**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**
**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ**
**імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

КАФЕДРА СИСТЕМНОГО ПРОГРАМУВАННЯ І
СПЕЦІАЛІЗОВАНИХ КОМП'ЮТЕРНИХ СИСТЕМ

«На правах рукопису»
УДК __004.855_____

«До захисту допущено»
Завідувач кафедри СПСКС

_____ В.П.Тарасенко_
(підпис) (ініціали, прізвище)
"___"_____2018р

# Магістерська дисертація

**на здобуття ступеня магістра**

зі спеціальності 123 Комп'ютерна інженерія
Системне програмування
на тему: **"Система рекомендацій з використанням соціальних мереж"**

Виконала: студентка II курсу, групи КВ-з71мп
(шифр групи)

*Чорна Катерина Юріївна*
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник: кандидат технічних наук, доцент

*Замятін Денис Станіславович*
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Рецензент _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студентка _____
(підпис)

Київ – 2018 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**
**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ**
**імені ІГОРЯ СІКОРСЬКОГО»**

Факультет прикладної математики

Кафедра системного програмування і спеціалізованих комп'ютерних систем

Рівень вищої освіти – другий (магістерський)
Спеціальність 123 Комп'ютерна інженерія
Системне програмування

ЗАТВЕРДЖУЮ
Завідувач кафедри СПСКС
_____ _В.П.Тарасенко__
(підпис) (ініціали, прізвище)
«___»_____2018р.

**ЗАВДАННЯ**
**на магістерську дисертацію студентки**
Чорній Катерині Юріївні

1. Тема дисертації: Система рекомендацій з використанням соціальних мереж,
науковий керівник дисертації: к.т.н., доцент Замятін Д.С. ,
затверджені наказом по університету від « 09» листопада 2018 р. № 4138-с.

2. Термін подання студентом дисертації: __ грудня 2018 р.

3. Об'єкт дослідження: методи та алгоритми формування систем рекомендацій фільмів на основі користувацьких оцінок та відгуків.

4. Предмет дослідження: програмна реалізація розробленого алгоритму генерації ландшафту в ігрових програмах.

5. Перелік завдань, які потрібно розробити:
- розглянути основні методи формування систем рекомендацій;
- провести порівняльний аналіз методів формування рекомендацій;
- дослідити підходи оптимізації існуючих алгоритмів;
- розробити та описати етапи обраного алгоритму рекомендації фільмів на основі користувацьких оцінок та відгуків;
- програмно реалізувати розроблений алгоритм рекомендації фільмів;
- провести експерименти і оцінити роботу оптимізованого алгоритму.

6. Перелік ілюстративного матеріалу:

- Структурна схема архітектури програмної системи;
- Презентація.

7. Перелік публікацій: IV Міжнародна науково-технічна конференція «Сучасні методи, інформаційне, програмне та технічне забезпечення систем керування організаційно-технічними та технологічними комплексами» (Київ, 22-23 листопада 2018 р.), XXIV Міжнародна інтернет-конференція «Новини науки XXI століття» (Вінниця, 23 листопада 2018 р.)

8. Дата видачі завдання 5 вересня 2017 р.

Календарний план

| № з/п | Назва етапів виконання магістерської дисертації | Термін виконання етапів магістерської дисертації | Примітка |
|---|---|---|---|
| 1 | Ознайомлення з предметною галуззю | 20.12.2017 | |
| 2 | Визначення структури магістерської дисертації; вивчення літератури, пошук додаткової літератури, патентний пошук | 07.02.2018 | |
| 3 | Робота над першим розділом магістерської дисертації; проведення наукового дослідження | 21.03.2018 | |
| 4 | Проведення наукового дослідження; робота над другим розділом магістерської дисертації; розроблення програмного забезпечення | 29.06.2018 | |
| 5 | Проведення наукового дослідження; робота над статтею за результатами наукового дослідження | 15.09.2018 | |
| 6 | Проведення наукового дослідження; робота над третім розділом магістерської дисертації; підготовка матеріалів доповіді на IV Міжнародну науково-технічну Internet-конференцію «Сучасні методи, інформаційне, програмне та технічне забезпечення систем керування організаційно-технічними та технологічними комплексами» та XXIV Міжнародну інтернет-конференцію «Новини науки XXI століття» | 04.10.2018 | |
| 7 | Завершення роботи над основною частиною магістерської дисертації; підготовка ілюстративного матеріалу | 17.10.2018 | |
| 8 | Оформлення текстової і графічної частини магістерської дисертації | 11.11.2018 | |
| 9 | Попередній розгляд магістерської дисертації на кафедрі | 26.11.2018 | |

Студент _____ Чорна К.Ю.

Науковий керівник дисертації _____ Замятін Д.С.

# РЕФЕРАТ

**Актуальність теми.** Об’єм даних у світі зростає, і швидкість його зростання перевищує швидкість зростання ресурсів, необхідних для обробки цих даних. Сьогодні людство повністю поглинуте інформаційним простором: фільми, книги, статті, новини тощо, і важко визначити, який продукт більше підходить конкретній людині. Відповідно, з’явилась потреба в технологіях, які можуть оперативно обробляти великі обсяги даних та виділяти лише ту інформацію, що є корисною для конкретного користувача. Одними з найпотужніших технологій вирішення таких задач є рекомендаційні системи.

Рекомендаційні системи – це системи, які визначають переваги та інтереси користувачів і надають відповідні рекомендації відносно цих даних. Такими даними можуть бути оцінки користувачів, приватна інформація, комунікація в соціальних мережах, географічне розташування користувачів тощо.

Протягом останніх років багато досліджень проводилося з метою аналізу та класифікації текстів і даних, та подальшою класифікацією на категорії на основі навченої моделі. Однак з’являється все більше досліджень щодо аналізу текстових даних для визначення того, як людина репрезентує свої «почуття» відносно певного предмету чи інформації. Це призвело до розвитку аналізу тональності відгуків користувачів та систем їх класифікації. Аналіз та класифікація тональностей відгуків використовують складні алгоритми, оскільки думки можна висловити тонкими та складними методами, включаючи використання неформальної мови (сленг), неоднозначності, іронії, гумору та акценту.

Аналіз та класифікацію почуттів почали розробляти з багатьох причин. Наприклад, для відстеження зросту та спаду популярності певного продукту, для порівняння ставлень онлайн-клієнтів до ряду продуктів тощо.

**Об'єктом дослідження** є методи та алгоритми формування систем рекомендацій фільмів на основі користувацьких оцінок та відгуків.

**Предметом дослідження** є рекомендаційні системи та методи аналізу тональності відгуків користувачів соціальних мереж.

**Метою дослідження** є підвищення ефективності методів аналізу даних на основі тональності відгуків користувачів соціальних мереж, що, в свою чергу, дозволяє поліпшити роботу рекомендаційної системи фільмів на основі аналізу текстових відгуків користувачів соціальних мереж.

**Методи дослідження**:

- аналіз;
- аналогія;
- дедукція;
- емпіричні: експериментальна перевірка ефективності розробленого програмного забезпечення;
- індукція;
- синтез.

**Результати та їх наукова новизна**:

1. Проаналізовано існуючі рекомендаційні системи та рішення аналізу тональності тексту.
2. Запропоновано новий метод генерації рекомендацій для користувачів.
3. Покращено алгоритм формування рекомендацій на основі поєднання аналізу тональності користувацьких відгуків та класичних алгоритмів.

**Галузь застосування**. Результати даної роботи можуть бути використані для покращення рекомендаційних систем шляхом визначення та врахування тональності коментарів користувачів.

**Практична цінність** отриманих в роботі результатів полягає в тому, що запропонований алгоритм формування рекомендаційної системи дозволяє ефективно виконувати відносно швидку обробку великих об’ємів даних і на їх основі формувати список рекомендацій кінцевому користувачеві. Також використання аналізу тональності відгуків із соціальних мереж дозволяє робити списки рекомендацій більш точними, що суттєво вплине на якість інформації, яку отримує кінцевий користувач.

Результати роботи розробленого алгоритму дозволяють точніше фільтрувати дані та ефективно використовувати ресурси і пам'ять мобільних пристроїв, що підвищує якість програмного продукту. А, відповідно, і його подальший розвиток та поширення.

**Апробація роботи.** Основні положення і результати роботи були представлені та обговорювались на V Міжнародній науково-технічній конференції «Сучасні методи, інформаційне, програмне та технічне забезпечення систем керування організаційно-технічними та технологічними комплексами» (Київ, 22-23 листопада 2018 р.), а також на XXIV Міжнародній інтернет-конференції «Новини науки XXI століття» (Вінниця, 23 листопада 2018 р.).

**Структура та обсяг роботи.** Магістерська дисертація складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

У вступі подано загальну характеристику роботи, зроблено оцінку сучасних методів формування рекомендацій, обґрунтовано актуальність напряму досліджень, сформульовано мету і задачі досліджень, показано наукову новизну отриманих результатів і практичну цінність роботи.

У першому розділі проаналізовано існуючі методи аналізу даних та формування рекомендацій на їх основі. Розглянуто переваги та недоліки платформ для обробки даних та можливість їх використання для мобільних

платформ. Проаналізовано вплив тональності відгуків користувачів на формування списків рекомендацій.

У другому розділі запропоновано підхід до побудови системи рекомендації на базі Spark із урахуванням аналізу тональності відгуків користувачів.

У третьому розділі досліджено поєднання класичних методів, що використовуються у сучасних рекомендаційних системах, та методу аналізу тональності коментарів користувачів соціальних мереж. Сформовано пропозицію побудови попереднього та фінального списку рекомендацій.

У висновках проаналізовано отримані результати роботи.

У додатках наведено презентацію, лістинг розробленого програмного продукту, копії публікацій та довідка про впровадження.

Магістерська дисертація виконана на 80 аркушах, містить 4 додатки та посилання на список використаних літературних джерел з 51 найменувань. У роботі наведено 7 рисунків та 3 таблиці.

**Ключові слова**: аналіз тональності, машинне навчання, рекомендаційна система, big data, Spark.

## ABSTRACT

**Theme urgency.** The volume of data in the world is growing, and its rate of growth exceeds the rate of growth of resources that are necessary for the processing such big amounts of data. Today mankind is completely absorbed by the information space: films, books, articles, news, etc., and it is difficult to determine which product is more suitable for a particular person. Therefore, there is a need for technologies that can quickly process big volumes of data and allocate only information that is useful to a particular user. One of the most powerful technologies for solving this kind of problems is recommendation systems.

