритм машинного навчання. Інформація з різних кінцевих пристроїв надсилається на аналіз. Однак функціональність кінцевих пристроїв також дозволить їм незалежно взаємодіяти з користувачами, іншими компонентами та продуктом, що виготовляється [105].

Використання інтелектуального виробництва дозволяє зменшити кількість персоналу, який обслуговує процес його роботи. Однак важливою особливістю цифровізації підприємств є не повна відмова від участі людини в процесах управління, а її заміна там, де краще справляються алгоритми аналізу даних і штучний інтелект. Операції, які вимагають швидкої обробки надвеликих даних, пошуку конкретної інформації та швидкої реакції на рутинні стандартні завдання, є прикладами тих завдань, які люди виконують повільніше та менш ефективно, ніж машини. Натомість визначення алгоритмів управління

промисловою системою, стратегії розвитку та програмування системи виконує персонал, який також стежить за правильністю виробництва. При визначенні пріоритетів у промисловій системі основними перевагами та недоліками використання тих чи інших методів є комплексні рішення, які виконуються спільно кваліфікованими фахівцями, та алгоритми обробки даних, що дозволяє максимально ефективно проводити розрахунки.

Кіберфізичні системи в IoT. Кіберфізичні системи – це сукупність новітніх ефективних технологій, які мають поєднати апаратні та програмні ресурси для створення принципово нової інформаційної мережі. Інтеграція кіберфізичних систем з іншими технологіями, такими як Інтернет речей, розширює можливості для розвитку розумного виробництва. Концепцію CPS можна побачити в персональних пристроях, таких як смартфон. Якщо деякий час тому телефон був лише засобом спілкування, то зараз користувачі можуть отримати доступ до величезного спектру послуг. Переосмислення ролі пристроїв промислового Інтернету речей також передбачає надання нових функцій пристроям, які раніше виконували лише певні завдання. Кіберфізичні системи стирають різницю між цифровим і реальним світами. Взаємодія користувачів через високоінтелектуальні пристрої та інтелектуальні промислові системи надає нові можливості керування роботою виробничих процесів.

Зараз багато галузей промисловості зазнають незворотних змін. Модернізація апаратного та програмного забезпечення вимагає змін у виробничому процесі. Необхідність надійного управління системою та розподілу ресурсів призвела до розвитку інформаційних технологій, які забезпечують розподілений збір, обробку та зберігання даних. Інтеграція кількох технологій в один виробничий процес дозволяє користувачам надавати набагато більшу різноманітність послуг. Замість використання окремого програмного та апаратного забезпечення для виконання певних функцій, інтелектуальне виробництво на основі кіберфізичних систем передбачає багатофункціональну структуру. Еволюція індустрії смарт почалася з використання вбудованих систем (ES), які спеціалізувалися на виконанні вузького кола завдань. Потім виникла необхідність інтегрувати вбудовані

системи з мережевою інфраструктурою. Стало зрозуміло, що неможливо обробити всі завдання на одному фізичному пристрої, навіть якщо він має потужні обчислювальні ресурси. Згодом розвиток ES призвів до появи кіберфізичних систем, які в свою чергу стали основою для Інтернету речей (IoT) [106,107,108,109].

Завдяки концепції IoT стало можливим використовувати речі та пристрої по-новому. Розширення функціональних можливостей пристроїв обробки даних та їх здатність самостійно виконувати певні завдання докорінно змінили існуючі інфокомунікаційні системи. Оскільки Інтернет речей довів свою ефективність у вирішенні локальних проблем, промисловий Інтернет речей використовується для більш глобальних проблем. Різноманітні види продуктів можна представити як розумні, багатофункціональні та гнучкі в управлінні системою. Кожен пристрій може виконувати свої завдання та приймати локальні рішення, взаємодіючи з іншими вузлами. Такий підхід полегшує обмін інформацією між пристроями промислового Інтернету речей і швидке навчання. Моделі машинного навчання в таких системах можна навчити швидше, поєднуючи досвід різних пристроїв.

Таким чином, еволюція промислових систем допомогла змінити багато сфер людського життя. Завдяки технічному прогресу користувачі тепер можуть отримувати готові товари та послуги, не турбуючись про особливості компанії, яка їх надає. Промисловий Інтернет речей використовується в розумних галузях. Автоматичне обчислення даних прискорює процес прийняття важливих рішень. Інтелектуальні алгоритми проводять комплексний моніторинг стану системи та визначають ті виробничі одиниці, які потребують вдосконалення. Зміна параметрів виробництва тепер аналізується не вручну, а за допомогою методів машинного навчання та штучного інтелекту. Еволюцію цифрових технологій у промислових системах показано на Рис.3.1.

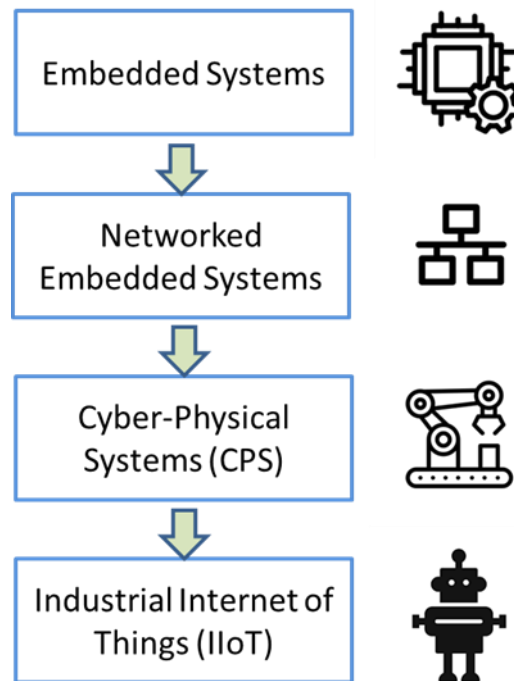


Рис.3.1.Еволюція цифрових технологій у промислових системах

Використання програмного забезпечення для управління промисловими системами викликає багато дискусій. Оскільки багато виробничих ситуацій можна вирішити не на місці, централізоване управління системою раніше не розглядалося. Враховуючи розвиток кіберфізичних систем та Інтернету речей, все більше функцій управління складними процесами на підприємствах виконують програмні алгоритми. Наразі не можна сказати, що програмне забезпечення відіграє ключову роль у процесі прийняття рішень у системах ІоТ. Натомість інтеграція управління за допомогою спеціальних алгоритмів стала важливою частиною розумного виробництва і дозволяє коригувати та усувати аварійні ситуації різного масштабу. Для проектування та експлуатації промислових систем важливо залучати висококваліфікованих спеціалістів, які з максимальною ефективністю можуть керувати як програмним, так і апаратним забезпеченням [110,111,112.113].

3.3. Аналіз використання алгоритмів машинного навчання для підвищення ефективності обробки великих даних

Використання промислового Інтернету речей у сучасному світі відіграє величезну роль. Багато функцій, які раніше виконували спеціалісти на

робочому місці, тепер змушені виконувати онлайн через ситуацію з пандемією. Такий підхід вимагає використання додаткового програмного забезпечення, захищених каналів передачі інформації та швидкої обробки даних. Перевагами ПоТ є використання великої кількості датчиків, контролерів тощо для вимірювання даних, які замінюють значну кількість персоналу. Також ефективніше організувати роботу персоналу. Функції екологічного моніторингу дозволяють очищати повітря щоразу, коли рівень забруднення перевищує допустимі значення. Контроль пересування людей може забезпечити дотримання соціальної дистанції та оптимальну присутність співробітників у кімнатах відпочинку, кафе тощо. Функції віртуальної реальності дозволяють керувати дистанційно без фізичного відвідування промислових об'єктів. Також спілкування між людьми можна здійснювати, уникаючи скупчення людей, тим самим знижуючи ризик захворіти. Таким чином, промисловий Інтернет речей сьогодні дозволяє більш ефективно організувати роботу виробничих систем, піклуючись як про їхню продуктивність, так і про здоров'я працівників. Індустрія 4.0 містить багато компонентів. Штучний інтелект, Інтернет речей, ПоТ та хмарні технології зараз є перспективними сферами (Рис.3.2).

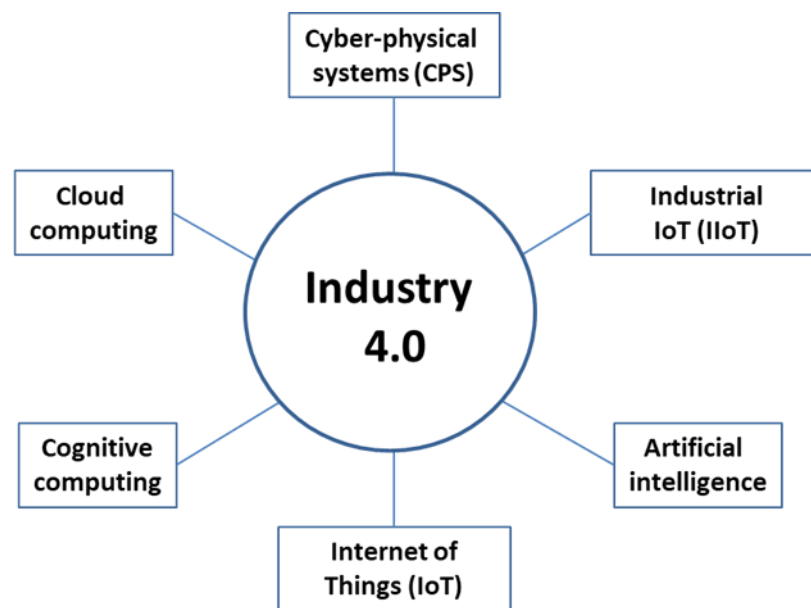


Рис.3.2.Компоненти Індустрії 4.0

Передумовою для створення промислового Інтернету речей був, звичайно, Інтернет речей. Особливістю Інтернету речей є використання

інтелектуальних пристроїв, здатних виконувати певні функції, які спілкуються один з одним і контролюють вузли через Інтернет. Інтернет речей спрощує повсякденне життя людини та виконує багато примітивних функцій. Зібрані дані використовуються для автоматизації роботи в системі IoT і зберігаються на хмарних ресурсах [114]. Успіх Інтернету речей призвів до появи його підгалузей, здатних автоматизувати роботу не тільки окремих будинків, а й великих підприємств. Промисловий Інтернет речей також містить багато кінцевих або периферійних пристроїв. Це датчики, виконавчі пристрої та ПЛК (програмовані логічні контролери). Ці пристрої збирають дані з різних частин підприємства про параметри роботи, продуктивність тощо. Зв'язок через Інтернет дозволяє координувати роботу багатьох компонентів системи.

Промисловий Інтернет речей, на відміну від традиційного IoT, покращує взаємодію між апаратним і програмним забезпеченням. Для ефективного управління виробничими ресурсами необхідно забезпечити високий рівень продуктивності окремих пристроїв, а також використовувати програмні алгоритми, здатні їх координувати. Безперервний збір, обробка та передача даних для зберігання на хмарних ресурсах вимагає залучення технологій на всіх рівнях моделі взаємодії відкритих систем (Open Systems Interconnection, OSI).

Основними компонентами промислового Інтернету речей є:

- розумні пристрої збору даних;
- мережева інфраструктура;
- обробка даних;
- програми.

Використання IoT дозволяє організувати роботу та управління розумним виробництвом набагато ефективніше, ніж у звичайних централізованих системах. Здатність збирати та аналізувати інформацію про нестандартні ситуації дозволяє не тільки швидко їх усувати, але й заздалегідь передбачати та приймати відповідні рішення [114,115,116].

Промислові системи Інтернету речей гнучкі у використанні та масштабовані. Оскільки потреба у використанні ресурсів для виробництва постійно змінюється, необхідно залучати нові та вивільняти надлишкові.

Раніше промислові системи склалися з певної кількості апаратного забезпечення, яке визначалося з урахуванням прогнозованого навантаження. Якщо обсяг виробництва різко зростає, потрібен був час для переоснащення системи, додавання нових пристроїв. У разі зниження навантаження на промислову систему частина обладнання простоювала, що економічно не вигідно. У системах ІоТ можливе залучення хмарних ресурсів, які розподіляються автоматично на основі аналізу стану підприємства та навантаження [10].

Використання ІоТ передбачає новий підхід до роботи з Big Data. Аналіз інформації з різних кінцевих пристроїв використовується для визначення параметрів системи, які так чи інакше впливають на її продуктивність. Зміни в роботі окремих компонентів системи можуть свідчити про проблеми локального або глобального характеру. Швидкість вирішення таких проблем важлива для забезпечення безперервного функціонування виробництва [117].

Однак використання Інтернету речей має і певні негативні сторони:

- загрози безпеці. Оскільки багато функцій виконуються кінцевими пристроями, існує підвищений ризик того, що зломисники можуть приховати себе під користувачів системи та надіслати неправильні дані. Обробка інформації користувача центральними вузлами також вимагає забезпечення конфіденційності даних при передачі по каналах зв'язку і безпосередньої обробки. ІоТ включає багато користувачів, які використовують свої дані під час взаємодії з системою. Тому забезпечення достовірності та конфіденційності даних є пріоритетом;

- сумісність. Оскільки сучасні промислові системи часто використовують різні типи обладнання, мережеві протоколи, служби тощо, їх сумісність необхідна для належної роботи виробництва.

Слід зазначити, що переваг промислового Інтернету речей набагато більше, ніж недоліків. Завдяки використанню спеціальних протоколів зв'язку, методів захисту даних, спеціального обладнання можна забезпечити надійну та ефективну роботу інтелектуального виробництва. Архітектура системи ІоТ показана на Рис.3.3.

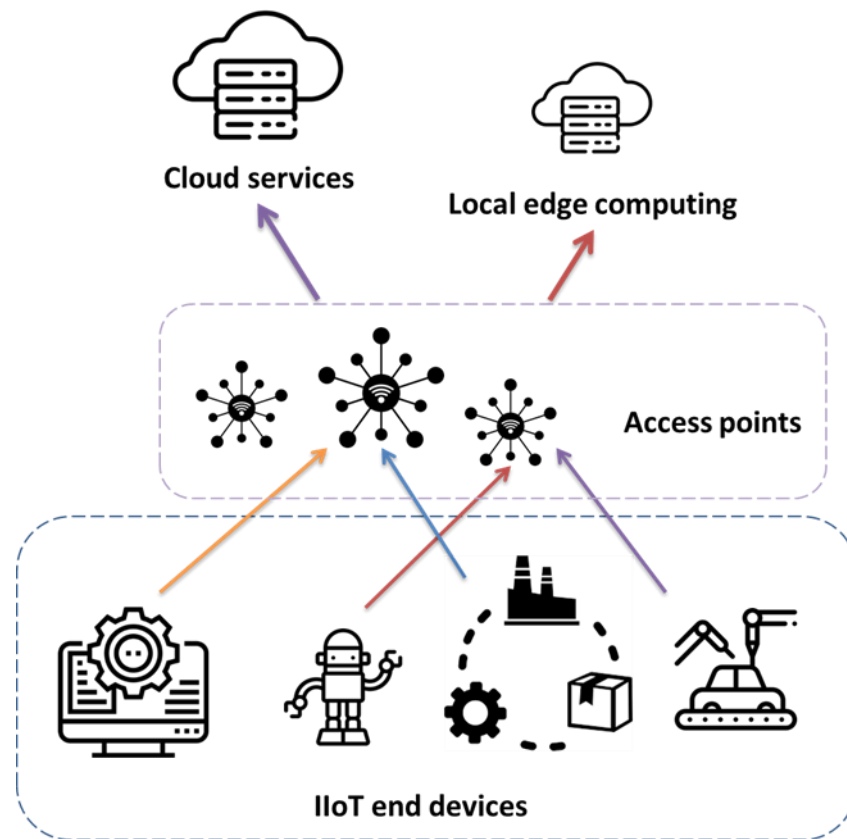


Рис.3.3.Архітектура системи ІІоТ

ІІоТ є важливим етапом у розвитку цифрового виробництва. Постійне вдосконалення його компонентів дозволяє підвищити ефективність промислових систем та усунути ризики та недоліки. Структура промислового Інтернету речей є оптимальною для концепції Індустрії 4.0 і дозволяє використовувати різноманітні технології та методи організації виробництва. Використання методів машинного навчання, штучного інтелекту та поглибленого аналізу покращує процес управління та обробки даних [118,119].

Обробка великих даних в ІІоТ

Машинне навчання в промислових системах. Оскільки звичайних обчислювальних алгоритмів недостатньо для обробки великих обсягів даних з різних пристроїв, методи машинного навчання допомагають аналізувати й оптимізувати інформацію. Завдяки машинному навчанню пристрої ІІоТ можна навчати та самонавчати, тому вони можуть вирішувати задачі різного рівня складності самостійно та без допомоги людини .

Машинне навчання (machine learning, ML) — це підгалузь штучного інтелекту, яка використовує методи аналізу статистичних даних для прийняття

рішень, які не були заздалегідь запрограмовані людьми. Алгоритми машинного навчання обробляють вхідні дані і на їх основі приймають рішення про подальші дії. Ефективність ML для вирішення задач класифікації, кластеризації та регресії сприяє його впровадженню в різних областях. Промислові інтелектуальні системи також мають багато проблем, які вимагають складних технологій обробки даних. Визначення важливих значень у величезних масивах інформації та прийняття рішень на основі попередніх подій без втручання персоналу значно скорочує час реагування на критичні умови промислової системи. Однак використання ML в IIoT також має деякі особливості. Зокрема, розподілена архітектура виробничих систем і величезна кількість кінцевих пристроїв викликають проблеми в роботі алгоритмів машинного навчання. Крім того, постійна передача великих даних створює навантаження на канали зв'язку та обчислювальні пристрої [119,120,121].

Для вирішення вищезазначених проблем було запропоновано федеративне машинне навчання (FL). Архітектура промислових систем з використанням федеративного навчання представлена на Рис.3.4.

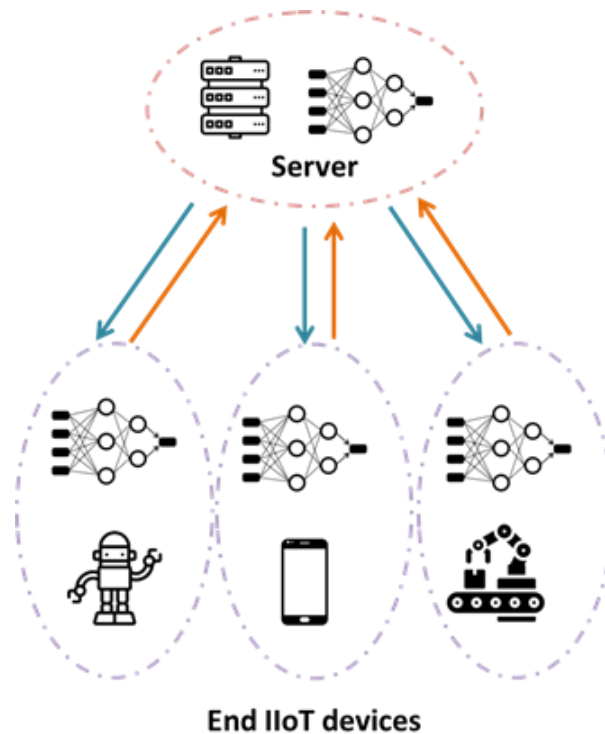


Рис. 3.4. Архітектура промислових систем з використанням федеративного навчання

Особливістю FL є те, що всі дані обробляються безпосередньо на кінцевих пристроях за допомогою локальних моделей ML. Потім лише результати навчання передаються на центральний пристрій для оновлення глобальної моделі, яка потім надсилається на всі пристрої. Таким чином, можна обмінюватися результатами навчання між різними вузлами без надсилання приватних даних користувача. Це також дозволяє передавати менші обсяги даних.

Алгоритм Singular Value Decomposition пропонується використовувати для оптимізації великих даних у промисловому Інтернеті речей. SVD дозволяє розкласти матрицю даних на добутки трьох підматриць і скоротити їх діапазон. Після перемноження цих підматриць отримуємо матрицю меншого розміру, яка з високою точністю повторює вихідну (Рис.3.5).

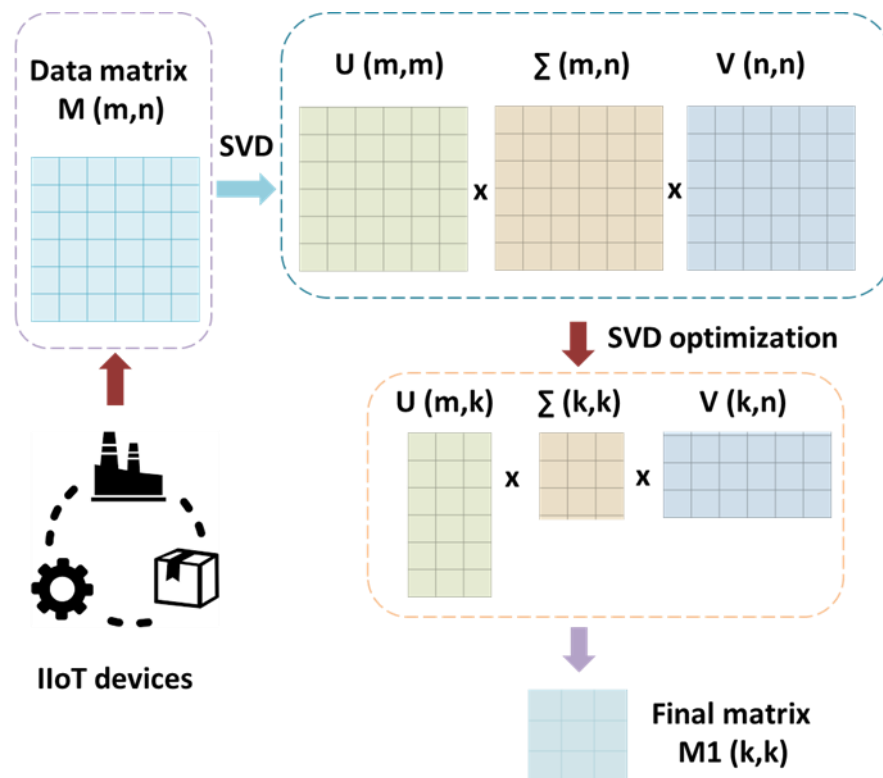


Рис.3.5.Алгоритм SVD в системах IIoT для оптимізації даних

Визначення важливих даних і видалення зайвих допомагає пришвидшити роботу алгоритмів машинного навчання та надати своєчасні результати, які можна використовувати для керування системою промислового Інтернету речей. Експериментальне дослідження ефективності SVD для розподілених промислових систем представлено в наступному розділі.