Recommendation systems are systems that determine preferences and interests of users and provide relevant guidance based on these data. Such data may include user ratings, private information, social networking, geographic location, etc.

In recent years, many studies have been conducted to analyze and classify texts and data, and further categorize them based on the trained model. However, more and more research is being done on the analysis of text data in order to determine how a person represents his "feelings" about a particular subject or information. This led to the development of a sentiment analysis and its classification systems. Success analysis and classification uses complex algorithms, because thoughts can be expressed in a subtle and complex way, including the usage of informal language (slang), ambiguity, irony, humor and accent. The analysis and classification of feelings began to develop for several reasons. For example, to track the growth of popularity or a denial of a particular product, to compare online customer position to a number of products, and more.

**Object of research** is the methods and algorithms of creating movie recommendation systems based on user ratings and reviews.

**Subject of research** is recommendation systems and sentiment analysis methods that analyze the connotation of feedback from social networks users.

**Research objective** is to improve the recommendation system of films based on the analysis of text responses of social networks users and to offer solutions for increasing the effectiveness of data analysis methods based on the connotation of feedback from social networks users.

**Research methods**:

• analysis;

• analogy;

• deduction;

• empirical: experimental verification of the effectiveness of the software developed;

• induction;

• synthesis

**Results and scientific novelty:**

1. The existing reference systems and the analysis of the tone of the text are analyzed.

2. A new method of generating recommendations for users is proposed.

3. The algorithm of recommendation generation is improved on the basis of a combination of sentiment analysis of user reviews and classical algorithms.

Field of application. The results of this work can be used to improve advisory systems by defining and taking into account the tone of user comments.

**Practical value** of the results obtained in the work is that the proposed algorithm for the formation of the advisory system allows efficiently execute relatively fast processing of large volumes of data and, on the basis of them, to formulate a list of recommendations to the end user. Also, the use of tone feedback analysis from social networks allows making lists of recommendations more precise, which will significantly affect the quality of information received by the end user.

The results of the developed algorithm allow more accurate data filtering and efficient use of resources and memory of mobile devices, which enhances the quality of the software product. And, accordingly, and its further development and distribution.

**Approbation**. The main provisions and results of the work were presented and discussed at the V International Scientific and Technical Conference "Modern methods, information, software and technical support of control systems for organizational, technological and technological complexes" (Kyiv, November 22-23, 2018), as well as at XXIV International Internet Conference "Science News of the 21st Century" (Vinnytsya, November 23, 2018).

**Structure and content of the thesis**. The master's dissertation consists of an introduction, three sections, conclusions, list of used sources and applications.

The introduction provides a general description of the work, assesses the modern methods of forming recommendations, substantiates the relevance of the research direction, formulates the purpose and objectives of the research, shows the scientific novelty of the results obtained and the practical value of the work.

In the first chapter the existing methods of data analysis and the creation of recommendations on their basis are analyzed. The advantages and disadvantages of platforms for data processing and the possibility of their use for mobile platforms are considered. The influence of the response of users on the formation of recommendations lists has been analyzed.

In the second chapter it is proposed to follow an approach to building a Spark-based recommendation system based on the analysis of the responsiveness of user feedback.

In the third chapter it is explored the combination of classical methods used in modern advisory systems, and a method for analyzing the tone of comments of users of social networks. The proposal for constructing the preliminary and final list of recommendations has been formed.

In the conclusions the results of work are analyzed.

In the appendixes following items are included: a presentation, a listing of the software product developed, a copy of the publications, and a certificate of implementation.

The thesis is presented in 80 pages, it contains 4 appendixes and 51 references to the used information sources. 7 figures and 3 tables are given in the thesis.

**Key words:** tonal analysis, machine learning, reference system, big data, spark.

# РЕФЕРАТ

**Актуальность темы.** Объем данных в мире растет, и скорость его роста превышает скорость роста ресурсов, необходимых для обработки этих данных. Сегодня человечество полностью поглощено информационным пространством: фильмы, книги, статьи, новости и т.д., и трудно определить, какой продукт больше подходит конкретному человеку. Соответственно, появилась потребность в технологиях, которые могут оперативно обрабатывать большие объемы данных и выделять только ту информацию, которая является полезной для конкретного пользователя. Одними из самых мощных технологий решения таких задач являются рекомендательные системы.

Рекомендательные системы — это системы, которые определяют преимущества и интересы пользователя и обеспечивают соответствующие списки рекомендаций относительно этих данных. Такими данными могут быть оценки пользователей, частная информация пользователя, коммуникация в социальных сетях, географическое расположение пользователей и тому подобное.

В последние годы много исследований проводилось с целью анализа и классификации текстов и данных, и последующей классификации их на категории на основе обученной модели. Однако появляется все больше исследований по анализу текстовых данных для определения того, как человек представляет свои «чувства» относительно определенного предмета или информации. Это привело к развитию анализа тональности отзывов и систем их классификации.

Анализ и классификация тональности отзывов используют сложные алгоритмы, поскольку мысли можно выразить тонкими и сложными

методами, включая использование неформальной речи (сленг), неоднозначности, иронии, юмора и акцента.

Анализ и классификацию чувств начали исследовать по нескольким причинам. Например, для отслеживания роста и спада популярности определенного продукта, для сравнения отношений онлайн-клиентов к ряду продуктов и тому подобное.

**Объектом исследования** являются методы и алгоритмы формирования систем рекомендаций фильмов на основе пользовательских оценок и отзывов.

**Предметом исследования** является рекомендательные системы и методы анализа тональности отзывов пользователей социальных сетей.

**Целью исследования** является улучшить работу рекомендательной системы фильмов на основе анализа текстовых отзывов пользователей социальных сетей и предложить решения для повышения эффективности методов анализа данных на основе тональности отзывов пользователей социальных сетей.

**Методы исследования**:

• анализ;

• аналогия;

• дедукция;

• эмпирические: экспериментальная проверка эффективности разработанного программного обеспечения;

• индукция;

• синтез.

**Результаты и их научная новизна**:

1. Проанализированы существующие рекомендательные системы и решения анализа тональности текста.

2. Предложен новый метод генерации рекомендаций для пользователей.

3. Улучшен алгоритм формирования рекомендаций на основе сочетания анализа тональности пользовательских отзывов и классических алгоритмов.

**Область применения**. Результаты данной работы могут быть использованы для улучшения рекомендательных систем путем определения и учета тональности комментариев пользователей.

**Практическая ценность** полученных в работе результатов заключается в том, что предложенный алгоритм формирования рекомендательной системы позволяет эффективно выполнять относительно быструю обработку больших объемов данных и на их основе формировать список рекомендаций конечному пользователю. Также использование анализа тональности отзывов из социальных сетей позволяет делать списки рекомендаций более точными, что существенно повлияет на качество информации, которую получает конечный пользователь.

Результаты работы разработанного алгоритма позволяют точнее фильтровать данные и эффективно использовать ресурсы и память мобильных устройств, повышает качество программного продукта. А, соответственно, его дальнейшее развитие и распространение.

**Апробация работы**. Основные положения и результаты работы были представлены и обсуждались на V Международной научно-технической конференции «Современные методы, информационное, программное и техническое обеспечение систем управления организационно-техническими и технологическими комплексами» (Киев, 22-23 ноября 2018), а также на XXIV

международной интернет-конференции «Новости науки XXI века» (Винница, 23 ноября 2018).

**Структура и объем работы**. Магистерская диссертация состоит из введения, трех глав, заключения, списка использованных источников и приложений.

Во вступлении представлена общая характеристика работы, произведена оценка современных методов формирования рекомендаций, обоснована актуальность направления исследований, сформулированы цели и задачи исследований, показано научную новизну полученных результатов и практическую ценность работы.

В первом разделе проанализированы существующие методы анализа данных и формирования рекомендаций на их основе. Рассмотрены преимущества и недостатки платформ для обработки данных и возможность их использования для мобильных платформ. Проанализировано влияние тональности отзывов пользователей на формирование списков рекомендаций.

Во втором разделе предложен подход к построению системы рекомендации на базе Spark с учетом анализа тональности отзывов пользователей.

В третьем разделе исследовано сочетание классических методов, используемых в современных рекомендательных системах, и метода анализа тональности комментариев пользователей социальных сетей. Сформирован предложение построения предварительного и финального списка рекомендаций.

В выводах проанализированы полученные результаты работы.

В приложениях приведены презентация, листинг разработанного программного продукта, копии публикаций и справка о внедрении.

Работа представлена на 80 страницах, содержит 4 приложения и ссылки на список использованных литературных источников из 51 наименования. В работе приведены 7 рисунков и 3 таблицы.

**Ключевые слова**: анализ тональности, машинное обучение, рекомендательная система, big data, Spark.

ЗМІСТ

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ALS – альтернативна найменша площина (alternating least squares)

GPS – глобальна радіонавігаційна супутникова система, яка надає GPS-приймачу геолокацію та часову інформацію будь-де на Землі або поблизу неї, де є чотири чи більше GPS-супутників без перешкод.

Hadoop – відкрита програмна платформа і фреймворк для організації розподіленого зберігання та обробки великих об'ємів даних на основі моделі MapReduce, при якій завдання розбивається на окремі фрагменти, кожен з яких запускається на окремому вузлі(ноді) кластера із серійних комп'ютерів.

ME (англ. maximum entropy) – максимальна ентропія

MF (англ. matrix factorization) – це клас алгоритму колаборативної фільтрації, що використовується у рекомендаційних системах. Алгоритми матричної факторизації працюють шляхом розбиття матриці елемента користувацької взаємодії на дві прямокутні матриці меншого розміру.

RDD (англ. resilient distributed dataset) – стійкі набори даних

(probabilistic matrix factorization, PMF)

Spark – високопродуктивний рушій для оброблення даних, що зберігаються в кластері Hadoop.

SVD – сингулярний розклад матриці

SVM (support vector machine, SVM)

TF-IDF (англ. term frequency, inverse document frequency) —показник, який застосовується з метою оцінки важливості слова у контексті тексту.

Байєсівські мережі

FM – Факторизаційні машини

КМ – Компютерна мережа

КС – Компютерна система

НМ – Нейронна мережа

ОС – Операційна система

ПЗ – Програмне забезпечення

# ВСТУП

Популярність мобільних пристроїв робить щоденне життя людей більш залежним від мобільних послуг. Люди отримують інформацію про бізнес, інформацію про продукт, інформацію про рекламу та рекомендаційну інформацію з мобільних пристроїв. Одне із важливих застосувань мобільних послуг – це рекомендація фільмів користувачу.