3.4. Модель для обробки даних великих обсягів в IoT системах

Завдяки використанню алгоритмів оптимізації даних можна вибрати з безлічі даних, які надсилаються різними пристроями, найбільш важливі для певного алгоритму. Метод Singular Value Decomposition, описаний у попередньому розділі, дозволяє зменшити масиви інформації, зберігаючи основні властивості. У роботі моделюється робота машини та алгоритмів федеративного навчання з використанням мови програмування Python [122,123]. Також перевірено переваги використання алгоритму SVD для зменшення розмірності даних і визначено, як змінюється тривалість обробки алгоритмів машинного навчання без і з різними параметрами оптимізації.

На Рис.3.6 показано результат роботи алгоритмів ML і FL при обробці однакової кількості даних і виконанні завдання сортування даних. Через те, що федеративне навчання передбачає реалізацію навчання на кінцевих вузлах, його розрахунок займає більше часу.

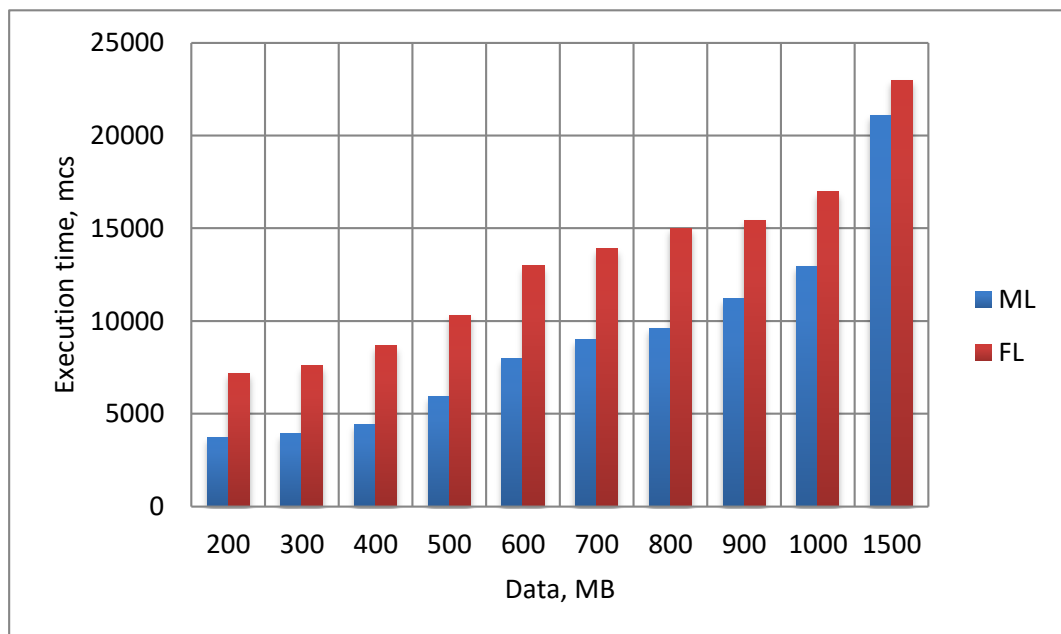


Рис.3.6.Залежності часу виконання для алгоритмів машинного та об'єднаного навчання від обсягу обчислювальних даних

Ми виконаємо попередню оптимізацію даних за допомогою алгоритму Singular Value Decomposition. Змінюючи кількість l відхилених сингулярних значень у розкладеній матриці, можна зменшити кількість даних, які потім обробляються алгоритмами машинного навчання. На Рис.3.7 показано реалізацію алгоритму федеративного навчання з використанням оптимізації.

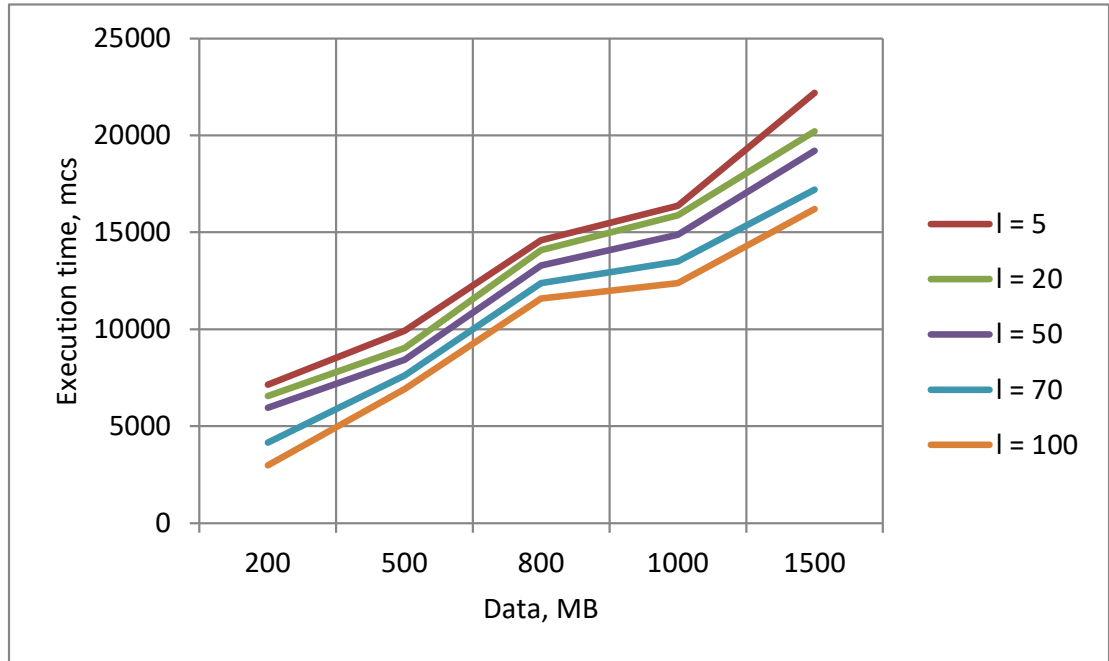


Рис.3.7.Залежність часу виконання для алгоритмів федеративного навчання від кількості відкинутих сингулярних чисел

Дослідження точності розрахунків після запропонованого методу оптимізації показано на Рис.3.8.

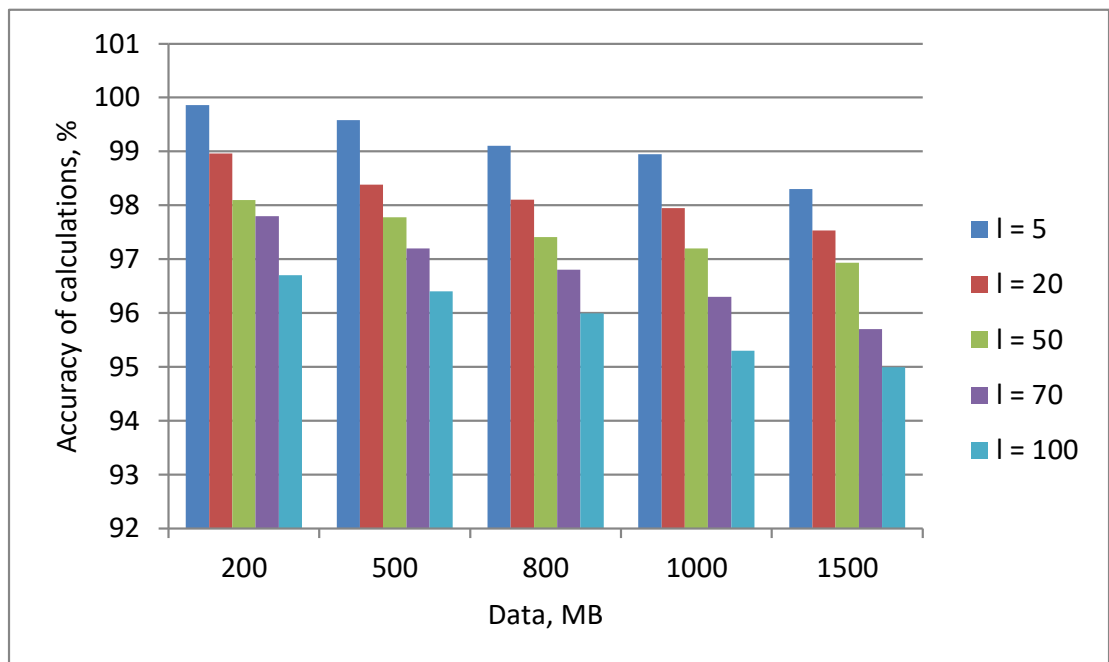


Рис.3.8.Залежності точності обчислень для алгоритмів федеративного навчання від даних

Як бачимо з результатів, особливості промислового Інтернету речей вимагають використання спеціальних розподілених алгоритмів обробки даних. Одним із таких методів є об'єднане навчання, яке дозволяє виконувати алгоритми машинного навчання безпосередньо на нетермінальних пристроях.

Оскільки обчислення з інтегрованим навчанням займають більше часу, запропоноване використання методу оптимізації SVD дозволяє виконувати обчислення швидше та з вищою точністю. Для використання в розподілених промислових системах алгоритми федеративного навчання часто незручні у реалізації та потребують додаткових витрат.

Раніше був запропонований модифікований алгоритм FedSVD для виконання розрахунку згаданого вище алгоритму SVD, враховуючи, що матриця даних складається з інформації, що належить різним користувачам [124]. Робота об'єднаного методу Singular Data Decomposition полягає в тому, що власник даних може використовувати певні випадкові числа для виконання математичних операцій. В результаті розрахунків на основі вихідних даних і даних для маскування виходить нова інформація, яка може бути передана по каналах зв'язку. Процес обчислення об'єднаного методу декомпозиції сингулярного значення показано на Рис.3.9.

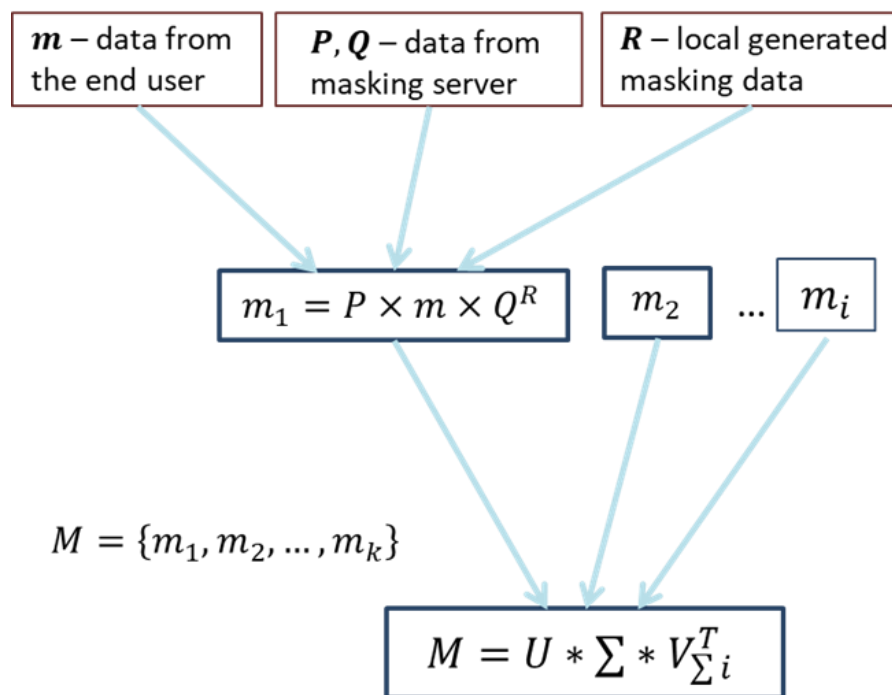


Рис.3.9. Розподілений розрахунок SVD

Навіть якщо приватні дані витікають третій стороні, це неможливо буде зрозуміти, не знаючи інформації про маскування. Цей підхід особливо ефективний у промислових системах Інтернету речей, оскільки кінцеві

користувачі, які використовують власні пристрої, тепер активно залучені до виробничого процесу. Використання таких пристроїв, як персональні комп'ютери, планшети та смартфони, може допомогти отримати інформацію з різних частин промислової системи та проаналізувати більш важливі фактори. Але також зростає ризик того, що дані, якими користувач не хоче ділитися, можуть бути вкрадені або пошкоджені [27].

Відповідно, для забезпечення конфіденційності використовується додатковий ступінь захисту даних на кінцевому пристрої, коли до локальних даних додається спеціальна маска, яка відома лише власнику даних. Тому при передачі даних на центральний пристрій без знання інформації про маскування особисті дані будуть незрозумілі стороннім.

Рис.9 пояснює принцип роботи алгоритму федеративного SVD. Локальні дані користувача обробляються перед відправкою на центральний пристрій. Числа P, Q генеруються випадковим чином і надсилаються на кілька обраних вузлів спеціальним сервером. Зв'язок між кінцевими користувачами та сервером має бути надійно захищений, оскільки він є вразливим місцем для можливих зловмисників.

Переваги об'єднаного алгоритму декомпозиції сингулярного значення:

- конфіденційність даних користувача;
- можливість збору даних з різних пристроїв;
- простота і надійність алгоритму.

Алгоритм FedSVD також має певні особливості. Сервер, який генерує випадкові числа, віддалений від кінцевого користувача. Під час передачі дані маскуванню можуть бути перехоплені. FedSVD також вимагає додаткового часу для генерації випадкових значень, додавання та видалення їх до даних користувача. Тому в цій статті ми пропонуємо генерувати всі випадкові числа, які використовуються в подальших обчисленнях, безпосередньо на локальному вузлі власника даних. Пропонована схема представлена на Рис.3.10.

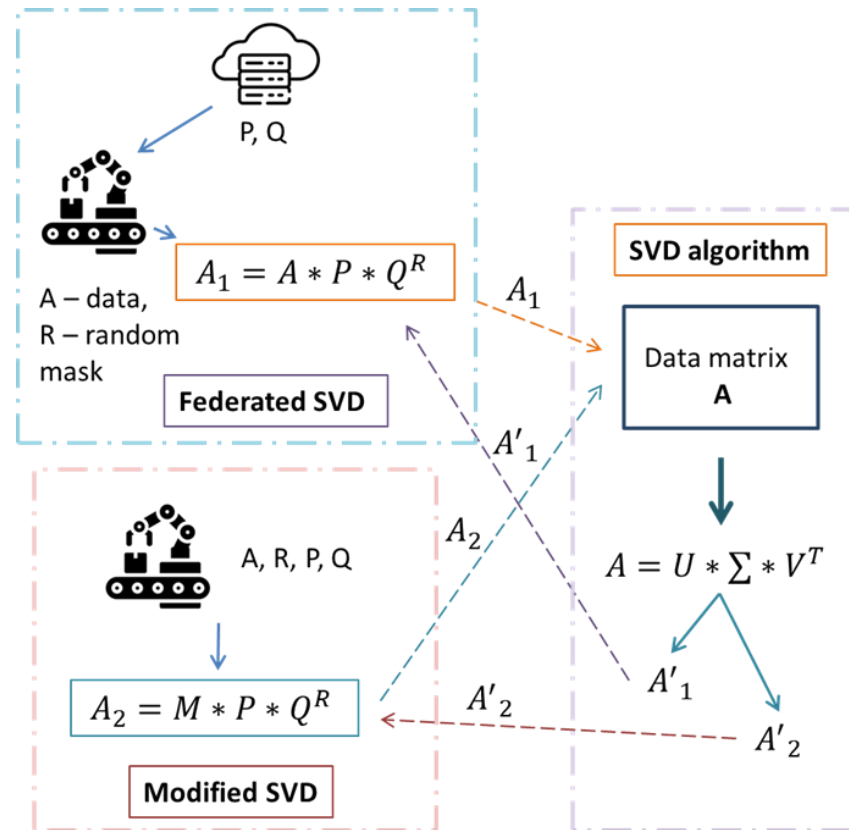


Рис.3.10.Схеми об'єднаних і модифікованих методів об'єднаних SVD, що використовуються в системах промислового Інтернету речей

Як бачимо з Рис.3.10, спрощення системи попередньої обробки дозволяє використовувати лише ресурси локального пристрою. У системах промислового Інтернету речей ця схема передбачає використання кінцевих або периферійних пристроїв для роботи з їхніми даними та захисту їх від пошкодження чи перехоплення. Недоліком запропонованої об'єднаної схеми SVD може бути недостатня обчислювальна потужність локальних пристроїв для генерації випадкових значень. У системах ПоТ використовуються пристрої різної потужності. Для ефективної обробки персональних даних і їх самозахисту потрібна відносно висока потужність пристрою. Поки що криптографічна стабільність випадкових значень, згенерованих кінцевим пристроєм, у статті не досліджувалася. Тому запропонована схема декомпозиції сингулярного значення є досить зручною для промислових систем, де необхідно спростити та прискорити процес генерації випадкових даних.

Розширення функцій кінцевих пристроїв може підвищити ефективність обробки інформації та конфіденційність персональних даних. Замість

делегування операцій над даними користувача центральним вузлам можлива їх локальна обробка, що забезпечує при дотриманні певних умов менший ризик конфіденційності та кращу швидкість обчислень.

На Рис.3.11 наведено результати обробки даних за допомогою алгоритму SVD та його модифікацій.

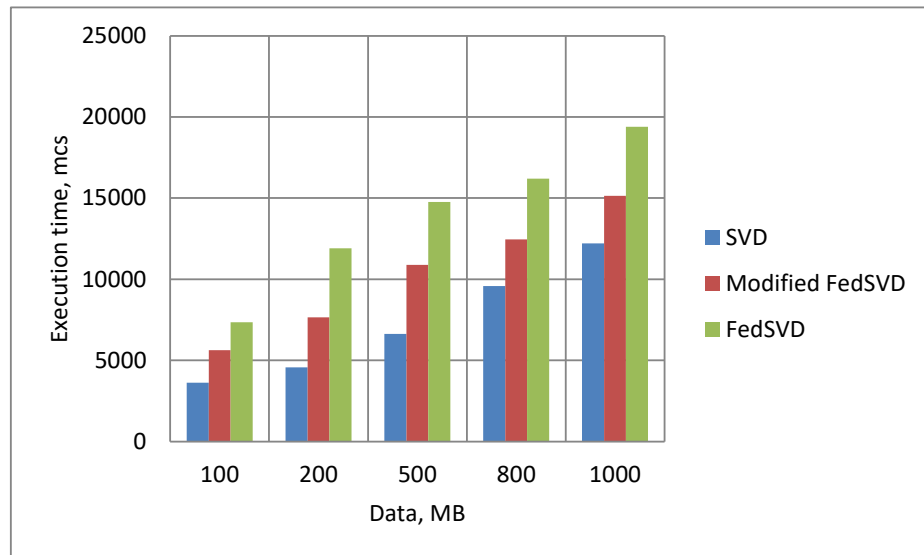


Рис.3.11. Залежності часу виконання алгоритмів СВД від обсягу даних

З Рис.3.11 зрозуміло, що FedSVD працює повільніше, оскільки вимагає додаткових ресурсів для маскуванню даних користувача порівняно з SVD. Натомість модифікований SVD обробляє дані трохи швидше, оскільки йому не потрібно спілкуватися з сервером. Визначимо, як використання різних алгоритмів декомпозиції сингулярного значення впливає на точність обробки даних (Рис.3.12). Використовувалися різні сингулярні числа матриці Σ , які використовуються для встановлення точності обчислень, згідно Рис.3.5.

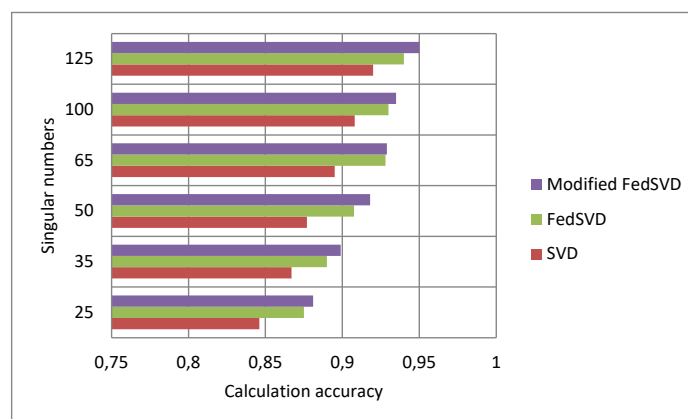


Рис.3.12. Залежності точності обчислень для алгоритмів СВД від сингулярних чисел

З Рис.3.12 можна зробити висновок, що об'єднаний метод сингулярної декомпозиції та його запропонована модифікація працюють точніше, ніж звичайний SVD. FedSVD, розглянутий у дослідженні, не належить до федеративних методів навчання, оскільки він збирав дані з цифрових пристроїв і обчислював на центральному комп'ютері. Натомість за попередню обробку та захист приватної інформації відповідає власник даних. При використанні простого збору даних, як ми бачимо з Рис.3.12, інформація користувача може оброблятися з меншою точністю в цілому. Під час надсилання результатів на кінцевий пристрій можуть виникати помилки. Об'єднаний SVD з використанням маскуванню даних дозволяє захистити і відновити результат.

Системи рекомендацій працюють з матрицями даних, які показують існуючі зв'язки між продуктами та користувачами, подіями тощо. На їх основі визначаються найбільш ймовірні майбутні події. Алгоритм SVD, розглянутий вище, який розкладає матрицю M на три підматриці U, Σ, V та визначає невідомий елемент $m_{i,j}$ наступним чином:

$$M_{i,j} = U_{i,j} \times V_{i,j} + b, \quad (3.1)$$

де b – коефіцієнт прогнозування, вибраний відповідно до параметрів попереднього навчання системи рекомендацій або випадковим чином.

Проблемою великомасштабних систем рекомендацій часто є розрідженість матриць даних. Тобто не всі комірки заповнені через відсутність інформації про відповідність між даними. Прогнозування на основі таких матриць ускладнює роботу алгоритму SVD. Також на зберігання таких таблиць виділяється багато ресурсів зберігання інформації. Для вирішення цих задач використовується алгоритм Funk-SVD [4]. Його особливістю є те, що дані $M(m, n)$ розкладаються на добуток двох матриць $P(\theta, m)$ і $Q(\theta, n)$, де θ визначає точність розкладання:

$$M = P^T \times Q. \quad (3.2)$$

Для визначення невідомих елементів у матриці M спочатку вибирається елемент $m_{i,j} = m_{const}$. Потім, використовуючи графік Funk-SVD, можна розрахувати прогнозоване значення:

$$m_{predicted} = \sum_{\theta} (p_{i,\theta}^T \times q_{j,\theta}). \quad (3.3)$$

Значення похибки передбачення:

$$N_{error} = (m_{const} + m_{predicted})^2. \quad (3.4)$$

Тепер будь-який невідомий елемент матриці M можна визначити за рівнянням:

$$m_{i,j} = p_{random} + 2 \times \sigma \times (m_{const} - m_{predicted}) \times q_{random}, \quad (3.5)$$

або:

$$m_{i,j} = q_{random} + 2 \times \sigma \times (m_{const} - m_{predicted}) \times p_{random}, \quad (3.6)$$

де:

q_{random}, p_{random} – випадкові значення з матриць P^T та Q ,

σ – коефіцієнт ефективності навчання.

Оскільки Funk-SVD дозволяє розкласти вхідну розріджену матрицю на дві менші, цей алгоритм працює трохи швидше, ніж стандартний SVD.

На Рис.3.13 показано порівняння тривалості розрахунку рекомендацій для алгоритмів SVD і Funk-SVD для матриць розріджених даних різного розміру.

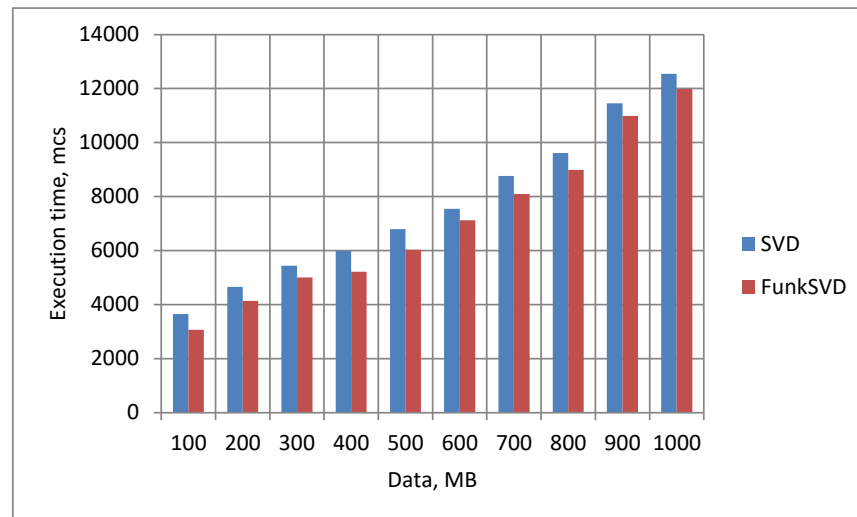


Рис.3.13. Залежності часу виконання для алгоритмів SVD і Funk-SVD

Як бачимо з Рис.3.13, тривалість обчислень можна зменшити за допомогою Funk-SVD. Коли підприємство збирає дані про різноманітні процеси, явища та зв'язки між ними, інформації часто бракує. Таким чином, використання Funk-SVD додатково дозволяє прискорити визначення необхідних даних та надання рекомендацій. На Рис.3.14 показано залежності індикатора середньоквадратичної помилки (Mean Squared Error, MSE) для двох

алгоритмів. MSE - це точність надання рекомендацій, тобто квадрат середнього відхилення між фактичним значенням і прогнозом:

$$MSE = \frac{(m_{i,j \text{ const}} - m_{i,j \text{ predicted}})^2}{k}, \quad (3.7)$$

де k - кількість передбачень.

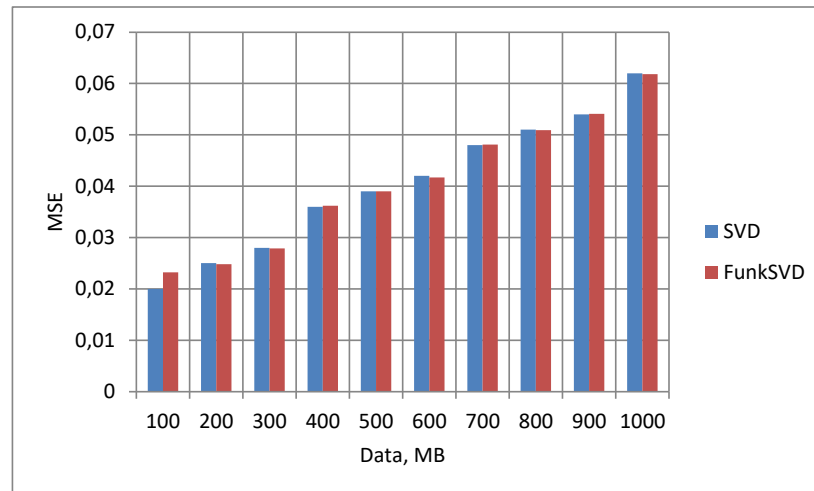


Рис.3.14. Залежності значення MSE для алгоритмів SVD і Funk-SVD

Як бачимо з Рис.3.14, алгоритми SVD і Funk-SVD працюють майже з однаковим значенням MSE. Таким чином, при використанні Funk-SVD можна скоротити час обчислень і обсяг пам'яті, необхідної для зберігання даних, зберігаючи при цьому точність надання рекомендацій. Коротко підсумовуючи, у статті спочатку досліджували ефективність використання алгоритму SVD для зменшення обсягу даних, що надходять від кінцевих користувачів алгоритму федеративного навчання. Результати на Рис.3.6-Рис.3.8 показують, що алгоритм SVD дозволяє відкинути частину даних, які не є важливими для подальшого аналізу, і зменшує тривалість обчислень. Ми можемо налаштувати параметри сингулярного розкладання, наприклад сингулярні значення, щоб досягти кращого балансу між точністю та швидкістю обчислень. У цій частині дослідження SVD виконувався лише на стороні користувача, щоб зменшити обсяг даних, потім був виконаний алгоритм федеративного навчання для пошуку вибраних значень в інформаційному масиві.

Також було досліджено обробку центральним пристроєм алгоритму SVD на основі інформації від різних користувачів (Рис.3.11 та Рис.3.12). Оскільки часто важко обчислити складні алгоритми, такі як SVD, за допомогою

федеративного навчання, використовується FedSVD, який дозволяє надсилати дані на центральний пристрій для обробки, попередньо замасковані. Дослідження також запропонувало змінити FedSVD і генерувати дані для маскуванню безпосередньо на локальному пристрої. Такий підхід підходить для випадків, коли пристрої мають достатню обчислювальну потужність, а система пред'являє високі вимоги до швидкості обробки даних. Точність розрахунків SVD в даному випадку - це різниця між вихідними даними користувача та даними, отриманими з центрального пристрою в результаті сингулярної декомпозиції. В результаті використання особистих даних маскуванню користувач може видалити маску та відновити дані з високою точністю. Використання алгоритму Funk-SVD (Рис.3.13 та Рис.3.14) дозволяє підвищити ефективність обробки розріджених даних в системах ІоТ. На основі Funk-SVD можна проводити контрольований алгоритм машинного навчання. Передаючи параметри точності рекомендацій між системами, можна прискорити процес навчання [124,125].

Висновки до третього розділу

У розділі було досліджено алгоритми розподіленого опрацювання даних, методи пошуку закономірностей і надлишковостей. Також було розглянуто підходи до пошуку відповідних товарів чи послуг для користувачів промислових систем, використовуючи методи машинного навчання. З урахуванням проаналізованого матеріалу було поставлено вимоги до створення та функціонування системи розподіленого оброблення користувацьких даних. Запропоновано алгоритми для оброблення великих даних в промислових ІоТ системах на основі Fed SVD та Funk SVD. Розроблено модель формування рекомендацій користувачам розподіленої промислової системи, що дозволяє проводити обмін результатами навчання між пристроями та використовує менше даних в порівнянні з існуючими. Дослідження ефективності запропонованих методів з наведено в наступному розділі. Результатом розділу є розв'язання задач № 4,5 дослідження.

РОЗДІЛ 4. ПРАКТИЧНІ АСПЕКТИ РЕАЛІЗАЦІЇ МОДЕЛІ РОЗПОДІЛЕНОЇ ОБРОБКИ ДАНИХ ВЕЛИКИХ ОБСЯГІВ.

У розділі визначено принципи роботи запропонованої моделі розподіленої обробки даних. Проведено опис особливостей проведеного експерименту. Наведено результати проведеного експерименту та практичні аспекти реалізації створеної моделі для обробки даних великих обсягів в ІоТ системах.

Результати розділу опубліковано в працях автора [2,4,5,8,10].

4.1. Принципи роботи запропонованої моделі розподіленої обробки інформаційно-комунікаційних систем промислової інфраструктури

Модернізація промислових систем внесла значні зміни в спосіб управління виробничими процесами. Концепція “Індустрія 4.0” уможливила оцифрування виробництва, що призвело до впровадження сучасних інформаційних систем, які вирішують кілька проблем для надання належної якості послуг. У промисловій системі необхідно вирішувати завдання збору даних з різних кінцевих пристроїв, зберігання та аналізу для подальшого прийняття рішень. Якість обслуговування користувачів є одним з найважливіших критеріїв оцінки ефективності промислових систем. Визначені характеристики надання послуг для одного користувача можуть бути використані для її покращення для інших. Чим більше джерел інформації аналізується, тим легше визначити загальні закономірності та можливі проблеми в роботі промислових систем. У той же час надлишок даних, які постійно надходять з різних кінцевих пристроїв, не завжди є критичним для обчислень. Необхідно використовувати методи аналізу та оптимізації даних [124,125].

Зберігання та обробка даних, отриманих з різних розподілених джерел, надзвичайно важливі для сучасних промислових систем. Хмарні технології використовуються для підвищення ефективності аналізу великих даних. Хмарне виробництво (SMfg) є важливою складовою концепції Industry 4.0 [3].

CMfg дозволяє промисловій системі контролювати свої ресурси та швидко отримувати до них доступ. Також користувачі можуть обмінюватися даними з іншими та глобальними вузлами управління, зберігаючи їх на хмарних серверах.

Системи рекомендацій широко використовуються в різних галузях промисловості, включаючи промислові мережі, для надання персоналізованих рекомендацій користувачам. У промислових мережах рекомендаційні системи можна використовувати, щоб рекомендувати підприємствам продукти, послуги та постачальників на основі їхніх минулих покупок, інтересів та поведінки. Загалом, рекомендаційні системи мають потенціал для надання значних переваг підприємствам у промислових мережах шляхом підвищення ефективності закупівель і скорочення часу та зусиль, необхідних для пошуку постачальників і продуктів. У промислових умовах системи рекомендацій можна використовувати різними способами, наприклад, рекомендувати графіки технічного обслуговування машин, оптимізувати виробничі процеси, рекомендувати постачальників сировини тощо. Завдяки високій пропускну здатності та низькій затримці мереж 5G ці системи можуть надавати рекомендації користувачам у реальному часі, дозволяючи їм приймати рішення швидко та ефективно. Однак впровадження систем рекомендацій у цьому контексті вимагає ретельного розгляду унікальних проблем і обмежень промислових мереж. Оскільки більшість кінцевих пристроїв промислового Інтернету речей (IIoT) розташовані віддалено один від одного, дані передаються через бездротові канали зв'язку. Системи мобільного зв'язку 5G дозволяють швидко та ефективно обслуговувати користувачів, надаючи їм широкий спектр послуг зв'язку [126,127,128]. Однак необхідно вирішити проблему передачі даних користувачам, оптимізації даних, розподілу ресурсів і навантаження на систему.

При розробці рекомендаційної системи також важливо враховувати конкретні потреби промислового середовища. Це може включати ідентифікацію найбільш релевантних джерел даних, вибір відповідних

алгоритмів для аналізу даних і включення предметно-специфічних знань у систему рекомендацій. У промислових мережах 5G може бути велика кількість користувачів і елементів, що може призвести до розрідження даних. Розріджені дані стосуються ситуацій, коли мало оцінок або взаємодії між користувачами та елементами. Через це може бути складно створити точні рекомендації за допомогою традиційних алгоритмів спільної фільтрації.

Збір даних про статистику продажів і використання послуг різними категоріями користувачів дозволяє нам краще зрозуміти їхні потреби та адаптувати до них виробничі процеси. Перелік товарів і послуг постійно розширюється, а вимоги до якості обслуговування зростають, тому необхідно постійно шукати нові методи обробки даних.

Новизна даної роботи полягає в пропозиції модифікацій алгоритму Funk SVD для обробки великих даних у промислових умовах з метою розробки ефективних промислових рекомендаційних систем 5G. Основні внески дослідження подвійні. По-перше, ми пропонуємо використовувати лише частину даних користувача для формування рекомендацій, що скорочує тривалість розрахунків і підтримує високий рівень точності обробки даних. По-друге, ми демонструємо ефективність включення додаткових функцій елементів для підвищення точності систем рекомендацій.

Загалом запропоновані методи оптимізації даних можуть революціонізувати роботу промислових мереж 5G, дозволяючи розробляти більш точні та швидкі системи рекомендацій. Результати дослідження підкреслюють перспективу використання модифікованих алгоритмів FunkSVD для підвищення якості обслуговування для різних типів користувачів систем 5G, що призведе до підвищення ефективності, зниження витрат і оптимізації процесів [129,130,131,132].

Проблеми та рішення для систем промислових рекомендацій 5G

Промислові рекомендаційні системи 5G мають потенціал кардинально змінити спосіб роботи фабрик і промислових підприємств, забезпечуючи збір і аналіз даних у режимі реального часу, прогнозне технічне обслуговування та

оптимізовані виробничі процеси. Загалом, успішне впровадження промислових систем рекомендацій 5G вимагає поєднання передових технологій, безпечних протоколів передачі даних і ефективних алгоритмів обробки та аналізу даних.

Для ефективної роботи алгоритмів ML необхідно використовувати великі обсяги інформації. Розвиток інформаційних технологій і стрімка цифровізація сприяють щоденній генерації значних наборів даних з різних джерел. Завдяки таким даним можна краще виявити особливості, характерні для окремих завдань, для точного визначення результату. Однак надлишок інформації все одно ускладнює роботу обчислювальних алгоритмів. Обробка і зберігання великих масивів даних призводить до споживання обчислювальних ресурсів і уповільнення обчислень. Різноманітність джерел інформації також вимагає адаптації алгоритмів ML і розподіленої обробки даних.

Федеративне навчання є одним із методів ефективної організації розподіленої обробки даних за допомогою ML. Замість того, щоб надсилати всі дані з різних розподілених вузлів один одному, вони обробляються на локальних вузлах. Тільки результати обчислень надсилаються на контролер, який оновлює всю систему машинного навчання. Федеративне навчання ефективно використовується в промислових системах, оскільки вирішує проблему обробки даних, зібраних з різних кінцевих пристроїв. Це покращує надійність і конфіденційність користувачів' приватна інформація, оскільки її не потрібно передавати по мережі.

Статистичний аналіз надзвичайно важливий для роботи комерційних систем, які надають користувачам певні товари чи послуги. За попередніми діями клієнтів можна визначити, наскільки вони задоволені послугами, якими користуються. Системи рекомендацій (RS) використовуються для встановлення зв'язків між користувачами та товарами чи послугами, які зазвичай називають елементами. У повсякденному житті ми часто бачимо приклади систем рекомендацій. Під час відвідування веб-сайтів або додатків користувачам пропонується реклама товарів, які можуть їх певним чином зацікавити. Наприклад, при пошуку певного об'єкта на сайті система рекомендацій також

визначає схожі за певними параметрами. Подібні продукти чи послуги позначаються як «це може вам сподобатися», «інші продукти, які вам можуть сподобатися» тощо. Існують різні підходи до визначення найбільш прийнятних товарів для користувачів. Рекомендації можуть бути персоналізовані, тобто визначатися з урахуванням особливостей конкретного користувача. Також можна розрахувати неперсоналізовані рекомендації, загальні для певної групи користувачів [133,134,135].

4.2. Опис особливостей проведеного експерименту

Для роботи рекомендаційних систем використовуються спеціальні методи обробки даних від користувачів. Результатом RS є пропозиції користувачам щодо нових продуктів або послуг, які їм, швидше за все, сподобаються. Найпоширенішими підходами до роботи рекомендаційних систем є:

- На основі вмісту, який аналізує схожість різних продуктів один з одним. Якщо користувачеві раніше сподобався певний продукт або послуга, велика ймовірність того, що він високо оцінить подібні в майбутньому. Перевагою такого підходу є простота реалізації. Недоліком є те, що нові та несхожі на інші продукти мають мало шансів потрапити до рекомендацій;
- Спільна фільтрація формує рекомендації, аналізуючи профілі користувачів. У разі виявлення подібності між двома або більше користувачами, їм рекомендуються продукти або послуги, раніше обрані іншими учасниками групи. Такий підхід демонструє високу ефективність. При цьому необхідно заздалегідь підготувати рекомендаційну систему, надавши інформацію про користувачів та їхні вподобання. За відсутності інформації виникають проблеми у формуванні рекомендацій;
- Гібридний підхід до формування рекомендацій, що поєднує вищезазначене [39].

Інтелектуальні виробничі системи зосереджені не лише на виробництві певної продукції, але й на її продажу кінцевим споживачам. Визначення рівня інтересу до певних товарів чи послуг надзвичайно важливо для ефективного

функціонування системи. Часто виникає ситуація, коли товар не користується попитом у користувачів через свої особливості. В результаті моніторингу відгуків і статистики продажів можна визначити й негайно впровадити можливі модифікації та вдосконалення послуг і товарів. Автоматизація управління інтелектуальною виробничою системою сприяє оперативному вирішенню існуючих проблем і запобіганню виникненню нових. Бездротова мережа 5G є важливою складовою інтелектуального виробництва.

Оскільки багато кінцевих пристроїв IoT знаходяться на значній відстані один від одного, бездротовий зв'язок дозволяє об'єднати їх в єдину систему обміну даними. Кінцеві користувачі можуть підключатися до мережі, коли їм потрібно надіслати або отримати інформацію. Технологія 5G дозволяє надавати різноманітні послуги зв'язку на високій швидкості та забезпечувати необхідну якість обслуговування [136,137,138]. Однак великі обсяги даних, якими обмінюються користувачі та система, створюють значне навантаження на канали зв'язку та пристрої обробки. У результаті деякі послуги можуть бути нижчої якості або взагалі недоступні. Архітектура промислової системи, яка використовує для зв'язку мережу 5G, показана на Рис.4.1.

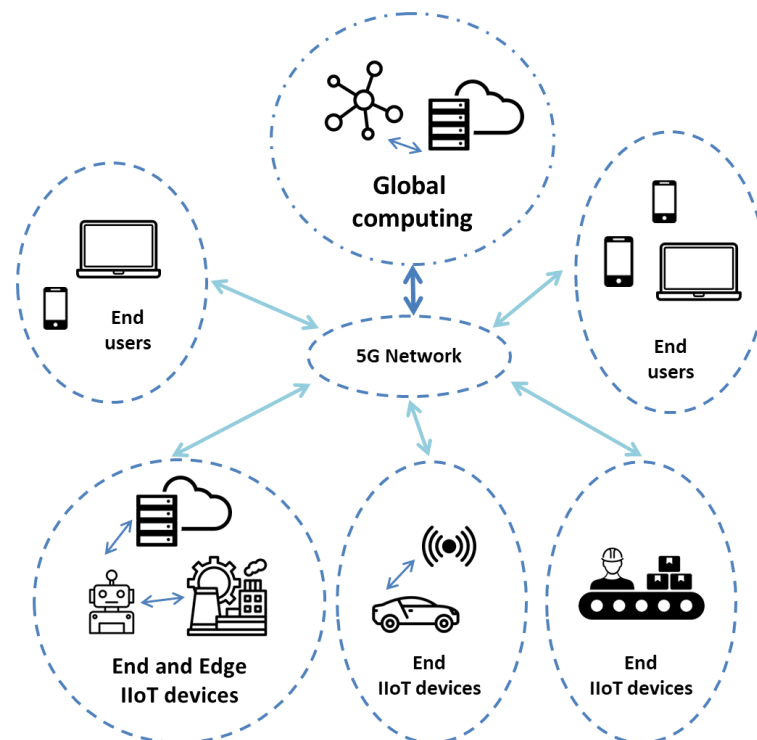


Рис.4.1. Архітектура промислової системи з використанням мережі 5G

Для оптимізації роботи мереж 5G використовуються методи ML і штучного інтелекту. Аналіз наданих послуг і зміни навантаження в залежності від різних параметрів допомагає краще налаштувати параметри системи. Методи ML дозволяють розподіляти ресурси мережі відповідно до потреб користувачів і швидко вирішувати проблемні ситуації. Таким чином, мобільна мережа може гнучко адаптуватися до різних умов роботи та визначати пріоритети в обслуговуванні користувачів. Визначивши зони системи мобільного зв'язку, які є найбільш завантаженими, можна автоматично залучати додаткові ресурси. Для цього здійснюється постійний моніторинг показників навантаження та продуктивності системи. Процес обробки запитів користувачів у системах 5G показано на Рис.4.2.

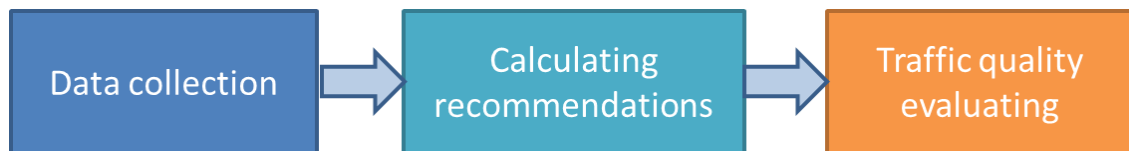


Рис.4.2. Процес обробки запитів користувачів у системах 5G

Різні типи трафіку, що надається користувачам систем 5G, також вимагають особливої уваги до надання пріоритетів послуг і розподілу необхідних комунікаційних ресурсів. Передача мультимедійних даних споживає більше енергії, ніж голосові повідомлення. Достовірність і конфіденційність даних користувачів також повинні бути забезпечені при передачі їх на обробку. Методи ML вирішують проблеми оптимізації, аналізу та прийняття рішень великих даних. Через велику кількість кінцевих користувачів алгоритми ML отримують багато інформації для навчання. Однак, оскільки дані часто містять надмірність і занадто громіздкі для обчислення, їх слід попередньо обробити та вибрати найбільш важливі.

Комплексне дослідження алгоритмів SVD і Funk SVD для оптимізації даних у промислових системах рекомендацій

Удосконалення промислових систем призвело до необхідності обробки величезних масивів даних. Інформація, що надходить з різних пристроїв, часто різного типу, неструктурована і має великий обсяг. Пристрої обробки великих

даних зазвичай не можуть обробляти їх ефективно, тому користувачі не отримують своєчасних результатів. Для більш швидкого обчислення дані слід попередньо підготувати та привести до чіткого та короткого вигляду. Оптимізація великих даних і відмова від надмірності є одним з найбільш актуальних для сучасних промислових систем [139,140]].