Система рекомендації фільмів виявилася потужним інструментом для надання корисних пропозицій фільмів для користувачів. За допомогою таких систем надаються рекомендації, які допомагають користувачам справитись із надмірною кількістю інформації, а також допомагають швидко та зручно підібрати відповідні фільми.

На відміну від персональних комп'ютерів, мобільні послуги приділяють більше уваги питанню швидкості надання послуг, що, у свою чергу, вимагає швидкої обробки та обчислень з боку провайдерів послуг. Таким чином, рекомендації фільмів у мобільних сервісах повинні виконувати якісну роботу як з точки зору точності наданих рекомендацій, так і з точки зору швидкості обробки інформації.

Формування системи рекомендації фільмів – це всеохоплююче і складне завдання, яке передбачає аналіз різних вподобань користувачів, різноманітності фільмів та інше. Тому для вирішення проблеми надання якісних рекомендацій у сучасному світі уже запропоновано багато методів.

На практиці часто використовують системи рекомендацій на основі контенту, колаборативні системи рекомендацій та гібридні рекомендаційні системи. Кожен метод має свої переваги у вирішенні конкретних проблем. Беручи до уваги особливості використання онлайн-інформації та контенту, створеного користувачами, колаборативна фільтрація вважається найбільш

популярною та широко застосовуваною технологією в рекомендаційних системах.

Метод колаборативної фільтрації може рекомендувати елементи, вимірюючи подібність між користувачами або самими елементами. Подібність між вподобаннями користувачів може вимірюватися кореляційним розрахунком. Таким чином користувачі, які мають схожі вподобання фільмів, відсортовуються в одну і ту ж групу, а потім фільми рекомендуються за відгуками та рейтингами тих фільмів, які вони вже подивилися. Проте кореляцію та подібність досить складно розрахувати через обмеженість доступу до базових даних користувача, таких як оцінка користувачів про переглянуті фільми та історія їх веб-переглядів.

Насправді, рецензії(відгуки) користувачів про фільми зазвичай містять більше інформації, наприклад, інформацію про уподобання користувачів. Більше того того, ігнорування тональності відгуків користувачів також є серйозною проблемою у формуванні рекомендацій фільмів.

На даний момент люди все частіше готові розміщувати свої власні відгуки в Інтернеті. У рецензіях користувачі можуть висловити свої уподобання та почуття про фільми, тож тональність цих відгуків також впливають на вибір інших користувачів. Користувачі дивляться відгуки, аналізують свій особистий досвід, вибирають корисні, на їх погляд, відгуки, видаляють оманливі чи навіть шкідливі відгуки, і в кінцевому підсумку роблять власні судження та приймають рішення. Відповідно, тональність відгуку є дуже важливим аспектом при оцінці фільму.

В загальному, користувачі більше схильні вибирати фільми, яким надала перевагу більшість людей, і відмовляються від фільмів, які більшості людей не сподобались. Рішення на основі досвіду інших людей приймається

користувачами з метою спрощення і покращення власного досвіду користування системою.

Із збільшенням об’єму даних швидкість забезпечення користувачів високоякісними рекомендаціями серед великих об'ємів інформації стала серйозною проблемою. Також із поширенням використання мобільних послуг швидкість відповіді на запит стала важливим показником якості обслуговування користувачів.

Технічні методи аналізу тексту та аналізу тональності коментарів, що використовуються для аналізу відгуків користувачів, ускладнюють системи рекомендаційної діяльності у традиційному середовищі. Нове покоління рекомендаційних систем потребує вирішення питання того, як швидко робити високоякісні рекомендації у великій кількості даних і як зробити систему максимально масштабованою.

Технологія передачі великих даних (big data) – це один з потужних інструментів для вирішення подібних проблем. Деякі рекомендаційні системи, які ґрунтуються на використанні класичної кластерної обчислювальної технології Hadoop, можуть пом'якшити вплив складності розрахунків, яка викликана збільшенням кількості даних. Проте в умовах складного процесу або великої кількості ітерацій Hadoop не є належним інструментом через надмірне використання операцій вводу-виводу. Надзвичайно довгий час обробки є критично важливим для Hadoop за точки зору вимоги високої швидкості відповіді.

Кластерна технологія Spark, у свою чергу, дозволяє подолати вказані недоліки. На відміну від використання дискових носіїв у технології Hadoop, Spark дозволяє зберігати проміжні результати в пам'яті в процесі розрахунку. Разом із тим, у технології Spark також оптимізовано ітеративний процес обчислення. Отже, у рекомендаційних системах продуктивність обробки Spark краще, ніж Hadoop.

У даному дослідженні для формування рекомендацій релевантних користувачам фільмів запропоновано гібридний підхід (аналіз на основі змісту та колаборативний підхід), що використовує платформу Spark. Також застосовується аналіз тональності відгуків користувачів, який є більш надійним, ніж простий рейтинг, завдяки тому, що він містить більше емоційної інформації, яка виявляється більш корисною при оцінці предметів мистецтва, зокрема, фільмів. Крім того, висока ефективність системи Spark дозволяє покращити швидкість надання мобільних послуг користувачеві.

# 1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ РОЗРОБКИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

## 1.1. Загальні відомості про рекомендаційні системи.

Рекомендаційна система – це програма, яка передбачає уподобання користувачів і рекомендує відповідні продукти або послуги конкретному користувачеві на основі його інформації та інформації про продукти або послуги. Дослідження систем рекомендацій розпочалося дослідницькою групою GroupLens з Університету Міннесоти. Їхнім об’єктом дослідження була рекомендаційна система фільмів, яка називається MovieLens.

Ранні дослідження зосереджені, головним чином, на об'єктах рекомендаційної системи, тобто аналізувалися характеристики самого об'єкта формування рекомендації [1]. Проте цей метод рекомендації може бути застосований лише до аналізу на основі контенту, що змушує дослідників та практиків вносити багато інвестицій у розробку нових систем рекомендацій. Дослідники запропонували системи, що формують рекомендації на основі колаборативної фільтрації, правил асоціації [2], корисності, знань, на основі даних із соціальних мереж [3], багатокритеріального програмування [4], кластеризації [5] та інших теорій і методів.

Дослідники також вивчали можливості формування рекомендацій на мобільних платформах. Більшість досліджень щодо мобільних рекомендацій зосереджуються на сервісах визначення місцезнаходження користувача. Наприклад, у роботі [6] автори використовують дані трекінгу GPS для вирішення проблеми формування рекомендацій на мобільних пристроях. Запропоновано метод, що орієнтується на взаємодію з деякими користувачами, на основі взаємозв'язків з користувальницькою локалізацією та взаємодією та методом спільного опитування. На підставі цього дослідження автори створили алгоритм, який використовує класифікований

колективний тензор та матричну факторизацію, для формування рекомендацій користувачам [7]. Більше того, у роботі [8] автори формують рекомендації ресторанів, використовуючи байєсівські мережі, що ґрунтуються на визначенні точки перебування мобільного пристрою та деякої іншої інформації.

## 1.2. Методи аналізу даних та формування рекомендацій на їх основі

### 1.2.1. Рекомендації на основі вмісту (контенту)

Методи рекомендацій на основі контенту широко вивчаються протягом останніх кількох років. У роботі [1] запропоновано систему для рекомендації фільмів на основі вмісту, використовуючи оцінки фільмів як соціальну інформацію. Експериментальна перевірка довела практичні переваги запропонованого методу. Більше того, у роботі [9] автори використали байєсівські мережі для побудови моделей вподобань користувачами фільмів на основі їх контексту. Очевидно, що різноманітні методи можуть бути використані для побудови параметрів користувачів та фільмів, щоб рекомендувати відповідні фільми.

Окрім використання нових технологій для побудови параметрів, також вивчаються нові перспективи для створення чітких профілів користувачів та фільмів. Наприклад, у роботі [10] автори ввели семантичну мережу, щоб проаналізувати приховану у фільмах фольклономію з метою допомогти користувачам відшукати відповідні фільми. У роботі [11] автори використовували соціальну мережа для аналізу індивідуальних контекстних особливостей поведінки покупців.

Попри велику кількість досліджень, розробка ефективних профілів завжди є проблематичним місцем у рекомендаційних системах, які грунтуються на аналізі контенту, оскільки представлення об'єктів

рекомендації або користувачів у вигляді векторів параметрів є складною задачею, яка не завжди приносить задовільний результат. І дослідники, і практики зробили великі кроки у розробці нового методу рекомендації, щоб уникнути недоліку систем, які ґрунтуються на аналізі змісту.

### 1.2.2. Рекомендації на основі колаборативної фільтрації

Колаборативна фільтрація використовується для усунення недоліків алгоритму на основі контенту. Алгоритм колаборативної фільтрації був розділений на частини для детального аналізу у рекомендаційній системі фільмів у роботі [12]. В той же час, у роботі [13] автор виявив, що уподобання користувачів з часом можуть змінюватися, тому він розробив рекомендаційний алгоритм, який використовує часову динаміку для вирішення проблеми. Більш того, у роботі [14] автор реалізував імовірнісний латентний семантичний аналіз Гауса у методі колаборативної фільтрації для формування рекомендацій щодо фільмів. Дослідники доклали багато зусиль та застосували сучасні технології для покращення ефективності колаборативної фільтрації для формування системи рекомендації фільмів, що дозволяють досягти хороших результатів.

Колаборативна фільтрація є поширеним інструментом, що використовується в системах рекомбінації [15]. У роботі [16] автор розробив метод колаборативної фільтрації, який ґрунтується на основі оцінок. У роботі [17] автори запропонували власний метод колаборативної фільтрації, що вони назвали факторизацією матриці імовірностей, який може обробляти великі набори даних. У той же час у роботі [18] автори показали можливість використання обмежених машин Больцмана (restricted Boltzmann machines, RBM) для підвищення продуктивності колаборативної фільтрації. Результати експерименту показали, що обмежені больцманівські машин перевершили сингулярний розклад матриці на наборі даних компанії Netflix. Крім того, у

роботі [19] автор поєднав покращені моделі латентних факторів та моделі сусідства на цьому ж наборі даних. Використана латентна факторна модель є за своєю суттю SVD, тоді як модель сусідства оптимізована з використанням функцію втрат.

Більше того, дослідники також запровадили інші способи пошуку даних, щоб оптимізувати рекомендаційні системи. Наприклад, у роботі [20] автор запропонував факторизаційні машини (FM), які об'єднують метод опорних векторів (SVM) з моделями факторизації. У роботі [21] автори використали відрегульовані факторизаційні машини, що використовуються імовірнісні факторизації матриць (factorization machines, PMF).