Для інтелектуальної обробки інформації використовуються методи ML, Deep Learning і Data Mining. Існує багато способів перетворити дані вищої розмірності в менші, зберігаючи їх властивості з певним рівнем точності. Наприклад, одним із найпопулярніших методів є аналіз головних компонент (principal component analysis, PCA), який вирішує проблему зменшення розмірності, зберігаючи якомога більше властивостей вихідних даних. PCA представляє послідовності даних як набір взаємозалежних змінних, головних компонентів (principal components, PC). Потім шукаються найбільш інформативні змінні, а неважливі відкидаються. Алгоритм латентного семантичного аналізу (latent semantic analysis, LSA) подібний до PCA, але він використовується більше для аналізу текстових послідовностей. Обидва методи можна розширити до алгоритму сингулярної декомпозиції значень (SVD), покращеного способу ідентифікації найважливіших даних і усунення надмірності.

SVD розкладає вихідну матрицю на добуток трьох взаємно ортогональних підматриць. Цей метод оптимальний для обробки та оптимізації матриць, які не є прямокутними. Порівняно з алгоритмом PCA, SVD більш ефективно визначає фундаментальні властивості в масивах даних і враховує їх різноманітність. Декомпозиція сингулярного значення передбачає представлення вихідної матриці даних A (m, n) у вигляді:

$$A = P \times \Delta \times Q, \quad (4/1)$$

де матриця P має розмірність (m, m) , матриця Q має розмірність (n, n) , матриця Δ - розмірність (m, n) і демонструє зв'язок між P і Q .

SVD широко використовується для рекомендаційних систем і неодноразово довела свою ефективність [27]. Однак у реальних системах, де обробляються великі дані з різних пристроїв, виникає кілька проблем. По-перше, інформацію необхідно привести у форму, придатну для формування рекомендацій. Таким чином, усі дані про взаємодію користувачів і елементів повинні бути представлені у вигляді числової оцінки. Для вирішення цієї проблеми використовуються методи попередньої обробки та оптимізації інформації. Важливо також враховувати, що тільки за ідеальних умов ми можемо отримати практично всі дані про оцінки кожного товару від усіх користувачів і сформувати матрицю рекомендацій. У такій таблиці рядки відповідають користувачам, стовпці — пунктам, а на перетині — рейтинги, тобто оцінки [141,142].

У реальних системах рекомендацій ми отримуємо лише частину інформації про користувачів' інтерес до певних товарів. Багато продуктів ще не отримали відгуків покупців, а нещодавно зареєстровані користувачі не мають історії покупок. Системи рекомендацій призначені для заповнення порожніх клітинок у таблицях рекомендацій, але також мають обробляти дані більш компактно та ефективніше. Таблиці рекомендацій зазвичай мають розміри (m,n) , де m — кількість користувачів, а n — елементи. Через розрідженість матриць даних немає сенсу обробляти їх усі, оскільки вони не несуть інформації для формування рекомендацій, але використовують багато обчислювальних ресурсів [143,144,145].

Для визначення найбільш релевантних послуг для конкретних категорій слід використовувати системи рекомендацій, які встановлюють відповідність між користувачами та продуктами. Таким чином, підвищується ефективність надання послуг. Рекомендаційні системи мають проблему обробки надзвичайно великих даних, які часто не є надзвичайно важливими для формування рекомендацій. Рекомендаційні матриці виглядають так:

$$A(n, m) = \begin{pmatrix} a_{11} & N/A & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & N/A \\ \dots & \dots & \dots & \dots \\ a_{n1} & N/A & \dots & a_{nm} \end{pmatrix}, \quad (4.2)$$

де a_{ij} значення взаємодії між користувачем i та елементом j , N/A – дані недоступні або невідомі.

Надлишок інформації призводить до уповільнення обчислень і перевантаження пристроїв. Розріджені матриці даних краще оптимізувати, відкидаючи порожні клітинки або ті, що містять неважливу інформацію. Для обробки розріджених даних у роботі пропонується вдосконалити існуючий алгоритм Funk SVD, який розкладає вихідну матрицю A на добуток двох підматриць [30, 31]:

$$A(n, m) = M(n, k) \times N(k, m)^T, \quad (4.3)$$

де $k < n$ і $k < m$.

Алгоритм FunkSVD використовує метод стохастичного градієнтного спуску для поступового зменшення похибки між вихідною матрицею та отриманою матрицею. Таким чином, можна визначити суму квадратів помилок для кожного розкладу:

$$\min \sum_{i \in n, j \in m} (a_{i,j} - m_i \times n_j^T)^2 + (|m_i|^2 + |n_j^T|^2), \quad (4.4)$$

де $a_{i,j}$ – елементи вихідної рекомендаційної матриці, n_i , m_j – елементи матриць N та M відповідно.

Значення елемента вихідної матриці $a'_{i,j}$ отримують шляхом перемноженням рядків і стовпців з відповідними індексами:

$$a'_{i,j} = \sum_{i,j=k} (m_i \times n_j^T), \quad (4.5)$$

Похибку розрахунку можна визначити наступним чином:

$$Err = a_{i,j} - a'_{i,j} \quad (4.6)$$

Ми можемо оновити елемент матриці M , використовуючи коефіцієнт навчання ε та поправковий коефіцієнт θ :

:

$$m'_i = m_i + \varepsilon(\text{Err} \times 2 \times n_j + \theta \times m_i), \quad (4.7)$$

Також обчислимо елемент матриці N :

$$n'_j = n_j + \varepsilon(\text{Err} \times 2 \times m_i + \theta \times n_j). \quad (4.8)$$

Тоді рекомендація для користувачів визначається як:

$$a'_{i,j} = m'_i \times n'_j{}^T + d_{i,j}, \quad (4.9)$$

$d_{i,j}$ – сумарний коефіцієнт відхилення показників користувача та товару від суми:

$$d_{i,j} = d_i + d_j + \Delta, \quad (4.10)$$

де d_i – відхилення від середнього значення для користувача, d_j – для елемента, Δ – коефіцієнт регуляризації.

Точність і ефективність системи рекомендацій можна визначити, представивши її матрицю плутанини (Таблиця 4.1):

Таблиця 4.1. Матриця плутанини для систем рекомендацій

	Користувач зацікавлений в товарі	Користувач не зацікавлений в товарі
Рекомендовано	$K_{true_positive}$	$K_{false_positive}$
Не рекомендовано	$K_{false_negative}$	$K_{true_negative}$

Згідно з Таблицею 4.1 $K_{true_positive}$ – ймовірність того, що рекомендований товар зацікавив користувача;

$K_{false_positive}$ відповідає ймовірності позитивної рекомендації щодо нецікавого товару;

$K_{false_negative}$ – ймовірність негативної рекомендації щодо цікавого товару;

$K_{true_negative}$ є імовірністю негативної рекомендації щодо нецікавого товару.

Імовірність надання позитивної рекомендації, якщо товар не цікавий користувачеві, можна представити як:

$$K_{true_positive_rate} = \frac{K_{true_positive}}{K_{true_positive} + K_{false_negative}}, \quad (4.11)$$

Імовірність дати негативну рекомендацію, якщо продукт не цікавий користувачеві:

$$K_{true_negative_rate} = \frac{K_{true_negative}}{K_{true_negative} + K_{false_positive}}, \quad (4.12)$$

Точність надання рекомендацій можна розрахувати таким чином:

$$K_{positive_predictive_value} = \frac{K_{true_positive}}{K_{true_positive} + K_{false_positive}}. \quad (4.13)$$

Однак, щоб оцінити ефективність алгоритму FunkSVD, рекомендації можна безпосередньо визначити середню абсолютну похибку (Mean Absolute Error, MAE) для матриці A з розмірністю N :

$$K_{mean_absolute_error} = \frac{\sum_{i,j=1}^N |a_{i,j} - a'_{i,j}|}{|N|}. \quad (4.14)$$

У роботі використовується середня квадратична помилка (Root Mean Square Error, RMS Error, RMSE) для перевірки точності розрахунків рекомендацій:

$$K_{rmse} (Accuracy) = \sqrt{\frac{\sum_{i,j=1}^N (a_{i,j} - a'_{i,j})^2}{|N|}}. \quad (4.15)$$

4.3. Результати проведеного експерименту

Для дослідження ефективності обробки даних алгоритмом FunkSVD створено програмну модель на мові програмування Python, використано дані відкритої рекомендаційної системи. Для кращого розуміння роботи алгоритму були витягнуті дані різного обсягу та з різним рівнем розрідженості.

Порівняння тривалості обчислення даних алгоритмами SVD та Funk SVD показано на Рис.4.3. Досліджувалися матриці даних з різним ступенем розрідженості, тобто комірки, заповнені інформацією.

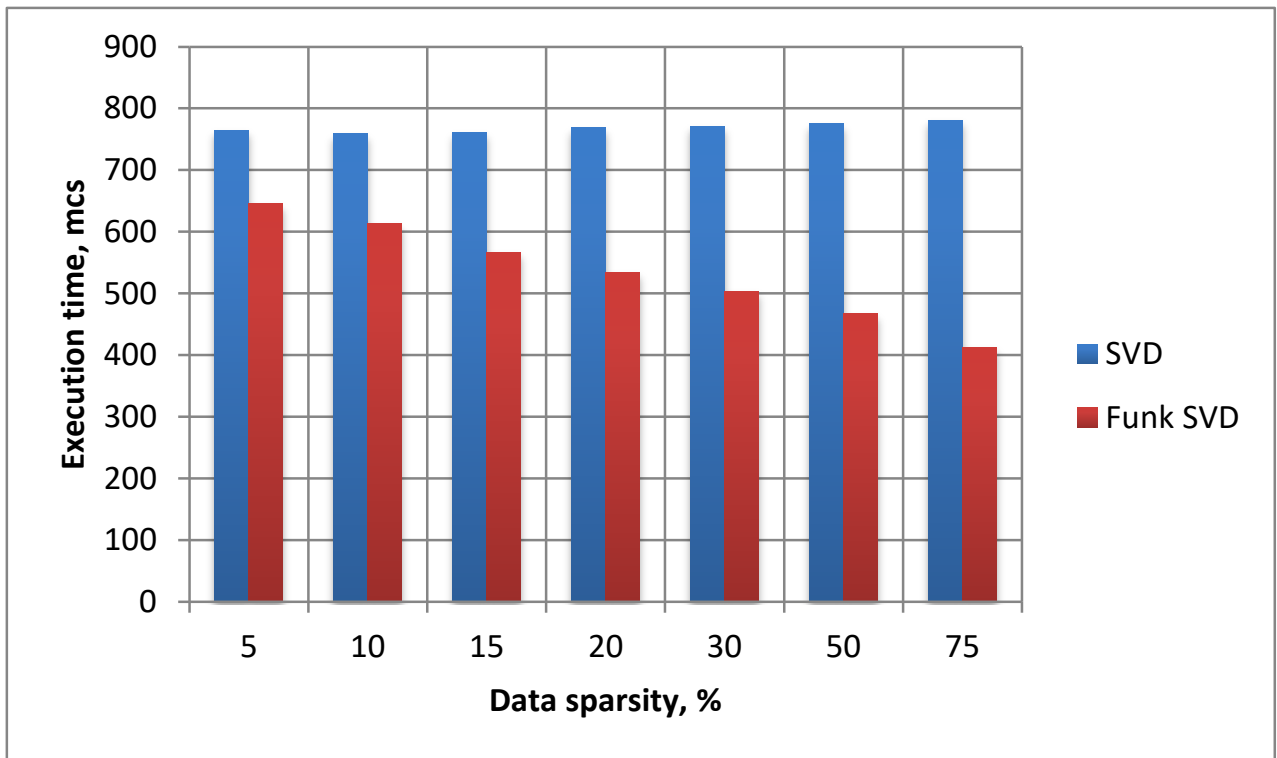


Рис.4.3. Порівняння часу виконання алгоритмами SVD і Funk SVD

Як бачимо з Рис.4.3, SVD працює довше, тоді як різниця з Funk SVD збільшується зі збільшенням відсотка розрідженості даних. Якщо система обробляє багато зайвих даних, Funk SVD дозволяє відкинути їх частину та прискорити обчислення.

Модифікований підхід Funk SVD для покращення часу виконання систем рекомендацій

Хоча FunkSVD досить добре працює з розрідженими даними, його все одно можна вдосконалити, щоб прискорити обчислення. Відповідно до ур. (5) алгоритм Funk SVD використовує дані про всіх користувачів і продукти для визначення невідомого значення комірки початкової матриці даних. У статті ми запропонували використовувати різну кількість користувачів δ у розрахунках для модифікації існуючого алгоритму Funk SVD:

:

$$a'_{i,j} = \sum_{i=\delta, j=k} (m_i \times n_j^T),$$

де $\delta < k$.

Кількість користувачів, чиї дані враховуються в розрахунках, вибирається довільно і може бути змінена для досягнення кращого результату. Таким чином, можна формувати рекомендації шляхом обробки меншої кількості даних, що, безсумнівно, прискорить роботу рекомендаційних систем. Результати порівняння тривалостей розрахунку наведені на Рис.4.4.

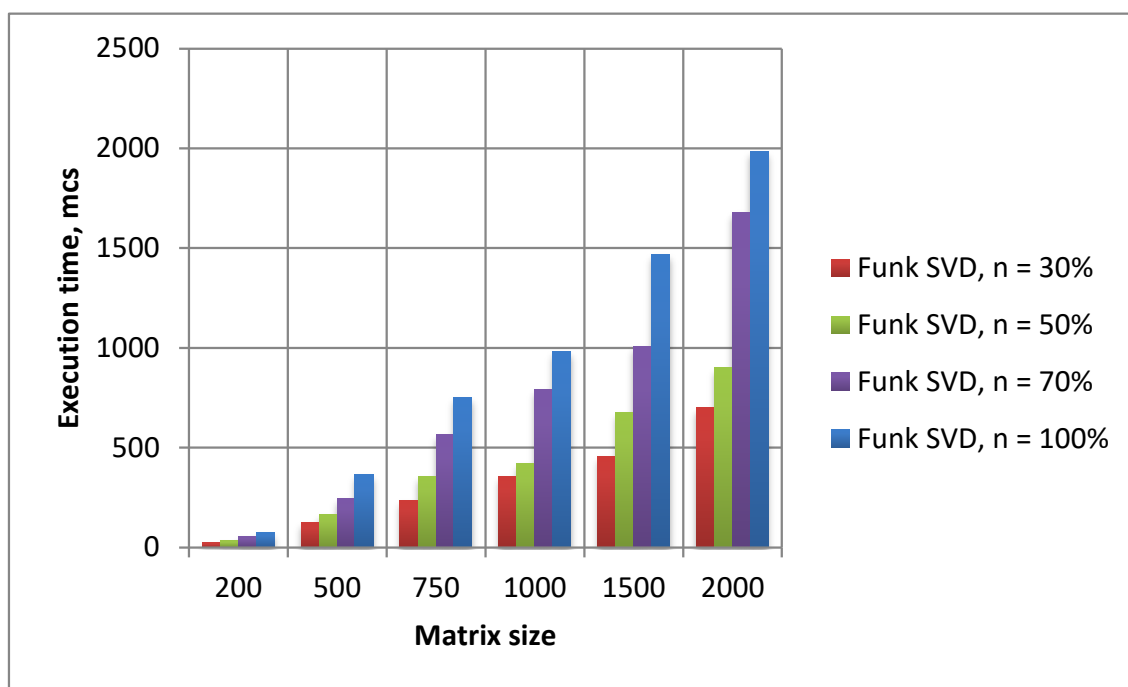


Рис.4.4. Порівняння часу виконання за алгоритмом Funk SVD з використанням різної кількості користувачів для надання рекомендацій

Відповідно до Рис.4.4 можна спостерігати закономірність прискорення обчислень при використанні меншої кількості інформації про користувачів. Тепер давайте порівняємо розрахунки' точність, тобто різниця між вихідною матрицею даних і відновленою після декомпозиції (Рис.4.5).

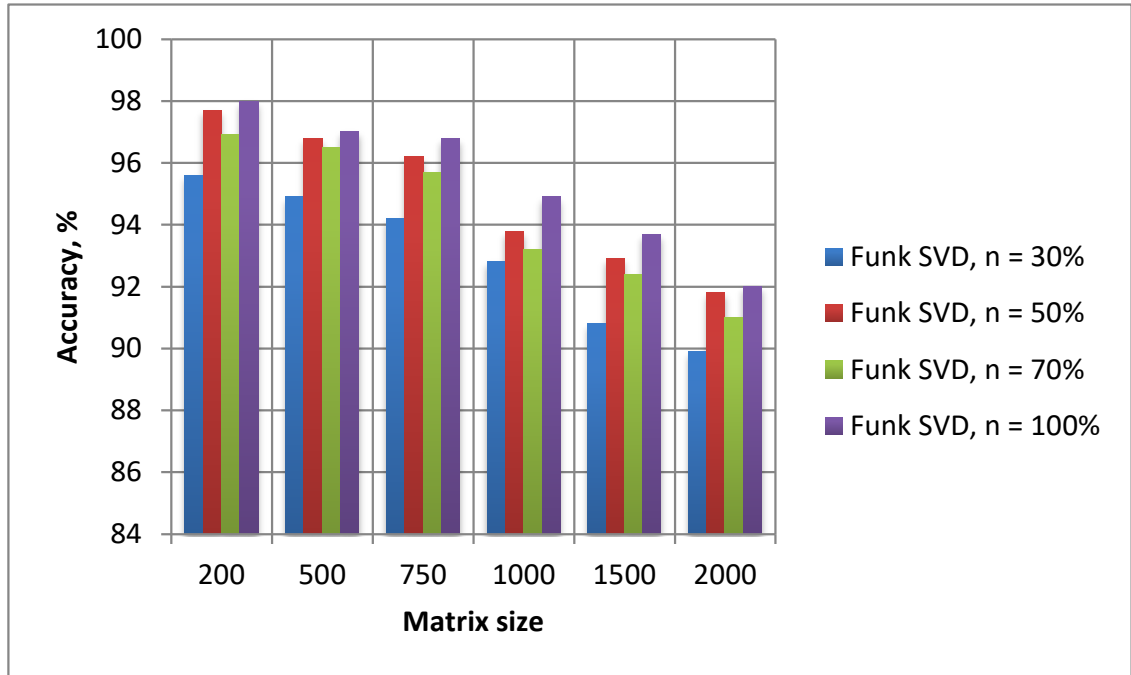


Рис.4.5. Порівняння точності за алгоритмом Funk SVD з використанням різної кількості користувачів для надання рекомендацій

Результати досліджень показують незначне погіршення точності розрахунків при використанні меншої кількості користувачів. Такий підхід дозволяє вибирати різні системи' оптимальні параметри обробки даних. Ми можемо залучати менше даних користувачів, якщо потрібно швидко обробляти інформацію. Більше даних можна використовувати для надання рекомендацій у щойно створених або динамічно оновлюваних системах [146,147].

Підвищення точності систем рекомендацій у промислових системах допомогою модифікованого алгоритму Funk SVD

Наступною модифікацією алгоритму FunkSVD, запропонованою в цій роботі, є підвищення точності рекомендацій. Для підвищення точності розрахунків алгоритму Funk SVD ми пропонуємо використовувати більше можливостей щодо позицій, які додаються до матриці товарів після розмітки стандартним алгоритмом. Такими характеристиками є дані про залежності елементів і користувачів, які впливають на точність формування рекомендацій. У системах 5G можна збирати додаткові дані про групи користувачів, їх взаємодію з товарами та особливості, які цікавлять певний сервіс. У початковій матриці рекомендацій, яка містить лише оцінки продукту, таких ознак немає.

Натомість ми можемо використовувати їх для покращення якості надання рекомендацій, якщо така потреба є (Рис.4.6).

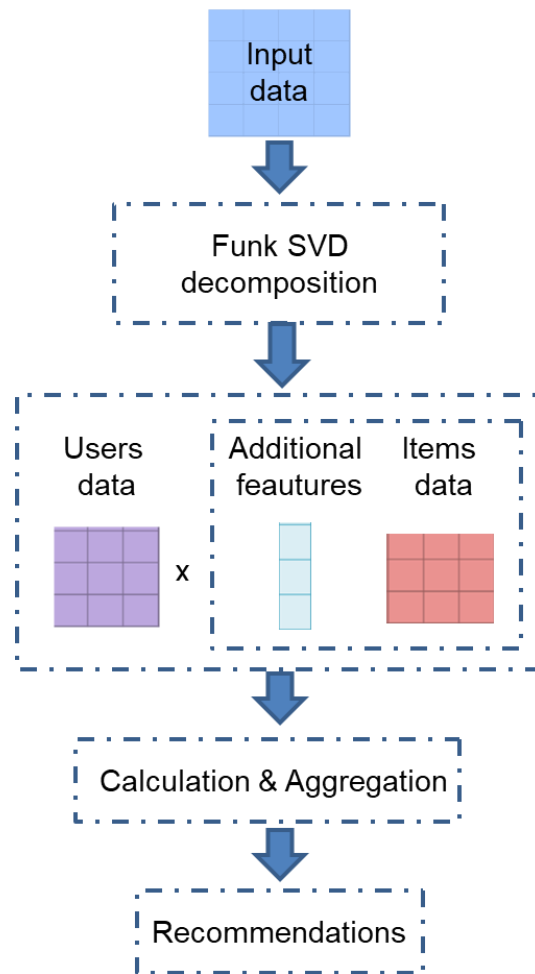


Рис.4.6. Схема використання додаткових можливостей про елементи для модифікованого алгоритму Funk SVD

Таким чином, щоб розрахувати рекомендацію користувачеві щодо певного продукту, ми формуємо матрицю нових елементів N_{mod} після декомпозиції FunkSVD, яка містить додаткові функції:

$$a'_{i,j} = \sum_{i,j=k} (m_i \times n_{j_{mod}}^T), \quad (4.17)$$

У системах мобільного зв'язку 5G користувачі поділяються на групи. Наприклад, надаються спеціальні послуги для потреб бізнесу, збору додаткової статистики. Таким клієнтам потрібні якісні рекомендації відповідно до індивідуальних особливостей. Ми провели дослідження продуктивності запропонованого алгоритму. Порівняння тривалості розрахунку для немодифікованого та модифікованого алгоритмів показано на Рис.4.7.

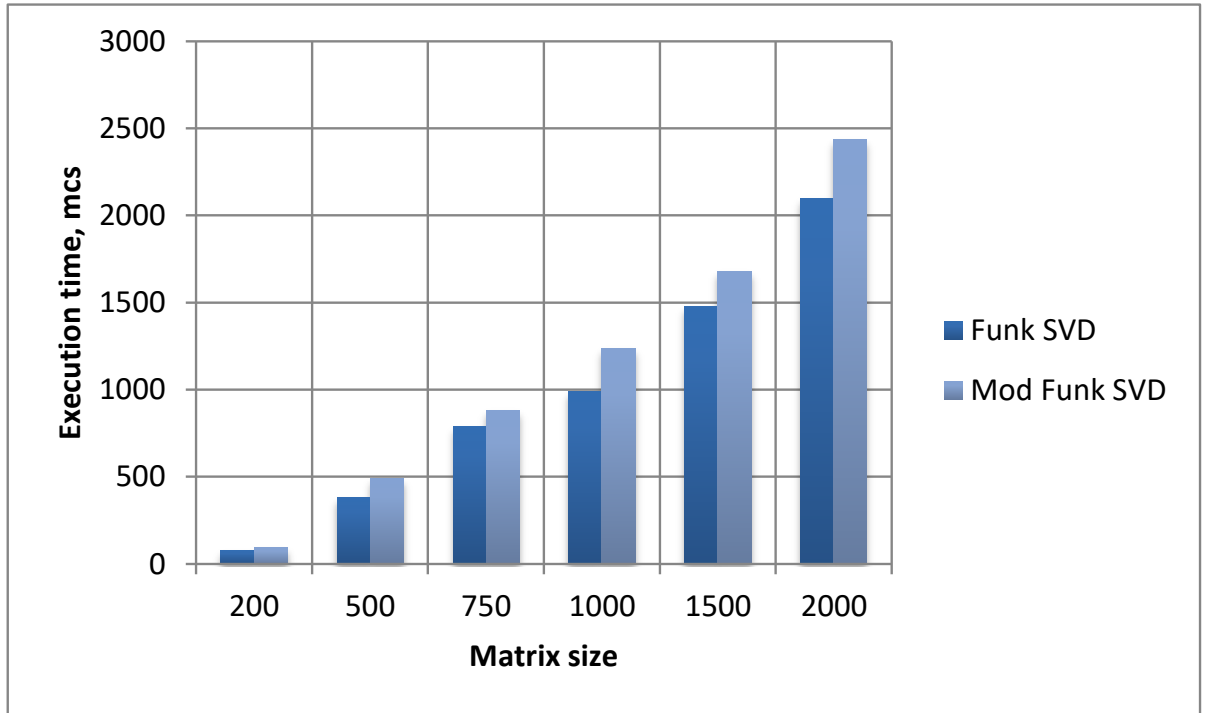


Рис.4.7. Порівняння часу виконання немодифікованим алгоритмом Funk SVD з використанням додаткових функцій

З Рис.4.7 можна зробити висновок, що запропонована модифікація вимагає залучення більшої обчислювальної потужності. Тому тривалість обробки даних трохи збільшується. Порівняння точності розрахунку показано на Рис.4.8.

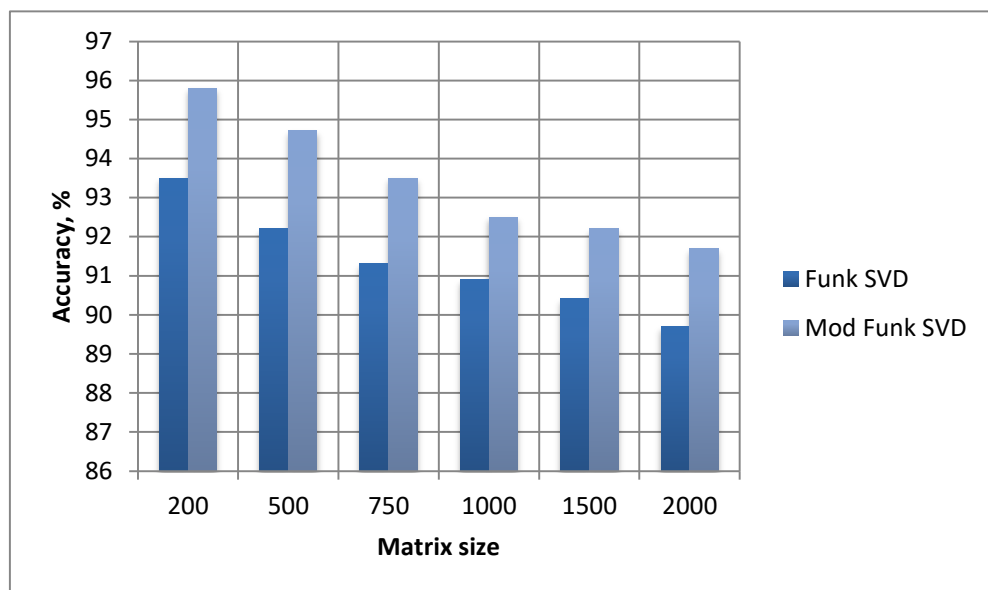


Рис.4.8. Порівняння точності алгоритмів Funk SVD без модифікації та з використанням додаткових функцій

Як показано на Рис.4.8, модифікований Funk SVD демонструє кращу точність обчислень, тому його можна використовувати для формування ефективних рекомендацій. В результаті використання додаткових властивостей продуктів можна краще визначити їх цінність для споживачів.

Загалом, проблема швидкої та точної обробки великих обсягів даних є ключовим моментом у розробці ефективних промислових систем рекомендацій 5G. Як бачимо з отриманих результатів, дві модифікації алгоритму FunkSVD дозволяють більш гнучко організувати розрахунок рекомендацій. Якщо необхідно зменшити тривалість алгоритму, використовуємо першу модифікацію. Для підвищення точності розрахунків використовуємо другий. Тепер давайте порівняємо ефективність одночасного використання двох модифікацій алгоритму FunkSVD у порівнянні з немодифікованим. Отже, визначимо рекомендації:

$$a'_{i,j} = \sum_{i=\delta, j=k} (m_i \times n_{j_mod}^T), \quad (4.18)$$

Давайте обчислимо похибку рекомендацій відповідно до рівняння 6. і оновимо матриці M та N :

$$m'_i = m_i + \varepsilon(Err \times 2 \times n_{j_mod} + \theta \times m_i), \quad (4.19)$$

$$n'_{j_mod} = n_{j_mod} + \varepsilon(Err \times 2 \times m_i + \theta \times n_{j_mod}). \quad (4.20)$$

Тепер ми можемо розрахувати оновлені рекомендації:

$$a'_{i,j} = m'_i \times n'_{j_mod}^T + d_{i,j}, \quad (4.21)$$

Результати порівняння тривалостей розрахунків для двічі модифікованого FunkSVD порівняно з немодифікованим наведено на Рис.4.9.

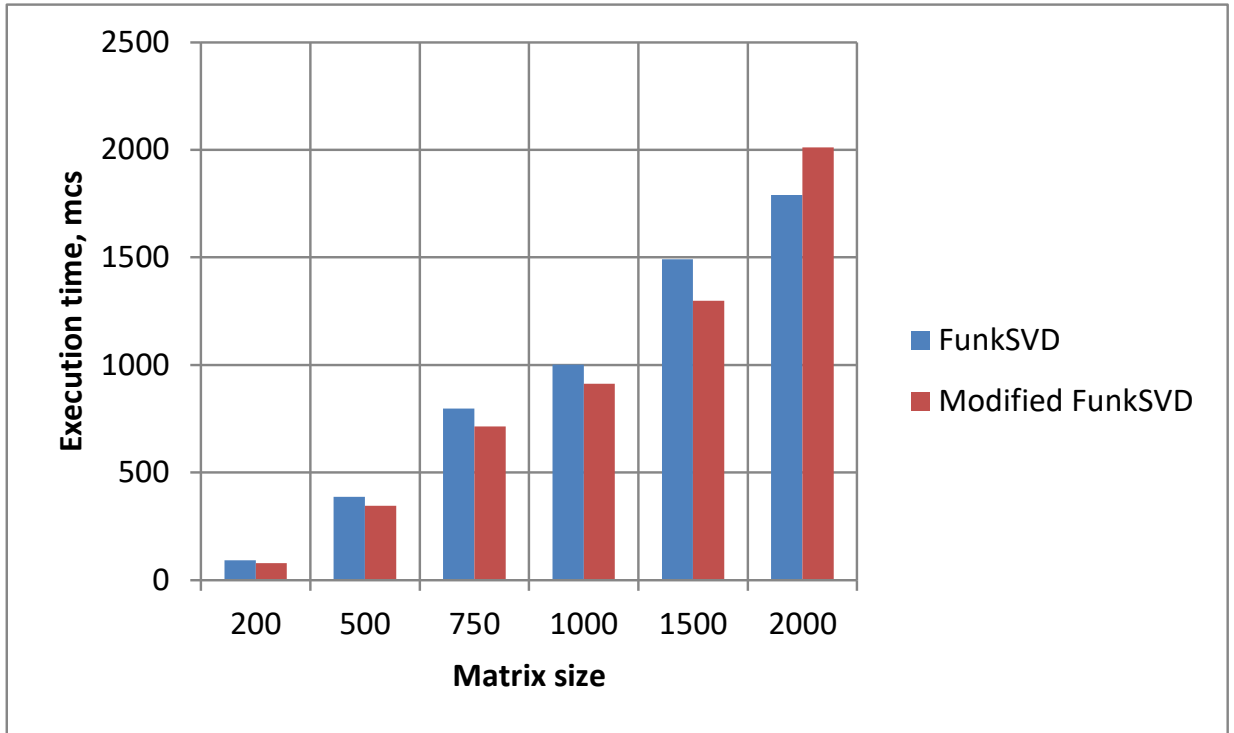


Рис.4.9. Час виконання двічі модифікованого алгоритму FunkSVD порівняно з немодифікованим

Порівняння точності надання рекомендацій для двох алгоритмів показано на Рис.4.10.

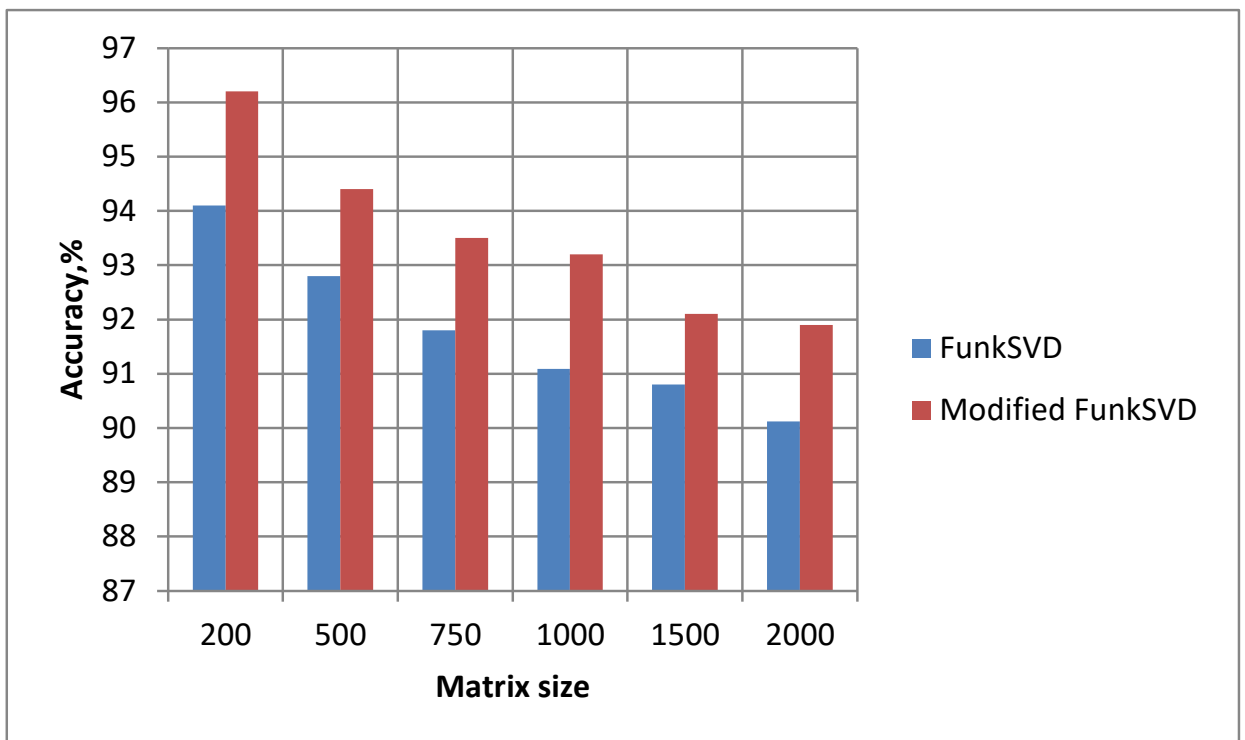


Рис.4.10. Точність надання рекомендацій для двічі модифікованого алгоритму FunkSVD порівняно з немодифікованим

Проведені дослідження демонструють ефективність Funk SVD для обробки великих даних. Запропоновані модифікації дозволяють використовувати цей алгоритм адаптовано до вимог роботи та навантаження на систему для досягнення найкращої ефективності обчислення даних користувачів у системах 5G. Для підвищення надійності та конфіденційності збору даних від кінцевих користувачів ми рекомендуємо модифікований федеративний алгоритм SVD, який розглядався в нашому попередньому дослідженні [147,148,149]. Підвищення надійності запропонованого в цій роботі модифікованого методу FunkSVD буде предметом наших майбутніх досліджень.

Швидкість і точність обробки даних сприяє кращому розумінню потреб клієнта та оперативному наданню відповідних послуг. Таким чином, запропоновані модифікації сприяють модернізації систем 5G, розширенню спектру їхніх послуг та покращенню взаємодії з користувачами. Враховуючи постійну появу нових послуг для користувачів у мережах мобільного зв'язку 5G, проблема гнучкого розподілу ресурсів вже є актуальною. Залежно від потреб і статусу клієнтів, рівня послуг, які їм потрібні, і стану мобільної мережі ми можемо підібрати відповідний алгоритм рекомендацій. Також одночасне використання двох модифікацій підвищує як точність, так і тривалість обчислень [150,151,152].

4.4. Практичні аспекти реалізації створеної моделі для обробки даних великих обсягів в IoT системах

Робота рекомендаційних систем вимагає використання спеціальних методів пошуку закономірностей в матрицях даних. Важливо також визначити найважливішу інформацію для подальшої обробки та систематизувати її залежно від завдань. Одним із основних алгоритмів, що використовуються в рекомендаційних системах, є SVD (Singular Value Decomposition), який розкладає початкову матрицю даних A на три підматриці U , S , V (U та V містять інформацію про користувачів і продукти, S – про зв'язки зв'язків). між ними).

Для отримання даних про елемент початкової матриці $a_{i,j}$ достатньо обчислити скалярне множення рядка i стовпця матриць U і V з відповідними індексами (i,j) . Матриця S використовується для визначення неважливих даних, оскільки вона містить елементи лише на головній діагоналі, розташовані в порядку спадання. Після визначення k найбільших елементів ми можемо відкинути інші рядки та стовпці, які не важливі для подальших обчислень. Також виберемо k рядків у матрицях U і V . Перемноженням скорочених матриць отримуємо нову, яка менша за розміром від вихідної, але містить її найважливіші характеристики. Модифікацією SVD є Funk SVD, яка передбачає відразу встановлення параметра k , без необхідності розкладання матриці на субфактори [153,154,155,156].

Ці алгоритми ефективно працюють в умовах обробки великих матриць даних. Однак їх досить складно використовувати в промислових системах, що характеризуються розподіленою архітектурою. Для вирішення цих проблем використовуються різні підходи. Дані можна збирати від користувачів, а обчислення виконувати на центральному пристрої, або вузли виконують обробку незалежно та обмінюються результатами. У першому випадку створюється високе навантаження на канали зв'язку, а також виникають ризики для безпеки даних користувачів. При другому підході виникає проблема визначення додаткового алгоритму локального розподілу даних та обміну з іншими вузлами, який забезпечить якісні обчислення. У дослідженні була запропонована розподілена модифікація Funk SVD, де кожен вузол обчислює декомпозицію Funk SVD, використовуючи свої дані:

$$A_{node} = U_{local} \times S_{local} \times V_{local}^T, \quad (4.22)$$

Потім U_{local} і S_{local} надсилаються кінцевими пристроями на центральний пристрій, який обчислює їх загальне середнє значення:

$$U_{global} = mean(U_{local_1}, \dots, U_{local_n}), \quad (4.23)$$

$$S_{global} = mean(S_{local_1}, \dots, S_{local_n}), \quad (4.34)$$

Отримані з центрального вузла U_{global} , S_{global} надсилаються до локальних вузлів, і кожен з них може обчислити V_{global}^T :

$$V_{global}^T = U_{global}^T \times A_{node} \div S_{global}. \quad (4.25)$$

Далі кожен вузол може обчислити власну матрицю рекомендацій, використовуючи U_{global} , S_{global} , отримані від центрального вузла та локально обчислене V_{global}^T :

$$A_{predicted} = U_{global} \times S_{global} \times V_{global}^T. \quad (4.26)$$

Відповідно до модифікованого алгоритму дані користувача не пересилаються між вузлами, але можливий обмін результатами локального навчання. Декомпозиція даних на основі модифікованого алгоритму показана на Рис.4.11.

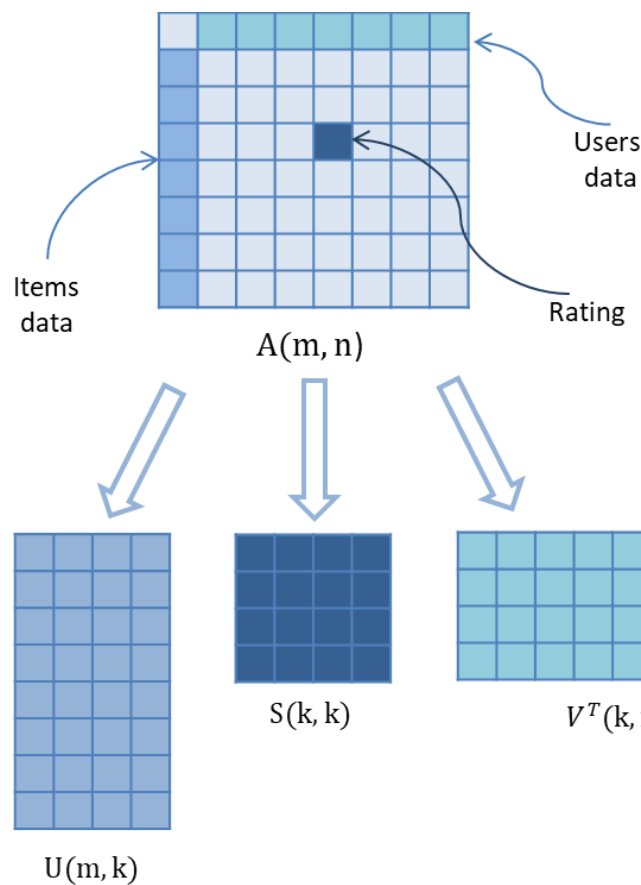


Рис.4.11. Декомпозиція даних за допомогою модифікованого алгоритму Funk SVD (к — кількість даних, вилучених для подальших обчислень)

На відміну від наших попередніх досліджень, зокрема [12], ми модифікуємо алгоритм Funk SVD, а не SVD, для роботи з розрідженими даними. Також відрізняється спосіб обміну даними, ми не просто збираємо дані з кінцевих пристроїв і обчислюємо їх на центральному. Натомість локальні пристрої здійснюють повноцінну декомпозицію даних і можуть самі формувати

рекомендації. Обмін результатами необхідний для кращого процесу навчання. Звичайно, такий підхід передбачає попереднє визначення пристроїв, здатних виконувати такі розрахунки [156,157,158].

Обсяг даних, які використовуються для подальшої обробки в алгоритмі Funk SVD, розраховується за допомогою методу градієнтного спуску (GD). У дослідженні к ми встановлювали самостійно, через складність проведення GD в розподілених системах, на рівні 70% вихідних даних (к можна змінювати в залежності від параметрів системи та вимог розрахунку). Обмін лише значеннями U_{local} і S_{local} дозволяє локальним пристроям захищати приватні дані, оскільки їх важко відновити, не знаючи значення V_{local}^T . Робота алгоритму показана на Рис.4.12.

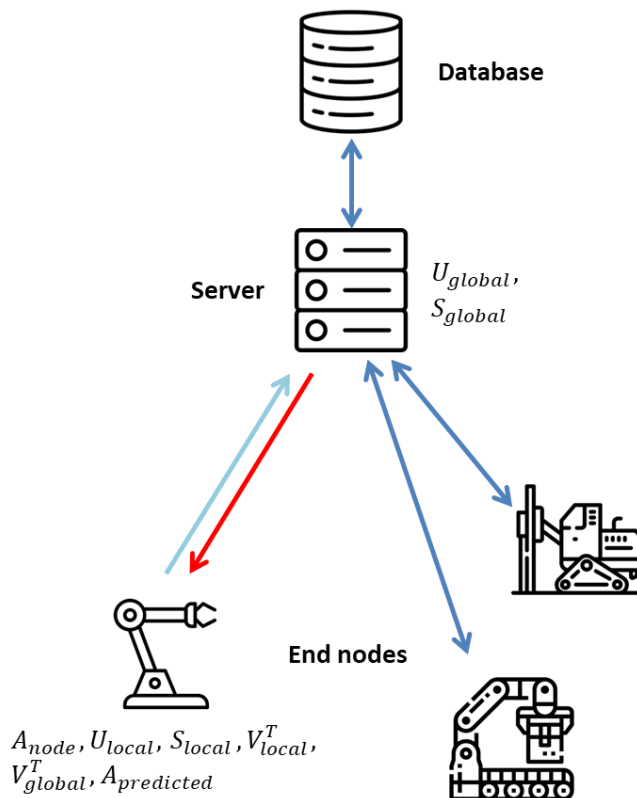


Рис.4.12. Модифікований алгоритм Funk SVD в розподіленій промисловій системі

Для порівняння ефективності запропонованого та немодифікованого алгоритму реалізовано модель їх розрахунку на мові програмування Python. Для цього використовуються вбудовані мовні бібліотеки, зокрема:

1. NumPy для обробки даних у матричному представленні;
2. PyTorch і TensorFlow для розподілених алгоритмів обчислень;

3. SciPy для оптимізації алгебраїчних обчислень.

Спочатку ми порівняли час розкладання матриці для модифікованого та немодифікованого алгоритмів Funk SVD. Для кожного алгоритму використовувався різний обсяг оброблених даних (розмір матриці). Немодифікований Funk SVD використовувався для обміну всіма даними користувача з центральним вузлом, який проводив декомпозицію та давав рекомендації. Модифікований алгоритм FunkSVD виконав обчислення, як пояснено вище (Рівняння 1-6). Обидва алгоритми в цьому експерименті працювали ітераційно, поки точність відновлення початкової матриці не становила 90% (частина даних з вихідної матриці була видалена, а налаштовані алгоритми мали завдання її відновлення). Дані були вибрані з бази даних MovieLens 10M. Було використано чотири розподілених обчислювальних процеси (Рис.4.13).