Тим не менш, у методі колаборативної фільтрації було виявлено нові недоліки, які не були характерні для методів рекомендацій на основі контенту. Наприклад, масштабованість колаборативної фільтрації є досить поганою. В моменти, коли користувачі формують нові моделі поведінки, колаборативна фільтрація неспроможна адекватно реагувати на такі зміни. Тому, як дослідники, так і практики схильні до гібридизації методу колаборативної фільтрації та методу аналізу контенту для вирішення вищезазначеної проблеми [22, 23].

Наприклад, у роботі [24] автори презентували рекомендаційну систему, яка ґрунтується як на методі колаборативної фільтрації, так і на методі аналізу контенту. У тій частині гібридної системи, яка базована на контенті, важливість цієї функції виражена дуже яскраво. У роботі [25] автори запропонували алгоритм вибору різноманітних предметів для оптимізації результатів методу колаборативної фільтрації для підвищення ефективності гібридної системи рекомендацій.

У роботі [26] автори представили уніфіковані машини Больцмана для гібридизації колаборативного методу фільтрації та фільтрації на основі змісту, кодуючи їхню інформацію. Інтегрувавши підходи колаборативного

методу фільтрації та рекомендації на основі змісту, автори роботи [27] створили алгоритм, який базується на основі аналізу відгуків, що робить рекомендацію більш точною.

У роботі [28] автор використав рейтингову модель у поєднанні з тематичною моделлю на основі відгуків для підвищення точності передбачень. Як видно з наведених вище досліджень, гібридна система рекомендацій може не тільки покращити ефективність, але й покращити масштабованість системи рекомендацій фільмів. Тому гібридна рекомендаційна модель – це найбільш перспективний спосіб формування рекомендацій фільмів.

## 1.3. Аналіз тональності відгуків

Аналіз тональності – це процес аналізу, обробки, узагальнення та обґрунтування емоційного тексту [29]. Дослідження аналізу тональності розпочалися в 2002 році [30] та досить добре розвинулися у напрямку аналізу онлайн-обговорень та коментарів. В даний час точність аналізу емоційної полярності текстових коментарів поступово зростає, однак існуючими в емоційному аналізі проблеми є відсутність поглибленого аналізу та застосування аналізу тональності.

У роботі [30] автори використовують контрольований метод навчання, щоб класифікувати емоційну полярність тексту коментарів до фільму на позитивний та негативний, використовуючи підхід N-грам та метод максимальної ентропії (ME).

У роботі [31] автор використав метод машинного навчання без вчителя, щоб визначати тональність тексту. У цій роботі було вперше використано теги для отримання пар слів з відгуків, а потім використовував метод PMI-IR (pointwise mutual information and information retrieval) для

обчислення подібності між словами в тексті та словами в текстовому наборі для визначення емоційної тональності тексту. Метод отримав точність 65,83% у наборі даних про фільми.

Тональність відгуків про фільмів та інші товари або послуг може бути поділена на позитивні, негативні та нейтральні [32]. Виходячи з цього висновку, в деяких дослідженнях було проведено аналіз тональності у відгуках користувача та отримано полярність відгуків. Для користувачів рекомендовані фільми з найбільш позитивною інформацією [33]. У роботі [34] автори запропонували підхід для створення системи, що рекомендує соціальну рекламу, з урахуванням тональності. У роботі [35] автори проаналізували тональність відгуків у колаборативній фільтрації, застосувавши тематичну модель.

## 1.4. Аналіз великих об’ємів даних для формування рекомендацій

Проблема масштабованості системи рекомендацій також ускладнює роботу дослідників і практиків для надання користувачам зручних та ефективних послуг. Багато досліджень присвячені спробам вирішення цієї проблеми [36-38]. Паралельні обчислення є одним з найбільш поширених рішень.

У роботі [39] автори побудували паралельну платформу з використанням мови Matlab для реалізації системи рекомендації фільмів на основі колаборативної фільтрації. При паралельних обчисленнях ефективність алгоритмів рекомендацій вище, ніж у режимі використання однієї машини. Впровадження розподіленої обчислювальної системи підвищує ефективність рекомендаційної системи.

Розподілена обчислювальна система – це така обчислювальна система, яка поєднує декілька обчислювальних машин, що виконують одну

задачу. Вузли такої систему мають певну ступінь свободи – у них є власне незалежне апаратне і програмне забезпечення. Незважаючи на це, вузли можуть використовувати спільні ресурси і інформацію, тому, з метою вирішення задачі, що стосується декількох чи навіть усіх вузлів, необхідно координувати їх роботу.

Оскільки існує велика кількість різних типів обчислювальних задач і підходів їх розв'язку, тому в галузі розподілених систем існує багато моделей і параметрів. У певних типах систем вузли працюють синхронно, в той час як в інших системах робота виконується асинхронно. Існують відносно прості гомогенні системи, у яких усі вузли мають один і той самий тип, або ж гетерогенні системи, де повинні взаємодіяти вузли різних типів, потенційно з різними можливостями і завданнями.

Також бувають різні підходи до організації процесу комунікації між елементами розподіленої обчислювальної системи. Вузли (ноди) можуть комунікувати за допомогою обміну повідомленнями або ж за допомогою використання спільної пам'яті. Іноді комунікаційна інфраструктура спеціально створюється і налаштовується для розподіленого застосування, або ж доводиться використовувати існуючу інфраструктуру.

Вузли розподіленої системи системи зазвичай працюють разом над виконанням одного глобального завдання. Але можливий і випадок, коли вузли можна розглядати як автономні агенти, що виконують власні задачі і змагаються за загальні ресурси.

У певних випадках елементи обчислювальної системи можна вважати такими, що працюють коректно, іноді ж у їх роботі можуть відбуватися збої. На відміну одноелементних систем, розподілені системи можуть продовжувати виконувати завдання у випадку збоїв, оскільки інші елементи можуть перебирати на себе частину роботи.

Розглядають декілька видів збоїв у розподілених системах: вузли можуть стати недоступними, або ж їх поведінка може бути помилковою, іноді навіть навмисно шкідливою. Можлива такою ситуація, коли елементи системи підпорядковуються загальним правилам, але водночас змінюють для отримання максимально можливої частини загальних ресурсів, тобто діють за егоїстичним алгоритмом.

З наведеного вище очевидно, що існує багато моделей і платформ для розподілених обчислень. Популярними прикладами систем є Apache Hadoop та Apache Spark. Ці системи розроблені для розподіленої обробки великих об'ємів даних на комп'ютерних кластерах.

Наприклад, Hadoop може допомогти методу колаборативної фільтрації досягти лінійного прискорення [40, 41]. І для великих наборів даних можливе досягнення кращого відносного пришвидшення, ніж для малих [42]. Незважаючи на те, що Hadoop до певної міри забезпечує масштабованість алгоритмів рекомендацій, підтримка MapReduce для алгоритмів колаборативної фільтрації не є досконалою. Причина полягає в тому, що колаборативна фільтрація вимагає постійного читання та запису даних при обчисленні подібності елементів. Проте, Hadoop – це платформа, яка базується на постійному використанні жорсткого диску, і постійне зчитування та запис даних стають вузьким місцем у обчисленнях. Тому, платформа Spark, що надає можливість використовувати оперативну пам’ять замість дискової, стала найпоширенішим рішенням для рекомендаційних систем.

У роботі [43] автори використали метод змінних найменших квадратів (ALS) і метод *k*-середніх для того, щоб досягнення розрідженості даних та масштабованість роботи колаборативних алгоритмів. У роботі [44] автори реалізували багатокритеріальний алгоритм колаборативної фільтрації, використовуючи систему Spark. Результати експериментів показали, що

ефективність алгоритмів покращилася з кількістю вузлів у кластерах Spark. Тому для того, щоб отримати більш високу ефективність обчислень, було вирішено надати перевагу використанню Spark в рекомендаційних системах.

Нижче наведено порівняльну таблицю переваг та недоліків платформ Hadoop MapReduce та Spark.

Таблиця 1.1 – Порівняння платформ Hadoop MapReduce та Spark

| Hadoop MapReduce | Spark |
|---|---|
| Умовно швидка обробка даних | У 100 разів швидша обробка даних за рахунок використання оперативної пам'яті |
| Пакетна обробка даних | Обробка даних у режимі реального часу |
| Дані зберігаються на жорстких дисках | Дані зберігаються у оперативній пам'яті |
| Мова написання – Java | Мова написання – Scala, Python, Java |
| Умовно невисока вартість використання | Умовно висока вартість використання |

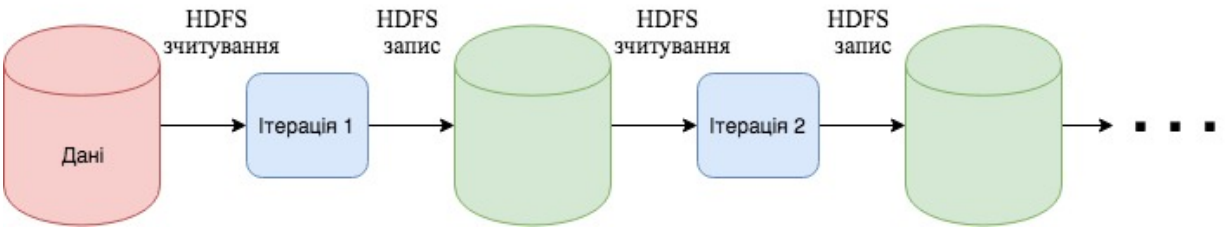


Рисунок 1.1 – Процес обробки даних на платформі Hadoop

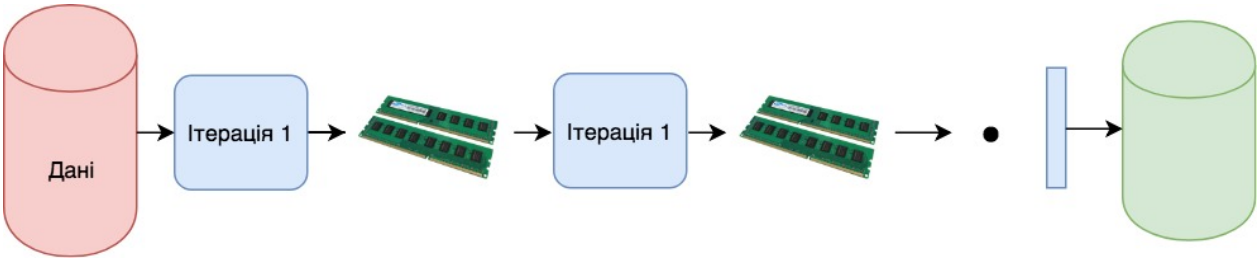


Рисунок Рисунок 1.2 – Процес обробки даних на платформі Spark

Як зазначено вище, існують різні моделі формування рекомендаційних систем для надання найбільш точних рекомендацій фільмів. Попередні практики та дослідження зосереджують увагу на покращенні ефективності рекомендацій за допомогою комбінації моделей рекомендацій. Проте, вони ігнорують те, що зі збільшенням кількості користувачів та рекомендованих об’єктів, обчислювальні витрати значно зростають. Тому в цій роботі запропоновано концепцію рекомендацій фільмів на базі платформи Spark для задоволення вимог мобільних користувачів для забезпечення швидкодії системи.

У запропонованому методі використовуються як методи фільтрації на основі контенту, так і метод колаборативної фільтрації. Виходячи з методу колаборативної фільтрації та методу фільтрації на основі контенту, попередні вихідні дані оптимізуються шляхом аналізу відгуку з точки зору як позитивної так і негативної інформації. Врешті-решт, проводяться експерименти з метою доведення ефективності методу, запропонованого у цьому дослідженні.

## 1.5. Висновки

Було проаналізовано існуючі дослідження формування систем рекомендацій. Виявлено, що більшість робіт ґрунтуються на класичних методах формування рекомендацій: метод колаборативної фільтрації, метод

фільтрації на основі змісту та гібридний метод фільтрації, який поєднує два вищезазначені методи фільтрації. При аналізі існуючих досліджень було виявлено, що гібридний метод фільтрації зазвичай є найбільш оптимальним рішенням при побудові рекомендаційних систем, адже саме в цьому методі компенсуються недоліки окремого використання системи колаборативної фільтрації чи фільтрації на основі змісту.

Формування максимально релевантного списку рекомендацій фільмів неможливе, якщо система ґрунтується лише на очевидних даних, таких як змісті, оцінки користувачів, базові дані про користувачів. Для покращення результатів рекомендаційної системи все більше дослідників звертаються до інтеграції з аналізом тональності відгуків користувачів. Найчастіше зустрічаються рішення, які визначають відгук як позитивний, негативний чи нейтральний. Така модель взаємодії надає змогу більш точно визначити релевантність того чи іншого продукту до списку рекомендацій користувача. При цьому, більшість досліджень ігнорують пролему «прихованого змісту», коли у відгуках застосовуються такі форми вираження думки як сарказм чи іронія.

При аналізі проблеми масштабованості системи рекомендацій було розглянуто дві найпоширеніші платформи – Hadoop і Spark. На основі вивчення існуючих досліджень було виявлено кілька значних недоліків у платформи Hadoop. Приміром, такими недоліками є неперервне використання жорсткого диску або постійне зчитування та запис даних.

Spark, у свою чергу, працює за принципом використання оперативної пам’яті, а не дискової, що дозволяє йому в рази швидше обробляти великі об’єми даних.

Так як у сучасному світі все більше і більше людей отримують інформацію з мобільних пристроїв, одним із важливих критеріїв роботи рекоменаціїйної системи є швидкість відповіді мобільного пристрою на запит

користувача. Тож для побудови рекомендаційних систем все більше і більше дослідників звертаються до платформи Spark.

# 2. РЕКОМЕНДАЦІЙНА СИСТЕМА НА ОСНОВІ ВИЗНАЧЕННЯ ТОНАЛЬНОСТІ ВІДГУКІВ КОРИСТУВАЧІВ

Як згадувалося раніше, у цьому дослідженні використовується колаборативна система фільтрації та система фільтрації на основі контенту. Колаборативний аналіз та аналіз на основі контенту підходять для компенсації недоліків один одного, що забезпечує точність та стабільність системи рекомендацій. З одного боку, колаборативна фільтрація може компенсувати відсутність персоналізації в методі на основі контенту. З іншого боку, метод на основі контенту може компенсувати такий недолік методу колаборації як масштабованість, яка у нього є відносно слабкою. Загалом спосіб гібридної рекомендації виконується на основі даних користувача та даних про фільм, щоб отримати попередній список рекомендацій.

Аналіз тональності впроваджується для оптимізації попереднього списку та отримання списку рекомендацій. Крім того, на основі гібридної рекомендаційної платформи це дослідження повністю бере до уваги ефективність системи рекомендацій.

У процесі рекомендації фільмів дослідження зосереджується на відгуках користувачів про фільми. В умовах колективної поведінки користувачі схильні вибирати товари або послуги, яким більшість людей віддає перевагу.

Отже, у порівнянні з фільмами, у яких багато негативних відгуків, фільмам з більш позитивними відгуками будуть надаватися пріоритети для рекомендації користувачам. Після оптимізації, генерується фінальний список рекомендацій, як показано на рисунку 2.1.

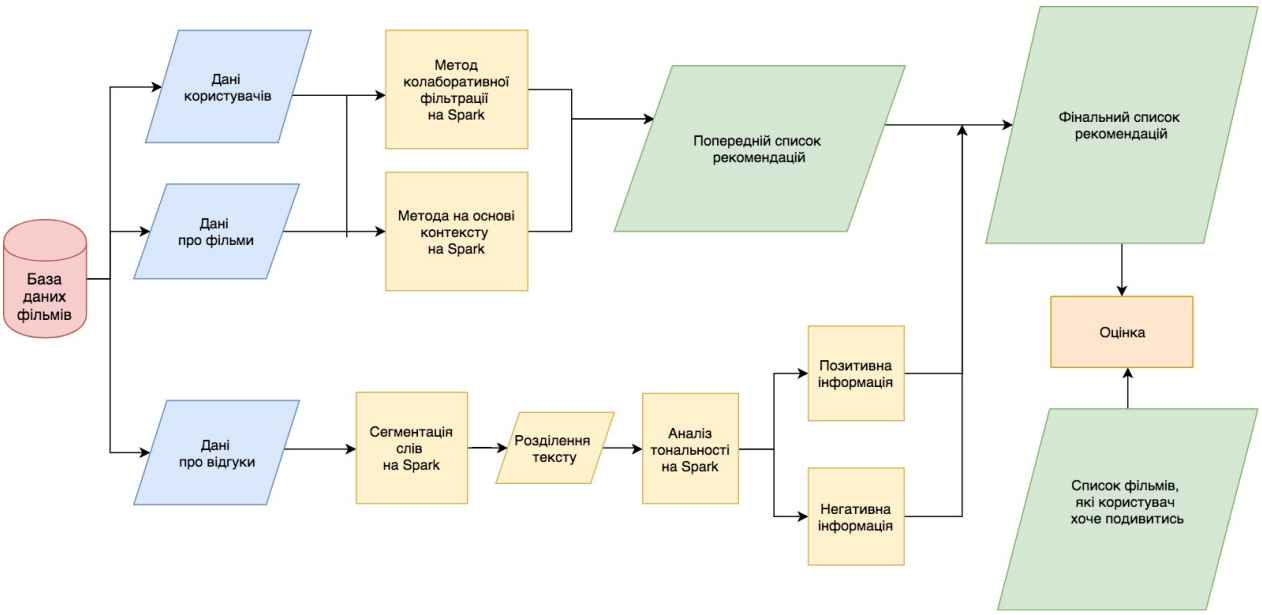


Рисунок 2.1 – Гібридна рекомендаційна модель з урахуванням тональності відгуків

## 2.1. Збір даних

У цьому дослідженні використовуються дані, отримані з сайту IMDB (https://www.imdb.com/), щоб перевірити правильність запропонованої моделі. Дані фільмів IMDB можна поділити на дані користувача та дані фільмів.

У якості джерела відгуків на фільми використовується соціальна мережа Twitter (https://twitter.com). Згідно даних ресурсу Alexa [https://www.alexa.com], Twitter займає 12-ту сходинку серед найбільш відвідуваних інтернет-ресурсів у світі з щомісячною активною аудиторією в 326 мільйонів (дані актуальні для 12 жовтня 2018). Така велика кількість активних користувачів забезпечує велику кількість повідомлень у цій соціальній мережі. Також Twitter надає програмний інтерфейс за протоколом HTTP для пошуку і завантаження повідомлень.

Сукупність наведених вище факторів дозволяє використовувати цю соціальну мережу для пошуку повідомлень на нові фільми, відгуків на які не було на момент тренування рекомендаційної системи. Це є великою перевагою над готовими наборами даних з відгуками, що за своєю природою можуть містити лише історичні дані.

У якості мови програмування для розробки гібридної рекомендаційної системи було обрано Python – інтерпретовану мову програмування високого рівня. За понад 25 років розвитку мова отримала потужні можливості, що забезпечило її популярність як засобу розробки широкого діапазону застосувань. Близько 14% розробників використовують її для операційних систем UNIX, Linux, MacOS, Windows та інших.

Основними особливостями мови, що забезпечили її широке розповсюдження, є:

- інтерактивність;
- модульність;
- динамічна типізація;
- портованість;
- підтримка декількох парадигм програмування, зокрема: об'єктно-орієнтованої, функціональної, процедурної;
- можливість створення C аоб C++ розширень.

Python має різноманітні застосування для розробки програмного забезпечення як веб, системні програми, скрипти, розподілені застосування, прототипування та інші. Основними перевагами мови є:

- наявність великої кількості допоміжних бібліотек.

Мова має об'ємну стандартну бібліотеку, що включає засоби для операцій з рядками і регулярними виразами, роботи в мережі Інтернет, допоміжні веб засоби, інтерфейси і протоколи для роботи з функціями ОС. Для багатьох поширених завдань уже наявні готові рішення у вигляді

стандартних модулів, що зменшує об'єм роботи, необхідної для виконання прикладних задач.

- інтеграційні можливості.

Мова дозволяє викликати компоненти, створені мовами C і C++, а також Java через інтерпретатор Jython або бібліотеку Py4J. Python також може виконувати обробку XML і інших мов розмітки. Виконання програми може відбуватися на усіх сучасних операційних системах через однаковий байт-код, що генерується інтерпретатором.

- забезпечення високою продуктивності розробників.

Наявність готових бібліотек і підтримка високорівневих парадигм дозволяє зменшити кількість рядків, необхідних для створення застосування, у 2 – 10 разів у порівнянні з такими мовами як Java, VB, Perl, C, C++ and C#.

- розвинена спільнота.

Екосистема Python має велику кількість уже наявних відкритих сторонніх, а також вбудовані механізми їх інтеграції.