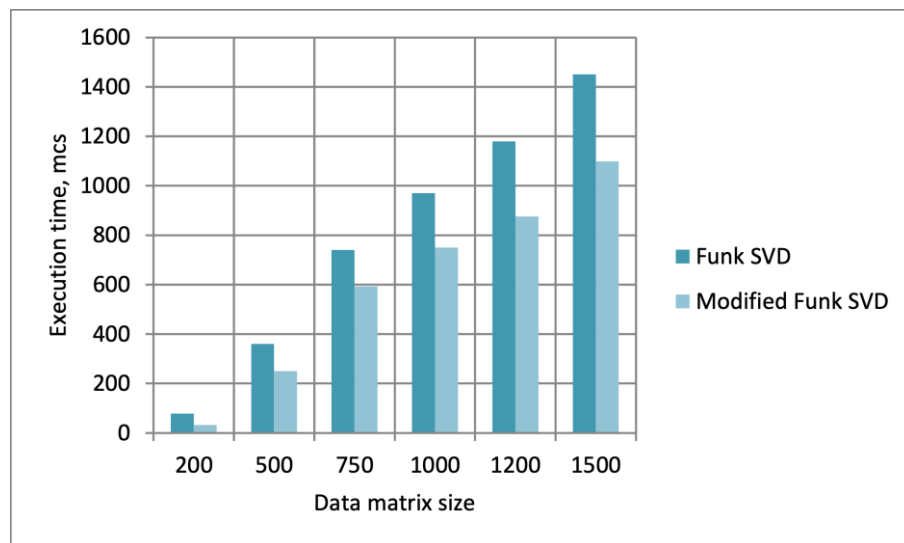


Рис.4.13. Залежність часу виконання від розміру матриці даних для Funk SVD і Modified Funk SVD

На Рис.4.13 показано кращу швидкість обчислень для модифікованого алгоритму, що пояснюється обміном результатами локальних обчислень між вузлами. Такий результат може відрізнятися в залежності від специфіки промислової системи. Тепер порівнюємо точність розрахунку рекомендацій, тобто відновлення відсутніх значень (рейтингів) у вихідній матриці. На Рис.4.14 показано продуктивність модифікованого та звичайного Funk SVD залежно від обсягу оброблених даних (на відміну від попереднього дослідження, де ми

проводили розрахунки до досягнення заданої точності рекомендацій, у цьому ми визначали точність після однієї ітерації алгоритми).

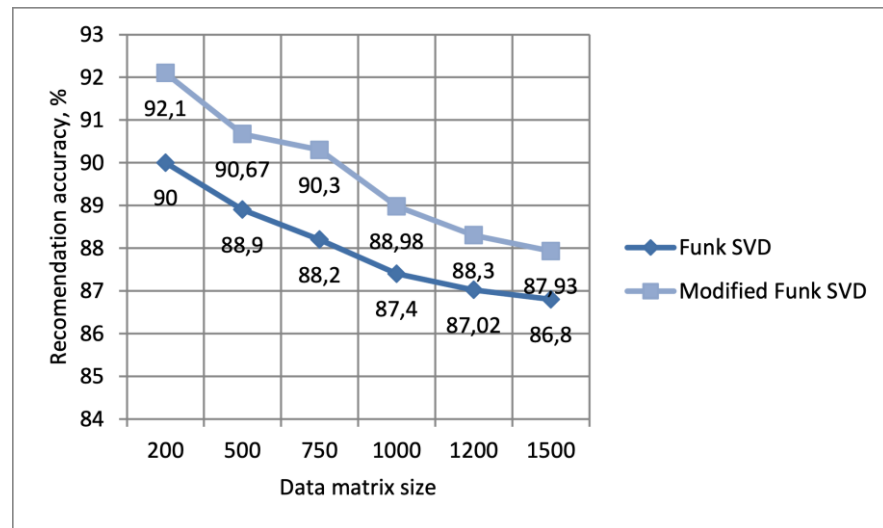


Рис.4.14. Залежність точності рекомендацій від розміру матриці даних для Funk SVD і Modified Funk SVD

Відповідно до Рис.4.14, модифікований Funk SVD виконує обчислення точніше, ніж немодифікований. Також порівнюємо тривалість обчислень модифікованого алгоритму залежно від кількості розподілених вузлів у системі (Рис.4.15).

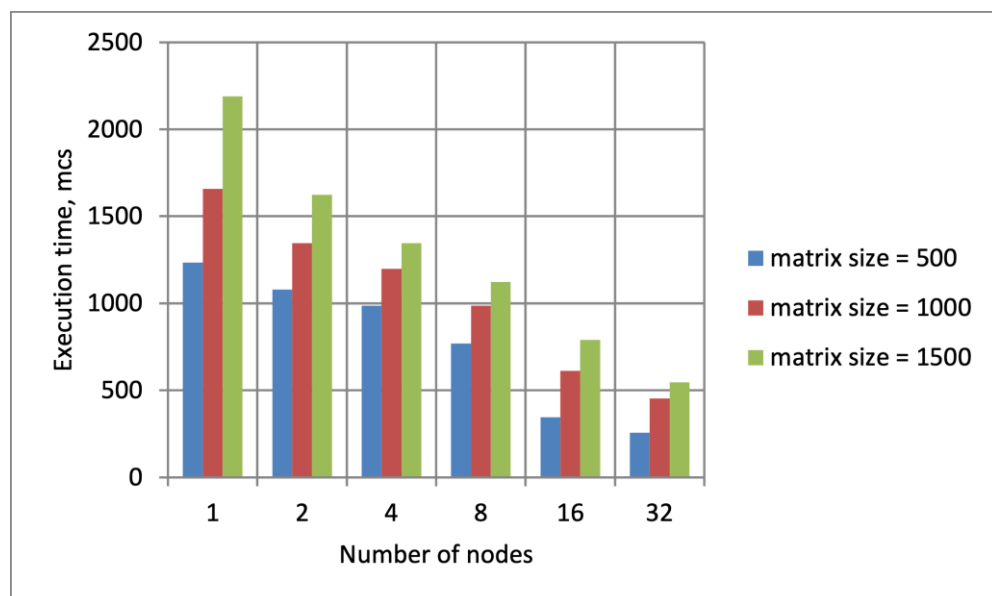


Рис.4.15. Залежність тривалості розрахунку від числа вузлів для Modified Funk SVD

Відповідно до Рис.4.15, регулювання кількості обчислювальних процесів допомагає регулювати швидкість обчислень залежно від вимог до роботи промислової системи. Результати рекомендацій, які придатні для вирішення

задачі однієї локальної підсистеми, можуть бути використані в інших підсистемах промислового виробництва. Результати досліджень показані на Рис.4.13-Рис.4.15 свідчать про переваги використання запропонованого методу в промислових системах обробки великих даних і забезпечують високі показники швидкості та точності обчислень. За допомогою запропонованого методу можна створити розумну модель обробки великих даних у промислових системах, яка дозволить гнучко керувати системою.

Структура рекомендаційної системи з використанням алгоритму Funk SVD наведена на Рис.4.16.

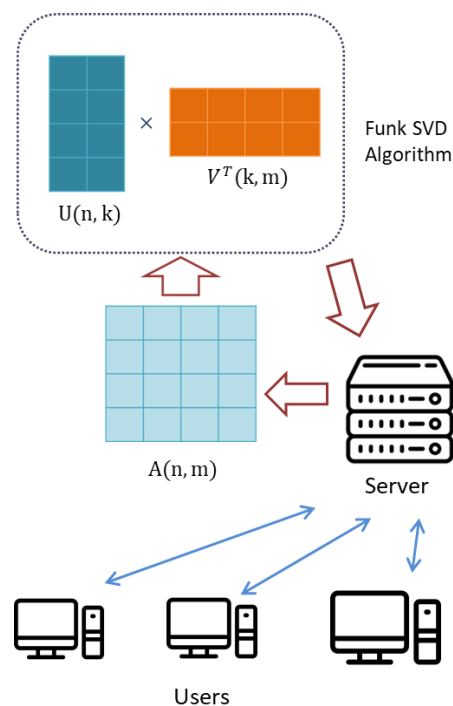


Рис.4.16. Алгоритм Funk SVD

Funk SVD ефективно використовується при обробці розріджених даних. Ця проблема актуальна для великомасштабних систем рекомендацій, де рейтингові дані для всіх користувачів і продуктів часто відсутні, а матриці рекомендацій містять багато порожніх клітинок.

Незважаючи на численні переваги, використання Funk SVD в розподілених промислових системах має певні недоліки. Зокрема, надсилання великих обсягів даних уповільнює систему рекомендацій і ставить під загрозу її конфіденційність. Тому в цій роботі пропонується вдосконалений розподілений алгоритм Funk SVD, який дозволяє виконувати обчислення

безпосередньо на кінцевих пристроях. Потім лише результати визначення рекомендацій надсилаються на сервер, який формує глобальну систему рекомендацій, яка потім знову надсилається всім локальним моделям (Рис.4.17).

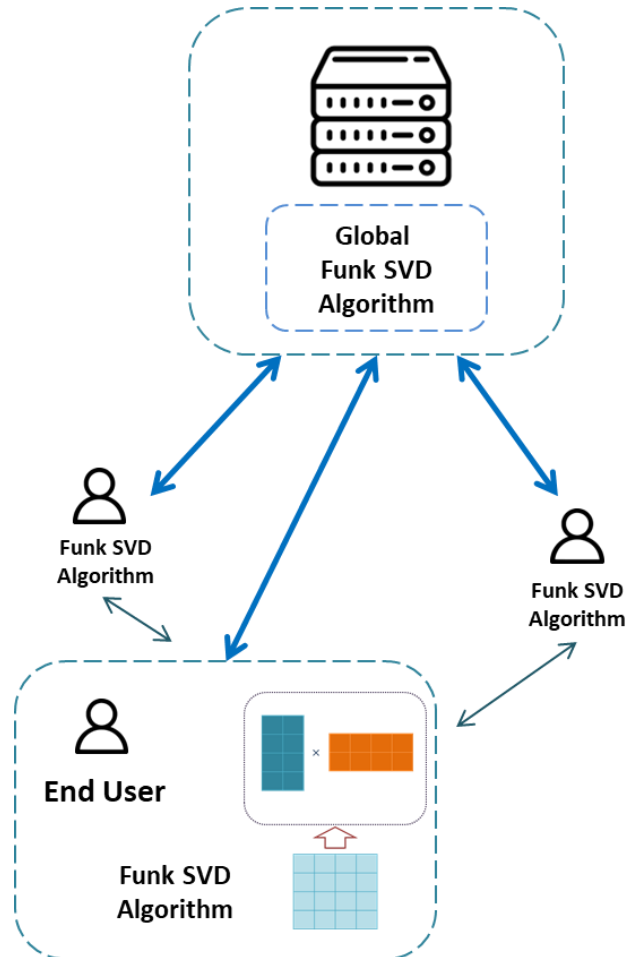


Рис.4.17. Алгоритм Distributed Funk SVD

Розподілений алгоритм Funk SVD сприяє обміну результатами формування рекомендацій між різними кінцевими пристроями. Таким чином, підвищується точність рекомендаційних систем

Висновки до четвертого розділу.

В даному розділі розроблено архітектуру розподіленої обробки великих даних. Архітектура системи було обрано з урахуванням необхідності збору даних з кінцевих малопотужних пристроїв, тому було запропоновано розподілений алгоритм машинного навчання. Також було використано алгоритми, запропоновані в попередньому розділі, для ефективнішої обробки

даних. Розроблено архітектуру рекомендаційної системи оброблення великих даних, що швидко і надійно опрацьовує інформацію про користувачів. В розділі представлено отриманні результати роботи, а саме проілюстровано графічні результати ефективності виконання запропонованих алгоритмів. У розділі розв'язано задачі дослідження 6,7.

ВИСНОВКИ

У дисертаційній роботі розв'язано актуальне наукове завдання розроблення методів та засобів оброблення великих даних в розподілених інформаційних системах.

Основні результати подано нижче.

1. Здійснено аналіз предметної області, особливостей аналізу даних, а саме: знаходження закономірностей, визначення найважливішої для оброблення інформації та відкинення надлишковостей, при збереженні тривалості та точності обчислень. Це дало змогу виділити невирішеня задачі існуючими методами та здійснити постановку задачі. Визначено рекомендаційні системи як інструмент знаходження подібностей між користувачами і товарами.

2. Вдосконалено метод розподіленого сингулярного розкладу даних Fed SVD на основі використання маски для захисту даних користувачів. Це стало основою для розроблення моделі для збору та оброблення даних з багатьох віддалених пристроїв. Також метод дозволяє підвищити точність обчислень рекомендацій користувачам щодо нових товарів.

3. Розроблено модель розподіленого опрацювання даних, що зібрані з різних джерел, в промислових інформаційних системах. Запропонована модель дає змогу знаходити взаємозв'язки між користувачами і товарами, попередньо визначивши найважливіші дані. Також це дало змогу зберігати високу точність і надійність оброблення користувацької інформації.

4. Вдосконалено метод сингулярного розкладу даних Funk SVD, що дозволяє проводити обчислення рекомендацій на основі використання меншої кількості даних про користувачів і товари. Такий підхід дозволяє прискорити обчислення та надавати оперативні результати користувачам.

5. Розроблено метод розподіленого формування рекомендацій Mod FedSVD, який, за допомогою модифікованого методу Funk SVD, передбачає визначення рекомендацій користувачами з використанням власних даних, а також збір локальних результатів і оновлення глобальної моделі рекомендацій.

6. На основі запропонованого методу модифікованого FedSVD розроблено модель інтелектуальної рекомендаційної системи для IoT, що надає рекомендації користувачам про потенційно цікаві товари чи послуги. Для забезпечення надійності, точності та швидкості обчислень запропоновані методи дозволяють гнучко обирати оптимальні параметри обчислень. Розроблено програмну модель, у якій реалізовані усі наукові результати.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Distributed Singular Value Decomposition Method for Fast Data Processing in Recommendation Systems”, Przystupa, K.; Beshley, M.; Hordiichuk-Bublivska, O.; Kyryk, M.; Beshley, H.; Pyrih, J.; Selech, J. *Energies* 2021, 14, 2284
2. Beshley, M.; HordiichukBublivska, O.; Beshley, H.; Ivanochko, I. Data Optimization for Industrial IoT-Based Recommendation Systems. *Electronics* 2023, 12, 33. [https:// doi.org/10.3390/electronics12010033](https://doi.org/10.3390/electronics12010033)
3. Olena Hordiichuk-Bublivska, Halyna Beshley, Marian Kyryk, Yuliia Pyrih, Oksana Urikova, and Mykola Beshley. A Modified Federated Singular Value Decomposition Method for Big Data and ML Optimization in IIoT Systems. In: Mikhailo Klymash, Andriy Luntovsky, Mykola Beshley, Igor Melnyk, Alexander Schill. (eds) *Emerging Networking in the Digital Transformation Age: Approaches, Protocols, Platforms, Best Practices, and Energy Efficiency. Lecture Notes in Electrical Engineering*, 2023, Springer, Cham. 965, P.246-268
4. Yiming, Z.; Fang, X.; Hordiichuk-Bublivska, O.; Beshley, H.; Beshley, M. Modified Masking-Based Federated Singular Value Decomposition Method for Fast Anomaly Detection in Smart Grid Systems. *Energies* 2023, 16, 5996. <https://doi.org/10.3390/en16165996>
5. Olena Hordiichuk-Bublivska, Halyna Beshley, Natalia Kryvinska and Mykola Beshley. A masking-based federated singular value decomposition method for anomaly detection in industrial internet of things. *International Journal of Web and Grid Services*, Vol. 19, No. 3. pp 287-317. <https://doi.org/10.1504/IJWGS.2023.133502>
6. M. Klymash, M. Kyryk, I. Demydov, O. Hordiichuk-Bublivska, H. Kopets and N. Pleskanka, “Research on Distributed Machine Learning Methods in Databases,” 2021 IEEE 4th International Conference on Advanced Information and Communication Technologies (AICT), 2021, pp. 128-131, doi: 10.1109/AICT52120.2021.9628949.
7. M. Klymash, M. Kyryk, O. Hordiichuk-Bublivska, L. Fabri, H. Kopets, “Big Data Analysis in IIoT Systems Using the Federated Machine Learning Method”, IEEE

- 16 International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), February 22 - 26, 2022, Lviv-Slavske, Ukraine.
8. “Big Data Analysis in Smart Grid Systems”, Yu Jun, Olena Hordiichuk-Bublivska, Yan Lingyu, Marian Kyryk, Mykola Beshley, Hu Jiwei, 18th IMEKO TC10 Conference “Measurement for Diagnostics, Optimisation and Control to Support Sustainability and Resilience” Warsaw, Poland, September 26–27, 2022
 9. Model of large sparse datasets processing efficiency in IIoT. M. Klymash, O. Hordiichuk-Bublivska, M. Kyryk, T. Andrukhiv, Y. Pyriih, CADSM 2023
 10. M. Klymash, M. Kyryk, O. Hordiichuk-Bublivska, L. Fabri, “Research on the Automated Decision-Making Effectiveness in Industrial Automation System” 2023 IEEE 5th International Conference on Advanced Information and Communication Technologies (AICT), 2023
 11. О. В. Гордїйчук-Бублівська, Л. П. Фабрі. МАТРИЧНА ФАКТОРИЗАЦІЯ ВЕЛИКИХ ДАНИХ У ПРОМИСЛОВИХ СИСТЕМАХ. УЖІТ. 2022; Випуск 4, Номер 2: 68-73. <https://doi.org/10.23939/ujit2022.02.068>
 12. Dobrilovic, D.; Brtko, V.; Stojanov, Z.; Jotanovic, G.; Perakovic, D.; Jausevac, G. A Model for Working Environment Monitoring in Smart Manufacturing. *Appl. Sci.* 2021, 11, 2850.
 13. Jun, S.; Przystupa, K.; Beshley, M.; Kochan, O.; Beshley, H.; Klymash, M.; Wang, J.; Pieniak, D. A Cost-Efficient Software Based Router and Traffic Generator for Simulation and Testing of IP Network. *Electronics* 2020, 9, 40.
 14. Beshley, M.; Kryvinska, N.; Seliuchenko, M.; Beshley, H.; Shakshuki, E.M.; Yasar, A.-U.-H. End-to-End QoS “Smart Queue” Management Algorithms and Traffic Prioritization Mechanisms for Narrow-Band Internet of Things Services in 4G/5G Networks. *Sensors* 2020, 20, 2324.
 15. Bzai, J.; Alam, F.; Dhafer, A.; Bojović, M.; Altowaijri, S.M.; Niazi, I.K.; Mehmood, R. Machine Learning-Enabled Internet of Things (IoT): Data, Applications, and Industry Perspective. *Electronics* 2022, 11, 2676.

16. Cui, Z.; Xu, X.; Xue, F.; Cai, X.; Cao, Y.; Zhang, W.; Chen, J. Personalized Recommendation System Based on Collaborative Filtering for IOT Scenarios. *IEEE Transactions on Services Computing* 2020, 13, 685–695.
17. Maheswari, M.; Brintha, N. C. Smart Manufacturing Technologies in Industry-4.0. In *2021 Sixth International Conference on Image Information Processing (ICIIP)*; Institute of Electrical and Electronics Engineers (IEEE), Shimla, India, 26-28 November 2021; Vol. 6, pp. 146–151.
18. Moutzidis, I.; Kamariotou, M.; Kitsios, F. Digital Transformation Strategies Enabled by Internet of Things and Big Data Analytics: The Use-Case of Telecommunication Companies in Greece. *Information* 2022, 13, 196.
19. Chen, M. Exploration of Geological Informatization Based on the Internet of Things and Cloud Computing in the Era of Big Data. In *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)*; Institute of Electrical and Electronics Engineers (IEEE), Erode, India, 29-31 March 2022; pp. 856–859.
20. Cui, Z.; Xu, X.; Xue, F.; Cai, X.; Cao, Y.; Zhang, W.; Chen, J. Personalized Recommendation System Based on Collaborative Filtering for IOT Scenarios. *IEEE Transactions on Services Computing* 2020, 13, 685–695.
21. Araújo, N.; Pacheco, V.; Costa, L. Smart Additive Manufacturing: The Path to the Digital Value Chain. *Technologies* 2021, 9, 88..
22. Bai, J.; Fang, S.; Tang, R.; Wu, Y. Bills of Standard Manufacturing Services (BOSS) Construction Based on Focused Crawler. In *2019 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE)*; Institute of Electrical and Electronics Engineers (IEEE), Chengdu, China, 19-21 March 2021; pp. 135–140.
23. Teng, Y.; Li, L.; Song, L.; Yu, F. R.; Leung, V. C. Profit Maximizing Smart Manufacturing over AI-Enabled Configurable Blockchains. *IEEE Internet of Things Journal* 2022, 9, 346–358.
24. Canese, L.; Cardarilli, G.C.; Di Nunzio, L.; Fazzolari, R.; Giardino, D.; Re, M.; Spanò, S. Multi-Agent Reinforcement Learning: A Review of Challenges and Applications. *Appl. Sci.* 2021, 11, 4948.

25. Liu, C.-L.; Chang, C.-C.; Tseng, C.-J. Actor-Critic Deep Reinforcement Learning for Solving Job Shop Scheduling Problems. *IEEE Access* 2020, 8, 71752–71762.
26. Calì, M. Smart Manufacturing Technology. *Appl. Sci.* 2021, 11, 8202.
27. Ortega, F.; González-Prieto, Á. Recommender Systems and Collaborative Filtering. *Appl. Sci.* 2020, 10, 7050.
28. Jia, X.; Liu, F. Research on Intelligent Recommendation System Model Supported by Data Mining and Algorithm Optimization. In 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT); Institute of Electrical and Electronics Engineers (IEEE), Chongqing, China, 22-24 November 2021; pp. 766–769.
29. Liu, F. Design of Personalized Catering Recommendation System Based on Spark Platform. In 2022 2nd Asia Conference on Information Engineering (ACIE); Institute of Electrical and Electronics Engineers (IEEE), Haikou, China, 15-17 January 2022; pp. 90–95.
30. Jin, J.; Guo, H.; Xu, J.; Wang, X.; Wang, F.-Y. An End-to-End Recommendation System for Urban Traffic Controls and Management under a Parallel Learning Framework. *IEEE Trans. Intell. Transp. Syst.* 2021, 22 (3), 1616–1626.
31. Bagul, D. V.; Barve, S. A Novel Content-Based Recommendation Approach Based on LDA Topic Modeling for Literature Recommendation. In 2021 6th International Conference on Inventive Computation Technologies (ICICT); Institute of Electrical and Electronics Engineers (IEEE), Coimbatore, India, 20-22 January 2021; pp. 954–961.
32. Benrhouma, O.; Alkhodre, A.B.; AlZahrani, A.; Namoun, A.; Bhat, W.A. Using Singular Value Decomposition and Chaotic Maps for Selective Encryption of Video Feeds in Smart Traffic Management. *Appl. Sci.* 2022, 12, 3917.
33. Gao, L.-Y.; Liu, M.-Z.; Yue, J.-Y.; Tian, Y.-H. Source Number Estimation Based on Improved Singular Value Decomposition at Low SNR. In 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC); Institute of Electrical and Electronics Engineers (IEEE), Beijing, China, 12-14 July; 2019; pp. 1–4.