Водночас, мова не набула широкого поширення у певних галузях розробки програмного забезпечення, наприклад у галузі розробки для великих підприємств та фінансових установ. Серед недоліків мови варто згадати:

- відносно невелика швидкість виконання.

Виконання програм відбувається за допомогою інтерпретатора, а не компілятора. Це спричиняє уповільнення обчислень, оскільки тип об'єктів не є відомим наперед, а отже операції не можуть бути оптимізовані компілятором.

- помилки під час виконання.

Іншим наслідком відсутності компілятора і динамічної типізації неможливість перевірки сумісності типів перед виконанням. Певні типи помилок виникають лише під час виконання програми і не можуть бути

перевірені перед запуском. Це може спричинити збільшення часу, необхідного на тестування програми. Частково цей недолік можна уникнути, використовуючи статичні аналізатори коду, доступні для сучасних версій інтерпретатора Python 3.

- слабка підтримка мобільних платформ.

Python набув розповсюдження на багатьох персональних та серверних платформ, але займає малу частину мобільного сегменту.

Можна зробити висновок, що Python є потужною зрілою технологією і є зручним інструментом для написання, підтримки і відлагодження програмного забезпечення. Python може бути використаний для автоматизації отримання відгуків користувачів соціальної мережі Twitter, оскільки основна затримка у цьому процесі виникає через операції мережевого обміну і не залежить від мови програмування, що використовується. Також Python може бути використаний для кластерних обчислень на платформі Spark, оскільки власне обчислення відбуваються у середовищі JVM (Java virtual machine) через модуль Py4J.

Для доступу до бази даних повідомлень Twitter у мові Python використовується бібліотека tweepy [http://www.tweepy.org/]. Tweety використовує відкритий механізм автентифікації і авторизації в мережі Інтернет Oauth [https://oauth.net/]. Приклад створення клієнта за допомогою tweepy:

```
import tweepy as tw

handler = tw.OAuthHandler(key, key_secret)
handler.set_access_token(token, token_secret)

twitter_api = tw.API(handler)
```

де `consumer_key, consumer_secret` – ключ і кодове слово, які можна отримати при реєстрації нового додатку в Twitter.

Створений вище клієнт може бути використаний для пошуку повідомлень в мережі Twitter. Для цього необхідно вказати пошуковий запит `query`, а також максимальну кількість повідомлень `max_messages`, які необхідно завантажити:

```
messages = [response._json for response in
tw.Cursor(twitter_api.search,q=query).items(max_messages)]
```

Дані користувача та дані про фільм використовуються як вхідні дані методі колаборативної фільтрації, а дані відгуків використовуються як вхідний матеріал методу на основі контенту. Вхідні дані потребують попередньої обробки, яка включає в себе очистку даних, інтеграцію даних та перетворення даних.

## 2.2. Побудова модуля гібридної рекомендації.

У запропонованому методі гібридний метод рекомендації є основним для створення попереднього списку рекомендацій. Для обробки гібридного методу на Spark потрібні наступні кроки.

### 2.2.1. Збір уподобань користувача та їх представлення.

Метод колаборативної фільтрації використовується для виявлення принципів поведінки користувачів та формування їх уподобань, тому підхід збору вподобань користувача є основою цього методу. У користувачів є безліч способів надати системі дані про свої вподобання, такі як оцінювання (рейтинги) та кліки. У запропонованому методі враховуються користувацькі рейтинги фільмів. Дані попередньо обробляються до того, як імпортуються

до моделі колаборативної фільтрації. Основним завданням тут є нормалізація та зменшення шуму. По-перше, шум повинен бути відфільтрований, оскільки існування шуму призведе до зниження продуктивності та ефективності системи рекомендацій. По-друге, вхідні дані потребують нормалізації. При нормалізованих даних метод може бути більш точний.

За допомогою наведених вище кроків отримується двомірна таблиця, в якій один параметр – це список користувачів, а інший – список фільмів, тоді як значення – це оцінки фільмів користувачами. Дані про вподобання перетворюються на стійкі розподілені набори даних (resilient distributed datasets, RDD), які можуть бути оброблені Spark.

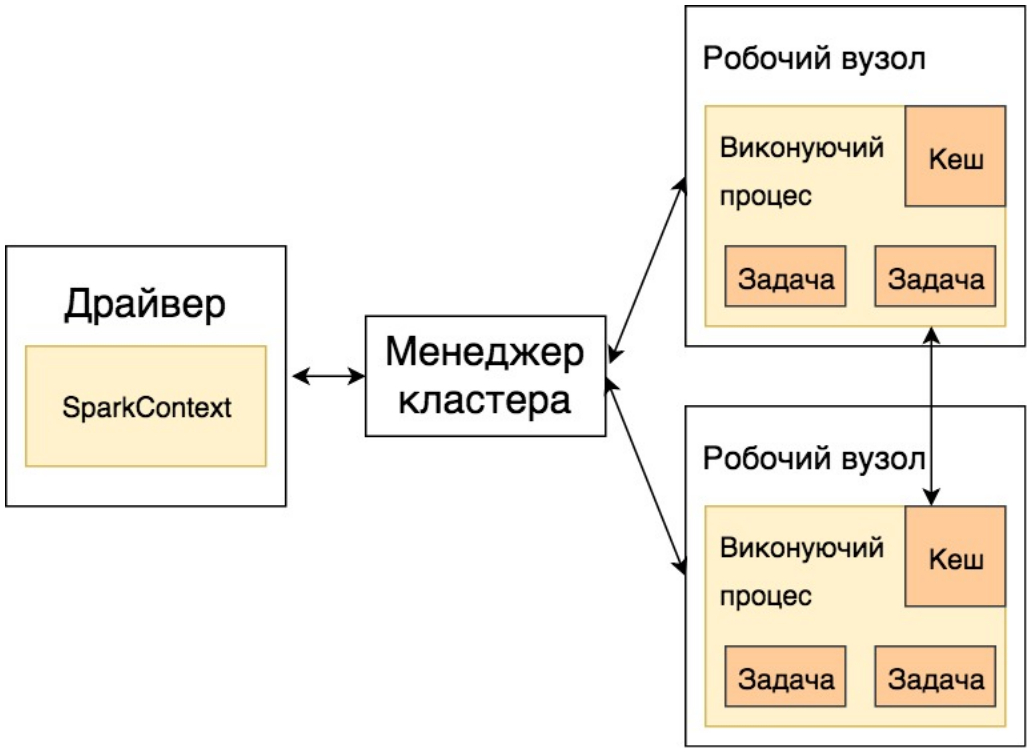


Рисунок 2.2 – Розподілене виконання Spark

Spark Context: Він підтримує зв'язок із менеджером кластерів Spark. Всі програми Spark працюють як незалежний набір процесів та координуються у програмі саме завдяки SparkContext.

Драйвер та виконуючий процес: Драйвер відповідає за процес запуску `main()` функції програми та створення SparkContext. З іншого боку, виконуючий процес – це будь-який вузол, який може запускати програму в кластері. Якщо процес запускається для програми, тоді ця програма починає виконуватись на робочому вузлі.

Менеджер кластера: Менеджер кластера розподіляє ресурси для кожної програми в програмі драйвера. Існує три типи менеджерів кластерів, які підтримуються Apache Spark – Standalone, Mesos і YARN. Apache Spark не підтримує ідею основного менеджера кластера, тому в процесі може бути встановлений будь-який менеджер кластера, кожен з яких має свої унікальні переваги залежно від мети.

Всі кластери різні за планом, безпекою та моніторингом. У момент, коли SparkContext підключається до менеджера кластера, він отримує виконавців на кластерному вузлі. Ці виконавці є робочими вузлами (нодами) на кластері, які працюють незалежно від кожного завдання та взаємодіють один з одним.

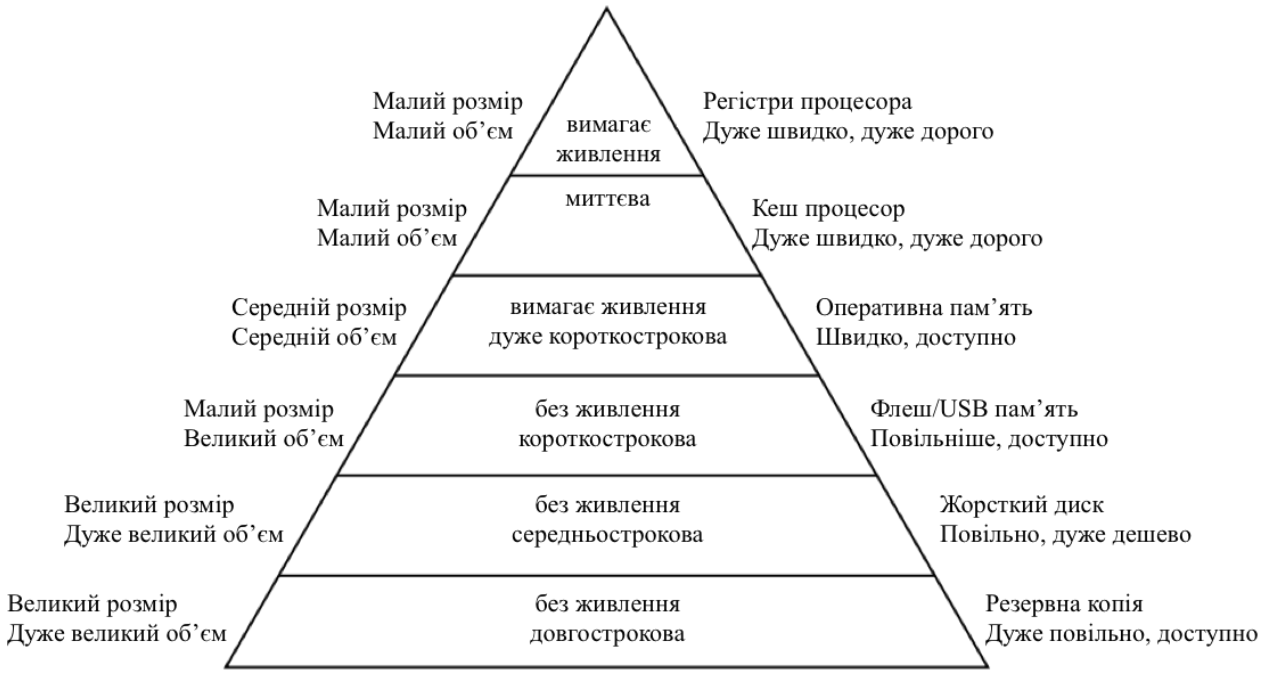

Рисунок 2.3 – Ієрархія комп’ютерної памяті

Обчислення у пам'яті: Найважливішою перевагою Apache Spark є той факт, що він зберігає та завантажує дані в оперативну пам'ять та з неї, а не з диска (жорсткий диск). Якщо говорити про ієрархію пам'яті, то оперативна пам'ять має набагато більшу швидкість обробки, ніж жорсткий диск (ілюструється на рисунку вище). Оскільки вартість пам'яті значно знизилися протягом останніх кількох років, обчислення в пам'яті отримали великиу популярність.

Spark використовує обчислення в пам'яті і, таким чином, процеси обробляються у 100 разів швидше, ніж при вифористанні платформи Hadoop.

RDD (Resilient Distributed Database) – це сукупність елементів, які можна розділити на декілька вузлів у кластері для паралельної обробки. Це також відмовостійкий набір елементів, що означає, що він може автоматично відновитись після збоїв. RDD незмінна, тобто створювати RDD можливо лише один раз, після створення її неможливо змінити. Проте, моживо запустити будь-яку кількість операцій з використанням оригінальної RDD, а після цього, зробивши висновки, створити іншу RDD, застосувавши деякі модифкації та уточнення.

З використанням RDD можна застосувати два типи операцій:

1. Трансформація: трансформація означає операцію, яка застосовується на RDD для створення нової RDD.
2. Дія: дія означає операцію, яка також застосовується на RDD, що виконує обчислення та відправляє результат виконаних обчислень назад на драйвер.

Приклад: Карта (Трансформація) виконує операцію на кожному елементі RDD та повертає у відповідь нову RDD. Але, у випадку Зменшення (Action), вихідні дані карти зменшуються/агрегуються завдяки застосуванню

деяких функцій (Reduce by key). У документації Apache Spark визначено багато перетворень та дій.

Спираючись на дані поведінки користувача та на дані його уподобань уможливлюється формування певної системи, яка допомагає формувати подальші рекомендації.

Через актуальність вимог щодо швидкості надання мобільних послуг важливим фактором є покращення ефективності розрахунків. У процесі розрахування вподобань користувача дані зберігаються в пам'яті Spark. У випадку, якщо етапи розрахунку рекомендацій на основі контенту обробляються після запису даних у пам’ять, то виконуються непотрібні операції вводу/виведення даних. У зв’язку з цим у роботі було прийнято рішення записувати дані в пам'ять і обробляти вподобання користувача в рамках методу колаборативної фільтрації та представляти елементи в рамках методу на основі змісту одночасно. У запропонованому методі фільми представлені за жанрами, режисерами та акторами.

### 2.2.2. Розподілений процес

Для обробки даних у розподіленій формі платформа Spark обчислює загальну кількість елементів, яким кожен користувач віддає перевагу, а також загальну кількість елементів, яким одночасно віддали перевагу будь-які два користувачі. Ці два типи статистики розподіляються на обчислювальних вузлах (нодах) платформи Spark, і результати зберігаються у формі RDD, відповідно.

### 2.2.3. Пошук схожих користувачів

Після отримання даних про вподобання користувача за допомогою аналізу його поведінки, схожі користувачі і певні одиниці можуть бути вирахувані на основі переваг користувачів.

Для пошуку користувачів із схожими уподобаннями, розраховується схожість між користувачами. У цій роботі використано Евклідову метрику для вимірювання подібності. Тому подібність між користувачами $u_x$, $u_y$ може бути розрахована за допомогою наступної формули:

$$sim(u_x u_y) = \sqrt{\frac{\sum_{i=1}^{n} (x_i - y_i)}{n}}$$

де $x_i$, $y_i$ репрезентують рейтинги $u_x$, $u_y$ у фільмі $i$.

2.2.4. Обчислення та формування рекомендацій

У попередніх кроках всі користувачі можуть бути оцінені згідно із значенням sim($u_x$, $u_y$). Для того, щоб надавати рекомендації фільмів користувачу $u_x$, вибирається топ $K$ найбільш схожих користувачів. Потім, відповідно до їх подібності та уподобань щодо фільмів, розраховується список рекомендованих фільмів для користувача $u_x$. Крім того, враховуються також і подібності між вподобаннями користувача $u_x$ та векторами репрезентації одиниць (фільмів).

Фільми, які не підходять для користувача $u_x$, видаляються зі списку. Потім цей список вважається попереднім списком рекомендацій, який буде використовуватися в якості основи пропонованого методу. Підраховуються оцінки, отримані з двох методів рекомендації:

$$Score_{CF,m} = \sum_i sim(u_t, u_i) R_{i,m}$$

$$Score_{CB,m} = R_{t,m} \sum_i sim(m_t, m_i)$$

де $\boxed{Score_{CF,m}}$ являє собою оцінку фільму в спільній програмі. $\boxed{sim(u_t, u_i)}$ позначає схожість між користувачем $u_t$ та користувачем-кандидатом $u_i$. $\boxed{R_{i,m}}$ – рейтинг фільму $m$ від користувача-кандидата. $\boxed{Score_{CB,m}}$ є оцінкою фільму $m$ у методі рекомендацій на основі контенту. $\boxed{sim(m_t, m_i)}$ позначає подібність між фільмом $m$ та фільмами, які користувач вже переглядав. $\boxed{R_{t,m}}$ – рейтинг від користувача $u_t$ на фільм $m$.

## 2.3. Модуль рекомендацій на основі тональності.

Перш за все, алгоритм використовує текстову інформацію, яка не може бути використана безпосередньо. Відповідно, застосовується інтелектуальнний аналіз текстів (майнинг) для того, щоб вилучити інформацію, яка прихована у текстових даних.

Вважається, що до 80 відсотків всієї інформації в організаціях зберігається в неструктурованому текстовому форматі. Ця інформація включає переліки потреб конкретних клієнтів, досьє з продажу, технічні характеристики продуктів, звіти про технічне обслуговування та відгуки зацікавлених сторін. За допомогою традиційних методів аналізу таких даних важко отримати бізнес-аналітику із таких неоднорідних даних. Тож замість цього виконується пошук на основі текстових даних або інтелектуальнний аналіз тексту.

Велика кількість корисних даних може бути прихована у документації компанії. Тож інтелектуальнний аналіз даних та пошук на їх основі є найбільш відповідним рішенням для покращення роботи пошукових систем.

Інтелектуальний аналіз тексту – це набір процесів, необхідних для перетворення неструктурованих текстових документів або ресурсів у чітку, зрозуміу та структуровану інформацію. Після цього структурована

інформація може бути використана для автоматичного виявлення прихованих моделей залежностей та прогнозування майбутніх результатів, за допомогою використання комбінації статистичних, лінгвістичних методів та методів розпізнавання образів.

Інтелектуальний аналіз тексту – це міждисциплінарне поле, яке спирається на пошук інформації, видобування даних, машинне навчання, статистику та обчислювальну лінгвістику.

Зазначені вище методи використовуються для виявлення та презентації знань – фактів, моделей, ділових правил та відносин – які у іншому випадку блокуються в звичайній текстовій формі, яка є непроникною для автоматизованої обробки.

Типовий процес видобування текстів включає в себе наступні етапи:

- Визначення та попередній процесинг тексту, який потрібно буде аналізувати. Цей крок передбачає очищення тексту, видалення непотрібної інформації з тексту, розділення тексту на окремі токени (тобто, менші компоненти) та визначення частини мови на основі граматики використовуваної мови.
- Виділення релевантної інформацію та перетворення її у структуровані дані. Інформація отримується шляхом пошуку за токенованим текстом та подальшого зберігання результатів у більш структурованій, організованій формі, яка піддається подальшому аналізу.
- Формування вибірки важливих особливостей для створення концепцій та моделей категорій. Кількість понять, що містяться у неструктурованих даних, зазвичай дуже велика. Основним моментом цього етапу є визначення найбільш відповідних функції та використання їх для створення симслових моделей на основі категорій даних та відносин.

- Аналіз структурованих даних з метою виявлення взаємозв'язку між поняттями. На цьому етапі процес видобування тексту зливається з традиційним процесом видобування даних. Класичні підходи інтелектуального аналізу тексту класифікації, такі як кластеризація, прогнозування та класифікація, можуть використовуватися на структурованих даних, що були отримані внаслідок виконання попередніх кроків.

Поширені програми, які були отримані внаслідок цих аналізів, включають в себе визнання названих об'єктів, автоматичне підведення підсумків, класифікацію на основі відповідних функцій та видобуток для почуттів і думок споживачів, виражених у тексті.

З точки зору обробки тексту, який включено у дослідження, немає взаємозв'язку між різними відгуками, тому дані можуть бути розподілені безпосередньо без спеціальної обробки.

### 2.3.1. Визначення мови відгуку.

Частина відгуків, отриманих автоматично за допомогою Twitter API, може бути написано мовою, що відрізняється від мови, для якої тренується модель визначення тональності тексту. Тому такі відгуки необхідно відфільтровувати перед подальшою обробкою.

Для визначення мови відгуку було використано бібліотеку *langid*. Дана бібліотека містить готову модель для визначення англійської мови, а також набір утиліт для тренування моделей для будь-яких мов на власному наборі даних.

Приклад використання бібліотеки наведено нижче:

```
from langid.langid import LanguageIdentifier as
Indentifier, model
```

```
classifier = Indentifier.from_modelstring(model)

def has_language(text, language="en"):
    return classifier.classify(text)[0] == language
```

2.3.2. Аналіз тональності.

Після сегментації слів відбувається аналіз результатів сегментації на предмет визначення тональності відгуків користувачів.

Формально можна визначити проблему загальної задачі класифікації текстів можна наступним чином:

- Вхідні дані:
  - Документ *d*
  - Фіксований набір (сет) класів C = $\{c_1,c_2,..,c_n\}$
- Дані виходу:
  - Передбачений клас c ∈ C

У даному випадку термін *документ* суб'єктивний. Під поняттям «документ» мається на увазі твіт, фраза, частина статтей новин, повні статті новин, повна стаття, посібник з виробництва, публіцистичний твір тощо. Причиною цієї термінології є *слово*, яке є атомним суб'єктом і досить малим у такому контексті. Отже, для позначення великих послідовностей слів цей термін *документ* використовується в цілому. Твіт означає коротший документ, тоді як стаття означає більший документ.

Отже, навчальний набір n-кількості відмічених документів виглядає так: $(d_1,c_1), (d_2,c_2),...,(d_n,c_n),$ а кінцевий вихід – це навчений класифікатор.