34. Jiang, J.; Yang, W.; Ren, H.; Chen, M. Diffraction Separation and Imaging Using an Improved Singular Value Decomposition Method. *IEEE Geoscience and Remote Sensing Letters* 2022, 19, 1–5.
35. Li, T.; Chen, Y.; Zhu, G. Hybrid Recommendation Algorithm Based on Hamming Clustering for User's Access Log and Weighted User Behavior. In *2018 15th International Conference on Service Systems and Service Management (ICSSSM)*, Hangzhou, China, 21-22 July 2018; pp. 1–7.
36. Kumar, B. A Novel Latent Factor Model for Recommender System. *Journal of Information Systems and Technology Management* 2016, 13.
37. Wróblewska, A.; Dąbrowski, J.; Pastuszak, M.; Michałowski, A.; Daniluk, M.; Rychalska, B.; Wieczorek, M.; Sysko-Romańczuk, S. Designing Multi-Modal Embedding Fusion-Based Recommender. *Electronics* 2022, 11, 1391.
38. Ko, H.; Lee, S.; Park, Y.; Choi, A. A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields. *Electronics* 2022, 11, 141.
39. F. Zou, L. Shen, Z. Jie, W. Zhang, and W. Liu, "A sufficient condition for convergences of Adam and RMSProp," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2019, pp. 11119–11127.
40. Z. Lin et al., "Refracting RIS-Aided Hybrid Satellite-Terrestrial Relay Networks: Joint Beamforming Design and Optimization," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 4, pp. 3717-3724, Aug. 2022, doi: 10.1109/TAES.2022.3155711.
41. Z. Lin et al., "SLNR-based Secure Energy Efficient Beamforming in Multibeam Satellite Systems," in *IEEE Transactions on Aerospace and Electronic Systems*, 2022, doi: 10.1109/TAES.2022.3190238.
42. Z. Lin, M. Lin, J. -B. Wang, T. de Cola and J. Wang, "Joint Beamforming and Power Allocation for Satellite-Terrestrial Integrated Networks With Non-Orthogonal Multiple Access," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 3, pp. 657-670, June 2019, doi: 10.1109/JSTSP.2019.2899731.

43. H. Niu et al., "Joint Beamforming Design for Secure RIS-Assisted IoT Networks," in *IEEE Internet of Things Journal*, 2022, doi: 10.1109/JIOT.2022.3210115.
44. Y. Xiaochen and L. Qicheng, "Parallel Algorithm of Improved FunkSVD Based on GPU," in *IEEE Access*, vol. 10, pp. 26002-26010, 2022, doi: 10.1109/ACCESS.2022.3156969
45. J. Gross et al., "A Systematic Literature Review of Data Privacy Solutions for Smart Meter Technologies," 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 2021, pp. 3305-3310, doi: 10.1109/BigData52589.2021.9671814.
46. M. Li, H. Wang and J. Li, "Mining conditional functional dependency rules on big data," in *Big Data Mining and Analytics*, vol. 3, no. 1, pp. 68-84, March 2020, doi: 10.26599/BDMA.2019.9020019.
47. A. Cuzzocrea and E. Damiani, "Privacy-Preserving Big Data Exchange: Models, Issues, Future Research Directions," 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 2021, pp. 5081-5084, doi: 10.1109/BigData52589.2021.9671686.
48. Z. Dong, "Research of Big Data Information Mining and Analysis : Technology Based on Hadoop Technology," 2022 International Conference on Big Data, Information and Computer Network (BDICN), Sanya, China, 2022, pp. 173-176, doi: 10.1109/BDICN55575.2022.00041.
49. W. Li, "Big Data Mining Algorithm of Internet of Things Based on Artificial Intelligence Technology," 2022 International Symposium on Advances in Informatics, Electronics and Education (ISAIEE), Frankfurt, Germany, 2022, pp. 115-121, doi: 10.1109/ISAIEE57420.2022.00032.
50. M. Sadeghi, SA Asghari and MM Pedram, "An Improved Method Multi-View Group Recommender System (IMVGRS)," 2020 8th Iranian Joint Congress on Fuzzy and intelligent Systems (CFIS), Mashhad, Iran, 2020, pp. 127-132, doi: 10.1109/CFIS49607.2020.9238688.
51. KA Fararni, F. Nafis, B. Aghoutane, A. Yahyaouy, J. Riffi and A. Sabri, "Hybrid recommender system for tourism based on big data and AI: A conceptual

- framework," in *Big Data Mining and Analytics*, vol. 4, no. 1, pp. 47-55, March 2021, doi: 10.26599/BDMA.2020.9020015.
52. AH Ragab and P. El-Kafrawy, "Using Knowledge Graph Embeddings in Embedding Based Recommender Systems," 2022 20th International Conference on Language Engineering (ESOLEC), Cairo, Egypt, 2022, pp. 129-132, doi: 10.1109/ESOLEC54569.2022.10009491.
53. A. Althbiti, R. Alshamrani, T. Alghamdi, S. Lee and X. Ma, "Addressing Data Sparsity in Collaborative Filtering Based Recommender Systems Using Clustering and Artificial Neural Network," 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), NV, USA, 2021, pp. 0218-0227, doi: 10.1109/CCWC51732.2021.9376008.
54. AP Sundar, F. Li, X. Zou, Q. Hu and T. Gao, "Multi-Armed-Bandit-based Shilling Attack on Collaborative Filtering Recommender Systems," 2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Delhi, India, 2020, pp. 347-355, doi: 10.1109/MASS50613.2020.00050.
55. X. Jiang, B. Liu, J. Qin, Y. Zhang and J. Qian, "FedNCF: Federated Neural Collaborative Filtering for Privacy-preserving Recommender System," 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 2022, pp. 1-8, doi: 10.1109/IJCNN55064.2022.9892909.
56. DF Murad, R. Hassan, BD Wijanarko, R. Leandros and SA Murad, "Evaluation of Hybrid Collaborative Filtering Approach with Context-Sensitive Recommendation System," 2022 7th International Conference on Business and Industrial Research (ICBIR), Bangkok, Thailand, 2022 , pp. 7-12, doi: 10.1109/ICBIR54589.2022.9786506.
57. X. Li and F. Sun, "Sports training analysis method based on collaborative filtering," 2021 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS), Macau, China, 2021, pp. 83-87, doi: 10.1109/HPBDIS53214.2021.9658449.
58. R. Prabakaran, J. Pradeepkandhasamy and M. Arun, "A Survey on Recommendation Systems using Collaborative Filtering Techniques," 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT),

- Tirunelveli, India, 2023, pp. 1445-1450, doi: 10.1109/ICSSIT55814.2023.10060889.
- 59.H. Sun, L. Huang, Z. Chen and Q. Zheng, "Research on collaborative filtering recommendation method of online learning resources based on learner model," 2022 IEEE 5th International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 2022, pp. 202-206, doi: 10.1109/ICISCAE55891.2022.9927652.
- 60.H. Wang, "MatMat: Matrix Factorization by Matrix Fitting," 2021 IEEE 4th International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 2021, pp. 99-101, doi: 10.1109/ICISCAE52414.2021.9590639.
- 61.DUK Putri et al., "Facial Expression Recognition Using Sparse Complex Matrix Factorization with Ridge Term Regularization," 2021 IEEE 10th Global Conference on Consumer Electronics (GCCE), Kyoto, Japan, 2021, pp. 45-46, doi: 10.1109/GCCE53005.2021.9621871.
- 62.OV Thanh, N. Gillis and F. Lecron, "Bounded Simplex-Structured Matrix Factorization," ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, Singapore, 2022, pp. 9062-9066, doi: 10.1109/ICASSP43922.2022.9747124.
- 63.M. Ueda, Y. Nomura, J. Miyao, T. Kurita and H. Yamada, "Non-negative Matrix Factorization of a set of Economic Time Series with Graph Based Smoothing of Basis Vectors and Sparseness of the Coefficients," 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, 2020, pp. 824-829, doi: 10.1109/SMC42975.2020.9283037.
- 64.J. Hu, Y. Xing, M. Han, F. Wang, K. Zhao and X. Che, "Nonnegative matrix tri-factorization based clustering in a heterogeneous information network with star network schema," in *Tsinghua Science and Technology*, vol. 27, no. 2, pp. 386-395, April 2022, doi: 10.26599/TST.2020.9010049.
- 65."Big Data Analysis in Smart Grid Systems", Yu Jun, Olena Hordiichuk-Bublivska, Yan Lingyu, Marian Kyryk, Mykola Beshley, Hu Jiwei, 18th IMEKO

- TC10 Conference “Measurement for Diagnostics, Optimisation and Control to Support Sustainability and Resilience” Warsaw, Poland, September 26–27, 2022
66. Ortega, F.; González-Prieto, A. Recommender systems and collaborative filtering. *Appl. Sci.* 2020, 10, 7050.
 67. Shafqat, W.; Byun, Y.-C. Enabling “Untact” Culture via Online Product Recommendations: An Optimized Graph-CNN based Approach. *Appl. Sci.* 2020, 10, 5445.
 68. Son, Y.; Choi, Y. Improving Matrix Factorization Based Expert Recommendation for Manuscript Editing Services by Refining User Opinions with Binary Ratings. *Appl. Sci.* 2020, 10, 3395.
 69. Zhang, D.; Liu, L.; Wei, Q.; Yang, Y.; Yang, P.; Liu, Q. “Neighborhood aggregation collaborative filtering based on knowledge graph. *Appl. Sci.* 2020, 10, 3818.
 70. Al-Ghuribi, S.M.; Noah, S.A.M. Multi-Criteria Review-Based Recommender System—The State of the Art. *IEEE Access* 2019, 7, 169446–169468.
 71. Yang, X.; Dong, M.; Chen, X.; Ota, K. Recommender System-Based Diffusion Inferring for Open Social Networks. *IEEE Trans. Comput. Soc. Syst.* 2020, 7, 24–34.
 72. Xiong, F.; Wang, X.; Pan, S.; Yang, H.; Wang, H.; Zhang, C. Social Recommendation with Evolutionary Opinion Dynamics. *IEEE Trans. Syst. Man, Cybern. Syst.* 2018, 1–1.
 73. Nouh, R.M.; Lee, H.-H.; Lee, W.-J.; Lee, J.-D. A Smart Recommender Based on Hybrid Learning Methods for Personal Well-Being Services. *Sensors* 2019, 19, 431.
 74. Rabiou, I.; Salim, N.; Da’U, A.; Osman, A. Recommender System Based on Temporal Models: A Systematic Review. *Appl. Sci.* 2020, 10, 2204.
 75. Wang, Z.; Wu, H.; Jiang, Z.; Ju, P.; Yang, J.; Zhou, Z.; Chen, X. Singular value decomposition-based load indexes for load profiles clustering. *IET Gener. Transm. Distrib.* 2020, 14, 4164–4172.
 76. Hunold, S.; Carpen-Amarie, A. Reproducible MPI Benchmarking is Still Not as Easy as You Think. *IEEE Trans. Parallel Distrib. Syst.* 2016, 27, 3617–3630.

77. Khan, M.; Jin, Y.; Li, M.; Xiang, Y.; Jiang, C. Hadoop Performance Modeling for Job Estimation and Resource Provisioning. *IEEE Trans. Parallel Distrib. Syst.* 2015, 27, 441–454.
78. Yeromenko, V.; Kochan, O. The conditional least squares method for thermocouples error modeling. In *Proceedings of the 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS)*; Institute of Electrical and Electronics Engineers (IEEE), 12–14 September 2013, Berlin, Germany; Volume 1, pp. 157–162.
79. Sridharan, K.; Komarasamy, G.; Raja, S.D.M. Hadoop framework for efficient sentiment classification using trees. *IET Networks* 2020, 9, 223–228.
80. Hu, Z.; Li, D.; Guo, D. Balance resource allocation for spark jobs based on prediction of the optimal resource. *Tsinghua Sci. Technol.* 2020, 25, 487–497.
81. Iannino, V.; Mocci, C.; Vannocci, M.; Colla, V.; Caputo, A.; Ferraris, F. An Event-Driven Agent-Based Simulation Model for Industrial Processes. *Appl. Sci.* 2020, 10, 434.
82. Zhao, T.; Ding, Z. Distributed Agent Consensus-Based Optimal Resource Management for Microgrids. *IEEE Trans. Sustain. Energy* 2017, 9, 443–452.
83. Beshley, M.; Kryvinska, N.; Seliuchenko, M.; Beshley, H.; Shakshuki, E.M.; Yasar, A.-U.-H. End-to-End QoS “Smart Queue” Management Algorithms and Traffic Prioritization Mechanisms for Narrow-Band Internet of Things Services in 4G/5G Net-works. *Sensors* 2020, 20, 2324.
84. Klymash, M.; Beshley, M.; Stryhaluk, B. System for increasing quality of service of multimedia data in convergent networks. In *Proceedings of the 2014 First International Scientific-Practical Conference Problems of Infocommunications Science and Technology*, 14–17 October 2014, Kharkiv, Ukraine; pp. 63–66.
85. Romanchuk, V.; Beshley, M.; Polishuk, A.; Seliuchenko, M. Method for processing multiservice traffic in network node based on adaptive management of buffer resource. In *Proceedings of the 2018 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*; Slavske, Ukraine, 20–24 February 2018; pp. 1118–1122.

86. Jun, S.; Przystupa, K.; Beshley, M.; Kochan, O.; Beshley, H.; Klymash, M.; Wang, J.; Pieniak, D. A Cost-Efficient Software Based Router and Traffic Generator for Simulation and Testing of IP Network. *Electronics* 2019, 9, 40.
87. Handri, K.E.; Idrissi, A. Parallelization of Topk Algorithm Through a New Hybrid Recommendation System for Big Data in Spark Cloud Computing Framework. *IEEE Syst. J.*
88. Chai, Z.; Li, Y.; Han, Y.; Zhu, S. Recommendation system based on singular value decomposition and multi-objective immune optimization. *IEEE Access* 2019, 7, 6060–6071.
89. Ji, Y.; Hong, W.; Shanguan, Y.; Wang, H.; Ma, J. Regularized singular value decomposition in news recommendation system. In *Proceedings of the 2016 11th International Conference on Computer Science & Education (ICCSE); Institute of Electrical and Electronics Engineers (IEEE), Nagoya, Japan, 23–25 August 2016; pp. 621–626.*
90. Achakulvisut, T.; Acuna, D.E.; Ruangrong, T.; Körding, K.P. Science Concierge: A Fast Content-Based Recommendation System for Scientific Publications. *PLOS ONE* 2016, 11, e0158423.
91. Li, G.; Hua, J.; Yuan, T.; Wu, J.; Jiang, Z.; Zhang, H.; Li, T. Novel Recommendation System for Tourist Spots Based on Hierarchical Sampling Statistics and SVD++. *Math. Probl. Eng.* 2019, 2019, 1–15.
92. Guo, X.; Yin, S.-C.; Zhang, Y.-W.; Li, W.; He, Q. Cold Start Recommendation Based on Attribute-Fused Singular Value Decomposition. *IEEE Access* 2019, 7, 11349–11359.
93. Chen, V.; Tang, T. “Incorporating singular value decomposition in user-based collaborative filtering technique for a movie recommendation system: A comparative study”. In *PRAI '19: Proceedings of the 2019 the International Conference on Pattern Recognition and Artificial Intelligence; Association for Computing Machinery: New York, NY, USA, 2019; pp. 12–15.*
94. Ferreira, D.; Silva, S.; Abelha, A.; Machado, J. Recommendation System Using Autoencoders. *Appl. Sci.* 2020, 10, 5510.

95. Hong-Xia, W. An Improved Collaborative Filtering Recommendation Algorithm. In Proceedings of the 2019 IEEE 4th International Conference on Big Data Analytics (ICBDA); Institute of Electrical and Electronics Engineers (IEEE), Suzhou, China, 15–18 March 2019; pp. 431–435.
96. Koprinarov, I.N.; Hitchcock, A.P.; McCrory, C.T.; Childs, R.F. Quantitative Mapping of Structured Polymeric Systems Using Singular Value Decomposition Analysis of Soft X-ray Images. *J. Phys. Chem. B* 2002, 106, 5358–5364.
97. Al-Sabaawi, A.M.A.; Karacan, H.; Yenice, Y.E. Exploiting implicit social relationships via dimension reduction to improve recommendation system performance. *PLOS ONE* 2020, 15, e0231457.
98. Cui, Z.; Xu, X.; Xue, F.; Cai, X.; Cao, Y.; Zhang, W.; Chen, J. Personalized Recommendation System Based on Collaborative Filtering for IoT Scenarios. *IEEE Trans. Serv. Comput.* 2020, 13, 685–695.
99. Luo, H.; Li, M.; Wang, S.; Liu, Q.; Li, Y.; Wang, J. Computational drug repositioning using low-rank matrix approximation and randomized algorithms. *Bioinformatics* 2018, 34, 1904–1912.
100. Chen, H.; Zhao, J.; Luo, Q.; Hou, Y. Distributed randomized singular value decomposition using count sketch. In Proceedings of the 2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC); Institute of Electrical and Electronics Engineers (IEEE), Shenzhen, China, 15–17 December 2017; pp. 187–191.
101. Sun, S.; Przystupa, K.; Wei, M.; Yu, H.; Ye, Z.; Kochan, O. Fast bearing fault diagnosis of rolling element using Lévy Moth-Flame optimization algorithm and Naive Bayes. *Eksploatacja Niezawodn. Maint. Reliab.* 2020, 22, 730–740
102. Wu, J. Simple technique to determine the Givens-rotation matrix in the two-source ICA problem for skewed sources. *Electron. Lett.* 2016, 52, 613–615.
103. 1. M. Klymash, M. Kyryk, I. Demydov, O. Hordiichuk-Bublivska, H. Kopets and N. Pleskanka, "Research on Distributed Machine Learning Methods in Databases," 2021 IEEE 4th International Conference on Advanced Information and Communication Technologies (AICT), 2021, pp. 128-131, doi: 10.1109/AICT52120.2021.9628949..

104. M. Klymash, M. Kyryk, O. Hordiichuk-Bublivska, L. Fabri, H. Kopets, "Big Data Analysis in IIoT Systems Using the Federated Machine Learning Method", IEEE 16 International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), February 22 - 26, 2022, Lviv-Slavske, Ukraine.
105. M. N. Hassan Reza, C. Agamudai Nambi Malarvizhi, S. Jayashree and M. Mohiuddin, "Industry 4.0–Technological Revolution and Sustainable Firm Performance," 2021 Emerging Trends in Industry 4.0 (ETI 4.0), 2021, pp. 1-6, doi: 10.1109/ETI4.051663.2021.9619363.
106. O. Elijah et al., "A Survey on Industry 4.0 for the Oil and Gas Industry: Upstream Sector," in IEEE Access, vol. 9, pp. 144438-144468, 2021, doi: 10.1109/ACCESS.2021.3121302.
107. Reza Rezazadegan; Mahdi Sharifzadeh, "Applications of Artificial Intelligence and Big Data in Industry 4.0 Technologies," in Industry 4.0 Vision for the Supply of Energy and Materials: Enabling Technologies and Emerging Applications , Wiley, 2022, pp.121-158, doi: 10.1002/9781119695868.ch5.
108. S. El Manti and L. El Abbadi, "Integration of Visual Management in the Industry 4.0: Case Study," 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), 2022, pp. 1-4, doi: 10.1109/IRASET52964.2022.9738102.
109. D. Cemernek, H. Gursch and R. Kern, "Big data as a promoter of industry 4.0: Lessons of the semiconductor industry," 2017 IEEE 15th International Conference on Industrial Informatics (INDIN), 2017, pp. 239-244, doi: 10.1109/INDIN.2017.8104778.
110. A. Gupta and S. Jain, "Optimizing performance of Real-Time Big Data stateful streaming applications on Cloud," 2022 IEEE International Conference on Big Data and Smart Computing (BigComp), 2022, pp. 1-4, doi: 10.1109/BigComp54360.2022.00010.
111. K. S. Aggour et al., "Federated Multimodal Big Data Storage & Analytics Platform for Additive Manufacturing," 2019 IEEE International Conference on

- Big Data (Big Data), 2019, pp. 1729-1738, doi: 10.1109/BigData47090.2019.9006495.
112. Z. Tang, "A Preliminary Study on Data Security Technology in Big Data Cloud Com-puting Environment," 2020 International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE), 2020, pp. 27-30, doi: 10.1109/ICBASE51474.2020.00013.
113. A. Imawan and J. Kwon, "A timeline visualization system for road traffic big data," 2015 IEEE International Conference on Big Data (Big Data), 2015, pp. 2928-2929, doi: 10.1109/BigData.2015.7364125.
114. M. P. Maharani, P. Tobianto Daely, J. M. Lee and D. -S. Kim, "Attack Detection in Fog Layer for IIoT Based on Machine Learning Approach," 2020 International Conference on Information and Communication Technology Convergence (ICTC), 2020, pp. 1880-1882, doi: 10.1109/ICTC49870.2020.9289380.
115. A. Kanawaday and A. Sane, "Machine learning for predictive maintenance of industrial machines using IoT sensor data," 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), 2017, pp. 87-90, doi: 10.1109/ICSESS.2017.8342870.
116. G. Abdelmoumin, D. B. Rawat and A. Rahman, "On the Performance of Machine Learning Models for Anomaly-Based Intelligent Intrusion Detection Systems for the Internet of Things," in IEEE Internet of Things Journal, vol. 9, no. 6, pp. 4280-4290, 15 March 2022, doi: 10.1109/JIOT.2021.3103829.
117. K. M. Ahmed, A. Imteaj and M. H. Amini, "Federated Deep Learning for Heterogeneous Edge Computing," 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), 2021, pp. 1146-1152, doi: 10.1109/ICMLA52953.2021.00187.
118. H. Kim, Y. Kim and H. Park, "Reducing Model Cost Based on the Weights of Each Layer for Federated Learning Clustering," 2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN), 2021, pp. 405-408, doi: 10.1109/ICUFN49451.2021.9528575.

119. S. N. Hashemipour et al., "Big Data Compression in Smart Grids via Optimal Singular Value Decomposition," 2020 IEEE Industry Applications Society Annual Meeting, 2020, pp. 1-8, doi: 10.1109/IAS44978.2020.9334900.
120. P. De, A. Chatterjee and A. Rakshit, "Regularized K-SVD-Based Dictionary Learning Approaches for PIR Sensor-Based Detection of Human Movement Direction," in IEEE Sensors Journal, vol. 21, no. 5, pp. 6459-6467, 1 March 2021, doi: 10.1109/JSEN.2020.3040228.
121. H. Dong, F. Cheng, M. Yang, W. Liu and X. Zhang, "On Low Rank Approximation Of 3-Tensors Based On Regularized t-SVD," 2019 Chinese Automation Congress (CAC), 2019, pp. 4827-4830, doi: 10.1109/CAC48633.2019.8997349.
122. F. Özyurt, H. Kutlu, E. Avci and D. Avci, "A New Method for Classification of Images Using Convolutional Neural Network Based on Dwt-Svd Perceptual Hash Function," 2018 3rd International Conference on Computer Science and Engineering (UBMK), 2018, pp. 410-413, doi: 10.1109/UBMK.2018.8566537.
123. E. A. Umoh and O. N. Iloanusi, "A Topology for Fingerprint Image Encryption based on HDWT-SVD and Hyperchaos," 2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON), 2022, pp. 1-5, doi: 10.1109/NIGERCON54645.2022.9803161.
124. A. H. Bentbib, K. Kreit and I. Labaali, "Randomized Tensor Singular Value Decomposition for Multidimensional Data Compression," 2022 11th International Symposium on Signal, Image, Video and Communications (ISIVC), 2022, pp. 1-6, doi: 10.1109/ISIVC54825.2022.9800729.
125. K. S. Singh, P. Singh and H. V. Singh, "Security of E-healthcare data using DWT-SVD based Hybrid watermarking," 2022 10th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-22), 2022, pp. 1-6, doi: 10.1109/ICETET-SIP-2254415.2022.9791599.
126. G. Xu et al., "TT-SVD: An Efficient Sparse Decision-Making Model With Two-Way Trust Recommendation in the AI-Enabled IoT Systems," in IEEE

- Internet of Things Journal, vol. 8, no. 12, pp. 9559-9567, 15 June 15, 2021, doi: 10.1109/JIOT.2020.3006066.
127. Di Chai, Leye Wang, Lianzhi Fu, Junxue Zhang, Kai Chen, Qiang Yang, "Federated Singular Vector Decomposition," arXiv:2105.08925v1 [cs.DC] 19 May 2021
128. Y. Xiaochen and L. Qicheng, "Parallel Algorithm of Improved FunkSVD Based on GPU," in IEEE Access, vol. 10, pp. 26002-26010, 2022, doi: 10.1109/ACCESS.2022.3156969.
129. Raharya, N.; She, C.; Hardjawana, W.; Vucetic, B. Deep Learning for Distributed User Association in Massive Industrial IoT Networks. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 2021, pp. 1-6. doi: 10.1109/WCNC49053.2021.9417303.
130. Wang, X.; Yang, L. T.; Wang, Y.; Ren, L.; Deen, M. J. ADTT: A Highly Efficient Distributed Tensor-Train Decomposition Method for IIoT Big Data. IEEE Transactions on Industrial Informatics, 2021, 17(3), 1573-1582. doi: 10.1109/TII.2020.2967768.
131. Yang, Y. et al., "ASTREAM: Data-Stream-Driven Scalable Anomaly Detection with Accuracy Guarantee in IIoT Environment," in IEEE Transactions on Network Science and Engineering, doi: 10.1109/TNSE.2022.3157730.
132. Liu, Y.; Garg, S.; Nie, J.; Zhang, Y.; Xiong, Z.; Kang, J.; Hossain, M. S. Deep Anomaly Detection for Time-series Data in Industrial IoT: A Communication-Efficient On-device Federated Learning Approach. IEEE Internet of Things Journal, 2020, 8(8), 6348-6358. doi:10.1109/JIOT.2020.3011726
133. Peiyong, Z.; Chao W.; Chunxiao J.; Zhu H. Deep Reinforcement Learning Assisted Federated Learning Algorithm for Data Management of IIoT. IEEE Transactions on Industrial Informatics 2021, 17(2), 8475-8484. doi:10.1109/tii.2021.3064351
134. Peiyong, Z.; Hao, S.; Jingyi S.; Chunxiao J.; Dongliang, X. Federated Transfer Learning for IIoT Devices With Low Computing Power Based on Blockchain and

- Edge Computing. IEEE Access 2021, 9, 98630-98638. doi:10.1109/access.2021.3095078
135. Xiao Guo; Xiang Li; Xiangyu Chang; Shusen Wang; Zhihua Zhang; Privacy-Preserving Distributed SVD via Federated Power, (),arXiv:2103.00704, 2023. <https://doi.org/10.48550/arXiv.2103.00704>
136. Chi Y.; Dong, Y.; Wang, J.; Yu, F. R.; Leung, V. C. M. Knowledge-Based Fault Diagnosis in Industrial Internet of Things: A Survey. IEEE Internet of Things Journal 2022, 9(15), 12886-12900. doi: 10.1109/JIOT.2022.3163606.
137. Chen, H.; Rodrigues, J. J.; Xia, F.; Das, S. Distributed Intelligence Over Internet of Things. IEEE Transactions on Industrial Informatics, 2022, 18(9), 6233-6235. doi: 10.1109/TII.2022.3162306.
138. Ma, C.; Zheng, M.; Liang, W.; Kasparick, M.; Lin, Y. Deploying Hierarchical Mesh Networks for Supporting Distributed Computing in Industrial Internet of Things. IEEE Systems Journal, 2022, 16(3), 4433-4444. doi:10.1109/JSYST.2022.3153339.
139. Kaur, M.; Khan, M. Z.; Gupta, S.; Alsaeedi, A. Adoption of Blockchain With 5G Networks for Industrial IoT: Recent Advances, Challenges, and Potential Solutions. IEEE Access 2022, 10, 981-997. doi: 10.1109/ACCESS.2021.3138754.
140. Hashemipour, N., et al. Optimal Singular Value Decomposition Based Big Data Compression Approach in Smart Grids. IEEE Transactions on Industry Applications 2021, 57(4), 3296-3305. doi: 10.1109/TIA.2021.3073640.
141. Anwar, A.; Abu Adnan Abir, S. M. Measurement Unit Placement Against Injection Attacks for the Secured Operation of an IIoT-Based Smart Grid. In Proceedings of the 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Guangzhou, China, 2020, pp. 767-774. doi: 10.1109/TrustCom50675.2020.00105.
142. Saigopal, V. V. R. G.; Raju, V. IIoT Digital Forensics and Major Security issues. In Proceedings of the 2020 International Conference on Computational Intelligence (ICCI), 2020, pp 233-236. doi:10.1109/ICCI51257.2020.9247685
143. Huo, R. et al. A Comprehensive Survey on Blockchain in Industrial Internet of Things: Motivations, Research Progresses, and Future Challenges. IEEE

- Communications Surveys & Tutorials 2022, 24(1), 88-122. doi: 10.1109/COMST.2022.3141490.
144. Lee, Y. ; Lee, D.; Oh, J.; Park, L.; Na, W.; Cho, S. Endorsement Policy for Industrial Internet of Thing with Private Blockchain. In Proceedings of the 2021 International Conference on Information and Communication Technology Convergence (ICTC), 2021, pp. 449-452. doi: 10.1109/ICTC52510.2021.9620893.
145. Al-Gumaei, K.; et al. A Survey of Internet of Things and Big Data integrated Solutions for Industrie 4.0. In Proceedings of the 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), 2018, pp. 1417-1424. doi: 10.1109/ETFA.2018.8502484.
146. Ren, L. ; Liu, Y.; Wang, X.; Lü J.; Deen, M. J. Cloud–Edge-Based Lightweight Temporal Convolutional Networks for Remaining Useful Life Prediction in IIoT. IEEE Internet of Things Journal 2021, 8(16), 12578-12587. doi: 10.1109/JIOT.2020.3008170.
147. Jaradat, Y.; Masoud, M.; Jannoud, I.; Manasrah, A.; Alia, M. A Tutorial on Singular Value Decomposition with Applications on Image Compression and Dimensionality Reduction. In Proceedings of the 2021 International Conference on Information Technology (ICIT), 2021, pp. 769-772, doi: 10.1109/ICIT52682.2021.9491732.
148. Reddy, B. S. ; Adum Babu, I.; Bachu, S. Implementation of Medical Image Watermarking using RDWT and SVD for Secure Medical Data Transmission in Healthcare Systems. In Proceedings of the 2022 First International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT), 2022, pp. 1-6. doi: 10.1109/ICEEICT53079.2022.9768429.
149. Di, C.; Leye, W.; Lianzhi, F.; Junxue, Z.; Kai, C.; Qiang, Y. Federated Singular Vector Decomposition. arXiv:2105.08925v1 [cs.DC] 19 May 2021.
150. J. Lee, S. Kim, G. Lebanon, and Y. Singer, “Local low-rank matrix approximation,” in Proc. Int. Conf. Mach. Learn., 2013, pp. 82–90

151. Huang T, Zhao R, Bi L, et al. Neural embedding singular value decomposition for collaborative filtering[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 33(10): 6021-6029.
152. S. Sedhain, A. K. Menon, S. Sanner, and L. Xie, "AutoRec: Autoen-coders meet collaborative filtering," in Proc. 24th Int. Conf. World Wide Web, 2015, pp. 111–112.
153. Y. Zheng, B. Tang, W. Ding, and H. Zhou, "A neural autoregressive approach to collaborative filtering," 2016, arXiv:1605.09477. [Online]. Available: <http://arxiv.org/abs/1605.09477>
154. Q. Li, X. Zheng, and X. Wu, "Neural collaborative autoencoder," 2017, arXiv:1712.09043. [Online]. Available: <http://arxiv.org/abs/1712.09043>
155. MN Hassan Reza, C. Agamudai Nambi Malarvizhi, S. Jayashree and M. Mohiuddin, "Industry 4.0–Technological Revolution and Sustainable Firm Performance," 2021 Emerging Trends in Industry 4.0 (ETI 4.0), Raigarh, India, 2021, pp. 1-6, doi: 10.1109/ETI4.051663.2021.9619363.
156. S. Kumar, SK Chandra, RN Shukla and L. Panigrahi, "Industry 4.0 based Machine Learning Models for Anomalous Product Detection and Classification," 2022 OPJU International Technology Conference on Emerging Technologies for Sustainable Development (OTCON), Raigarh, Chhattisgarh, India, 2023, pp. 1-6, doi: 10.1109/OTCON56053.2023.10114045.
157. P. Kundu, X. Luo and Y. Qin, "A Computationally Efficient Machine Learning Strategy for Automatic Identification of Process and Product Fingerprints for Smart Manufacturing Systems," 2022 8th International Conference on Nanomanufacturing & 4th AET Symposium on ACSM and Digital Manufacturing (Nanoman -AETS), Dublin, Ireland, 2022, pp. 1-5, doi: 10.1109/Nanoman-AETS56035.2022.10119508.
158. LAC Salazar and B. Vogel-Heuser, "Industrial Artificial Intelligence: A Predictive Agent Concept for Industry 4.0," 2022 IEEE 20th International Conference on Industrial Informatics (INDIN), Perth, Australia, 2022, pp. 27-32, doi: 10.1109/INDIN51773.2022.9976159.

159. R. Sinha, S. Patil, L. Gomes and V. Vyatkin, "A Survey of Static Formal Methods for Building Dependable Industrial Automation Systems," in *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 3772-3783, July 2019, doi: 10.1109/TII.2019.2908665.
160. S. Karnouskos, P. Leitao, L. Ribeiro and AW Colombo, "Industrial Agents as a Key Enabler for Realizing Industrial Cyber-Physical Systems: Multiagent Systems Entering Industry 4.0," in *IEEE Industrial Electronics Magazine*, vol. 14, no. 3, pp. 18-32, Sept. 2020, doi: 10.1109/MIE.2019.2962225.
161. X. Wang, LT Yang, Y. Wang, L. Ren and MJ Deen, "ADTT: A Highly Efficient Distributed Tensor-Train Decomposition Method for IIoT Big Data," in *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 1573-1582, March 2021, doi: 10.1109/TII.2020.2967768.
162. M. Aloqaily, I. Al Ridhawi, F. Karray and M. Guizani, "Towards Blockchain-based Hierarchical Federated Learning for Cyber-Physical Systems," 2022 International Balkan Conference on Communications and Networking (BalkanCom), Sarajevo, Bosnia and Herzegovina, 2022, pp. 46-50, doi: 10.1109/BalkanCom55633.2022.9900546.

ДОДАТОК А. АКТИ ВИКОРИСТАННЯ ТА ВПРОВАДЖЕННЯ

"ЗАТВЕРДЖУЮ"

Проректор з наукової роботи
Національного університету
«Львівська політехніка»

_____ Іван ДЕМИДОВ

«___» _____ 2024 р.

АКТ

використання наукових результатів дисертаційної роботи

Гордійчук-Бублівської Олени Василівни

«Методи та засоби опрацювання великих даних в розподілених інформаційних системах»,
представленої на здобуття наукового ступеня доктора філософії за спеціальністю 122 –
комп'ютерні науки при виконанні держбюджетної НДР

Комісія в складі: голови комісії – начальника науково-дослідної частини, д.т.н., ст. досл. Небесного Р. В. та членів комісії – завідувача кафедри телекомунікацій, д.т.н., професора Климаша М.М., завідувача відділу науково-організаційного супроводу наукових досліджень, к.т.н. Лазько Г. В. і в. о. заступника начальника планово-фінансового відділу Фаст І. І., цим актом підтверджують, що результати дисертаційної роботи Гордійчук-Бублівської О. В., використовувалися при виконанні держбюджетної НДР: «Розроблення інноваційних методів та моделей побудови інтелектуальних інформаційно-комунікаційних систем для цифровізації промисловості» (номер державної реєстрації 0122U000817), в якій О. В. Гордійчук-Бублівською було обґрунтовано підходи до вдосконалення існуючих та створення нових методів та підходів до оброблення великих даних, що дозволяє підвищувати ефективність роботи сучасних інформаційних систем, швидше приймати рішення про управління обслуговування користувачів.

Голова комісії:

начальник науково-дослідної
частини, д.т.н., ст. досл.

_____ Роман НЕБЕСНИЙ

Члени комісії:

зав. відділу науково-організаційного
супроводу наукових досліджень, к.т.н.

_____ Галина ЛАЗЬКО

в. о. заст. нач. планово-фінансового відділу

_____ Ірина ФАСТ

зав. каф. телекомунікацій,
д.т.н., проф.

_____ Михайло КЛИМАШ

ЗАТВЕРДЖУЮ
Директор ВАТ „ УКРТЕЛЕКОМ ”
_____ Тарас АНДРУХІВ
« ____ » _____ 2024 р.

АКТ
про впровадження результатів
дисертаційної роботи аспіранта кафедри «Автоматизовані системи
управління»
Національного університету «Львівська політехніка»
Гордійчук-Бублівської Олени Василівни

Цей акт підтверджує, що результати дисертаційної роботи на здобуття наукового ступеня доктора філософії Гордійчук-Бублівської О. В., були використані та впроваджені в практичну діяльність при розробці рекомендаційної системи для надання персональних рекомендацій клієнтам. Автором було розроблено методи та побудовано рекомендаційну систему на основі алгоритмів Funk SVD та Fed SVD.

Терміни проведення досліджень: вересень 2023 р. - грудень 2023 р.

Впровадження отриманих результатів дисертаційного дослідження Оленою Гордійчук-Бублівською полягає у наступному:

- аналіз ефективності роботи існуючих рекомендаційних систем;
- розробка методів оброблення великих даних у системах надання рекомендацій клієнтам;
- реалізація та апробація розроблених методів.

Використання розроблених у дисертаційному дослідженні методів та засобів дало змогу побудувати та реалізувати інтелектуальну рекомендаційну систему для роботи з клієнтами.

Члени комісії:

д.т.н., доцент кафедри ТК _____

Тарас МАКСИМЮК

д.т.н., доцент кафедри ТК _____

Микола БЕШЛЕЙ

ЗАТВЕРДЖУЮ

Директор ТОВ „ ДІДЖИТАЛ-
ТЕХНОЛОДЖІ ЛТД ”

_____ Андрій ТРУБІЩИН
« ___ » _____ 2024 р.

АКТ

**про впровадження результатів дисертаційної роботи
аспіранта кафедри «Автоматизовані системи управління»
Національного університету «Львівська політехніка»
Гордійчук-Бублівської Олени Василівни**

Цей акт підтверджує, що результати дисертаційної роботи на здобуття наукового ступеня доктора філософії Гордійчук-Бублівської О. В., були використані та впроваджені в практичну діяльність при розробці рекомендаційної системи для надання персональних рекомендацій клієнтам. Автором було розроблено методи та побудовано рекомендаційну систему, що проводить аналіз поведінки користувачів та визначає для них оптимальні послуги.

Терміни проведення досліджень: серпень 2023 р. - вересень 2023 р.

Впровадження отриманих результатів дисертаційного дослідження Оленою Гордійчук-Бублівською полягає у наступному:

- аналіз існуючих методів надання рекомендацій;
- розробка методів надання рекомендацій користувачам на основі їх поведінки;
- реалізація та оцінка ефективності роботи розроблених методів.

Використання розроблених методів та засобів дало змогу побудувати та реалізувати рекомендаційну систему для роботи з даними про користувачів.

Члени комісії:

д.т.н., доцент кафедри ТК

Тарас МАКСИМЮК

д.т.н., доцент кафедри ТК

Микола БЕШЛЕЙ

ЗАТВЕРДЖУЮ
Директор ТзОВ „ГІГАЦЕНТР УКРАЇНА”

Назарій КУРОЧКО
« ____ » _____ 2024 р.

АКТ
про впровадження результатів
дисертаційної роботи аспіранта кафедри «Автоматизовані системи
управління»
Національного університету «Львівська політехніка»
Гордійчук-Бублівської Олени Василівни

Цей акт підтверджує, що результати дисертаційної роботи на здобуття наукового ступеня доктора філософії Гордійчук-Бублівської О. В., були використані та впроваджені в практичну діяльність підприємства при аналізі даних користувачів та надання їм персональних рекомендацій. Автором запропоновано методи та модель рекомендаційної систем на основі модифікованого алгоритму Fed SVD.

Терміни проведення досліджень: серпень 2023 р. - жовтень 2023 р.

Впровадження отриманих результатів дисертаційного дослідження Оленою Гордійчук-Бублівською полягає у наступному:

- аналіз ефективності роботи існуючих методів надання рекомендацій користувачам;
- розробка нових методів та моделі для аналізу даних та надання рекомендацій клієнтам про відповідні їм сервіси;
- реалізація та оцінка ефективності роботи запропонованих методів та моделі.

Використання розроблених у дисертаційному дослідженні методів та засобів дало змогу покращити ефективність формування персональних рекомендацій користувачам системи.

Члени комісії:

д.т.н., доцент кафедри ТК

Тарас МАКСИМЮК

д.т.н., доцент кафедри ТК

Микола БЕШЛЕЙ

ЗАТВЕРДЖУЮ

Директор ТзОВ „ МАКСІТЕХ”

_____ Сергій ЗАБЛОЦЬКИЙ

« ____ » _____ 2024 р.

АКТ**про впровадження результатів
дисертаційної роботи аспіранта кафедри «Автоматизовані системи
управління»****Національного університету «Львівська політехніка»
Гордійчук-Бублівської Олени Василівни**

Цей акт підтверджує, що результати дисертаційної роботи на здобуття наукового ступеня доктора філософії Гордійчук-Бублівської О. В., були використані та впроваджені в практичну діяльність при розробці рекомендаційної системи для аналізу даних про клієнтів та надання їм персональних рекомендацій. Автором було розроблено методи надання рекомендацій, що дозволяють проводити аналіз поведінки користувачів та визначати перелік відповідних послуг.

Терміни проведення досліджень: серпень 2023 р. - листопад 2023 р.

Впровадження отриманих результатів дисертаційного дослідження Оленою Гордійчук-Бублівською полягає у наступному:

- аналіз існуючої системи надання рекомендацій;
- розробка методів надання персональних рекомендацій користувачам;
- впровадження в роботу існуючої рекомендаційної системи розроблених методів та аналіз їх ефективності.

Використання розроблених у дисертаційному дослідженні методів та засобів дало змогу реалізувати рекомендаційну систему для ефективного обчислення рекомендацій.

Члени комісії:

д.т.н., доцент кафедри ТК _____

Тарас МАКСИМЮК

д.т.н., доцент кафедри ТК _____

Микола БЕШЛЕЙ

ЗАТВЕРДЖУЮ

Проректор з науково-педагогічної роботи
Національного університету
«Львівська політехніка»_____ Олег ДАВИДЧАК
« ___ » _____ 2024 р.**АКТ****про впровадження в навчальний процес результатів
дисертаційної роботи на здобуття наукового ступеня доктора філософії
Гордійчук-Бублівської Олени Василівни**

Цей акт складений про те, що результати дисертаційної роботи Гордійчук-Бублівської О. В. на тему «Методи та засоби опрацювання великих даних в розподілених інформаційних системах», представленої на здобуття наукового ступеня доктора філософії, використовуються у навчальному процесі кафедри «Телекомунікації» Національного університету «Львівська політехніка». Матеріали дисертаційного дослідження використовуються під час викладання дисциплін «Програмні платформи інформаційно-комунікаційних систем».

Зокрема, у навчальному процесі використовуються запропоновані О. В. Гордійчук-Бублівською:

- Методи аналізу та моделювання роботи систем оброблення великих даних (дисципліна «Програмні платформи інформаційно-комунікаційних систем» для студентів освітньо-кваліфікаційного рівня «бакалавр», що навчаються за спеціальністю 126 «Інформаційні системи та технології», з галузі знань 12 «Інформаційні технології»);

- Методи побудови програмних моделей для оптимізації великих даних (дисципліна «Програмні платформи інформаційно-комунікаційних систем» для студентів освітньо-кваліфікаційного рівня «бакалавр», що навчаються за спеціальністю 126 «Інформаційні системи та технології», з галузі знань 12 «Інформаційні технології»).

Директор ІТРЕ,
д.т.н., професор

Богдан СТРИХАЛЮК

Завідувач кафедри ТК,
д.т.н., професор

Михайло КЛИМАШ

Професор кафедри ТК,
д.т.н., професор

Микола КАЙДАН