Один із найважливіших моментів, які потрібно враховувати при роботі з аналізом тональності текстів, є те, що не всі слова у фразі передають почуття цієї фрази. Слова типу «я», «є» тощо не сприяють передачі будь-яких почуттів і, отже, вони не стосуються контексту аналізу тональності текстів.

В розрізі аналізу тональності тексту також можна проблему вибору основних функцій. Процес вибору функції зосереджується навколо спроб виявлення найрелевантніших функцій, які найбільше стосуються назви класу. Та ж сама ідея застосовується і при аналізі тональності текстів. Тому в цьому процесі беруть участь лише декілька слів у фразі, і саме їх виявлення та вилучення з фраз є досить складним завданням.

Після вилучення ключових слів, текстовий відгук виражається як векторна просторова модель (VSM).

VSM передбачає, що слова, з яких складається текст, є незалежні одне від одного, тож текст може бути представлений цими словами, які є основою для подання математичної моделі.

Вираження тексту як VSM робить презентацію тексту та його обробку зручнішими. Текстова категорія стосується лише певних слів, що містяться в тексті, та їх частоти в тексті. Так, відгук $D$ може бути виражено як вектор

$$D = \{(t_1 w_1), (t_2, w_2) \dots (t_n, w_n)\}$$

$t_i$ виражає $i$-те слово у огляді. $w_i$ представляє вагу $t_i$. У даному дослідженні використовується TF-iDF термінів як функціональні ваги.

Після того, як отримано векторне просторове уявлення про відгуки фільму, з'являється можливість виконання аналізу тональності відгуків на основі лексики.

Відгуки(рецензії) класифікуються на позитивні та негативні частини відповідно до лексикону. Лексикон побудований відповідно до жанру фільмів. Слова, такі як «хороший» та «чудовий» у відгуках свідчать про те, що користувач мав позитивне враження про фільм. Якщо більшість користувачів мають позитивну оцінку у фільмі, фільм слід розглядати як

апріорний – той, що буде рекомендований користувачам, котрі ще його не переглядали.

Після аналізу та обробки емоційно забарвлених слів у рецензії фільму та сентиментального лексикону у відповідних категоріях відгуків про фільми, розраховується значення тональності відгуку – $H$, а $H_m$ репрезентує відсоток певної тональності відгуку $m$:

$$H_m = \sum_{n=1}^{l} W_l \omega_l$$

де $W_l$ представляє «вагу» (важливість) слова в лексиконі відповідної категорії фільмів, а $\omega_l$ – це вага (важливість) слів у представленні векторної просторовості.

Тренування моделі для визначення тональності тексту відгуків до фільмів може бути виконано за допомогою наступної послідовності дій у платформі Spark:

```
tokenizer = Tokenizer(inputCol='text,
outputCol='splitted_words')
vectorizer = CountVectorizer(vocabSize=2**16,
inputCol='splitted_words' , outputCol='vectorized')
# Using min document frequency to delete sparse words
idf = IDF(inputCol='vectorized', outputCol='idf',
minDocFreq=10)
label = StringIndexer(inputCol='target', outputCol='label')
regression = LogisticRegression(maxIter=50)
model = Pipeline(stages=[tokenizer, vectorizer, idf, label,
regression])

fitted = model.fit(train_set)
```

```
fitted.save("sentiment.model")
```

2.4. Рейтинг та рекомендації

Готовий попередній список рекомендацій, який базується на гібридному методі рекомендації, містить фільми, сортовані за їхніми оцінками. Оцінки отримані від розрахунків методу колаборативної фільтрації та методу рекомендацій на основі контенту, як показано у наступній формулі:

$$Score_{hybrid,m} = Score_{CF,m} + Score_{CB,m}$$

де $Score_{hybrid,m}$ є оцінкою оцінку фільму $m$ в гібридній системі рекомендацій.

Аналіз тональності тексту оптимізує попередній список рекомендацій. Оцінка тональності додається до оцінки фільму. Тож оцінка кожного фільму обчислюється наступним чином:

$$Score_{final,m} = W_{hybrid}Score_{hybrid,m} + W_{SA}Score_{SA,m}$$

де $W_{hybrid}$ та $W_{SA}$ представляють вагу двох методів рекомендацій, а $Score_{SA,m}$ є оцінкою фільму $m$, отриманою на основі аналізу тональності відгуків користувачів. $Score_{SA,m}$ – це сума всіх $H_m$ відгуків про фільм.

Остаточний список рекомендацій формується відповідно до нової оцінки. Отриманий список – це група фільмів, які розташовані без будь-якого порядку. Щоб адаптуватись до цієї ситуації, список рекомендацій надається без будь-якого порядку. Тому, щоб надати достатню кількість відповідних фільмів, більше фільмів вибирають методом гібридної рекомендації, а потім деякі з них відкидаються за допомогою фінальних оцінок.

Рекомендаційна система відображає оптимізований список рекомендацій користувачам. Користувачі сайту IMDB мають “список побажань”, у якому перераховані фільми, які користувачі хочуть подивитись, але ще не бачили. Тому дане дослідження використовує “список побажань” для оцінки запропонованої моделі.

## 2.5. Висновки

# 3. ЕМПІРИЧНИЙ АНАЛІЗ РОБОТИ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ

## 3.1. Опис даних.

Дані, використані у цьому дослідженні, є реальними даними, отриманими із веб-сайту IMDB, який надає користувачам інформацію про фільми. Відгуки для кожного фільму можна отримати з соціальної мережі Twitter, як це описано у попередньому розділі.

## 3.2. Критерії формування оцінки.

Для оцінки ефективності запропонованої моделі аналізу результатів використовуються чотири критерії. Ці критерії виходять з матриці помилок, як показано в таблиці 3.1.

Таблиця 3.1 – Матриця помилок

| Список побажань | Рекомендаційний список | |
|---|---|---|
| | У списку | Не у списку |
| У списку | TP | FN |
| Не у списку | FP | TN |

Точність і повнота (precision and recall) у деякій мірі суперечать одне одному, тому у дослідженні було використано $F$-метрику. $F$-метрика – це середньозважена гармонійний точність та повнота, що дозволяє краще оцінити продуктивність моделі в більш повній перспективі.

$$TP\ rate = \frac{TP}{TP + FN}$$

$$precision = \frac{TP}{TP + FP}$$

$$F1 = \frac{2\ x\ precision\ x\ recall}{precision + recall}$$

$$FP\ rate = \frac{FP}{FP + TN}$$

3.3. Експериментальні результати.

Результати аналізу тональності відгуків про фільми тісно пов'язані з оцінкою попереднього списку рекомендацій. Аналіз тональності може оптимізувати список фільмів кандидатів. Таким чином, комбінація колаборативного методу фільтрації та методу фільтрації на основі контенту удосконалює роботу моделі, поданої у дослідженні.

Для порівняння було оцінено різні методи рекомендації. Експериментальні результати показані у таблиці 3.2 та на рисунку 3.1.

Таблиця 3.2 – Продуктивність рекомендаційних моделей

| | TP | FP | Точність | F1 |
|---|---|---|---|---|
| CF + CB | 0,645 | 0,355 | 0,531 | 0,582 |
| CF + CB + SA | 0,761 | 0,239 | 0,782 | 0,771 |

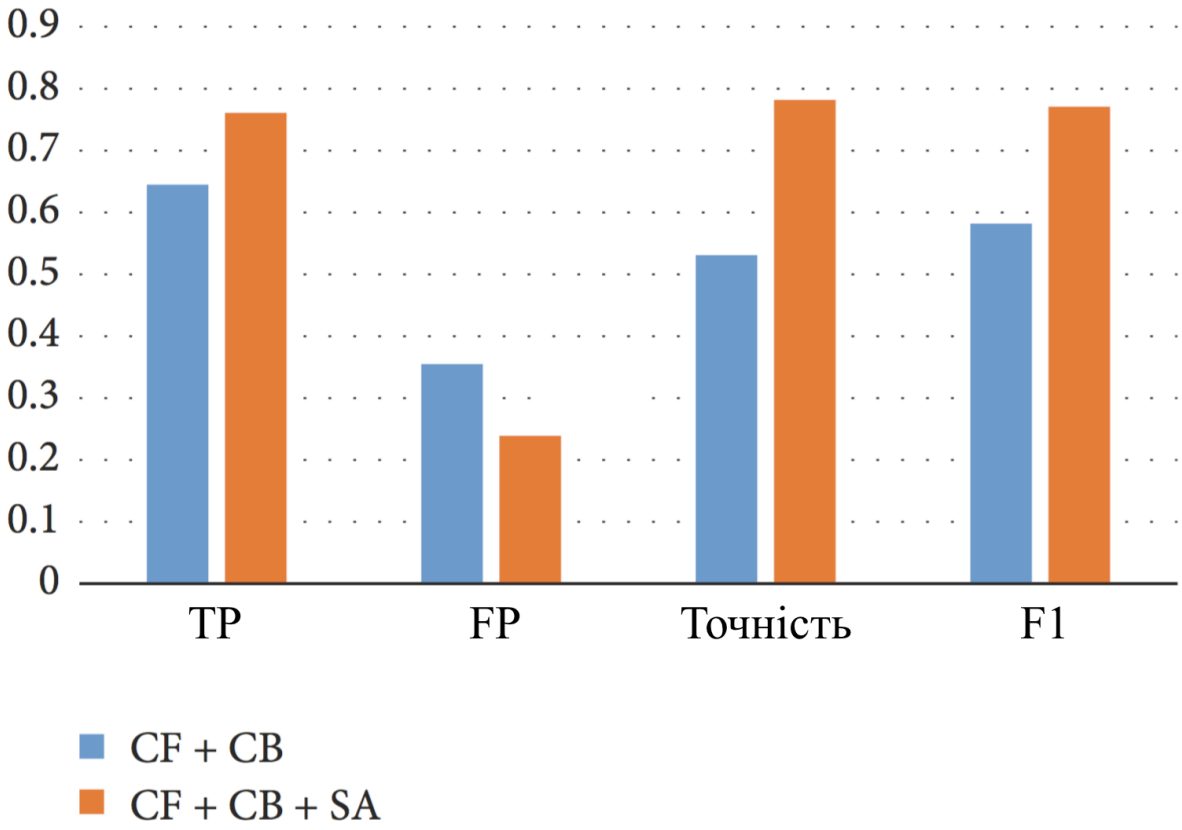


Рис. 3.1 – Продуктивність рекомендаційних моделей

Модель, запропонована у цьому дослідженні, працює краще, ніж базові рекомендації з точки зору частоти TP (TP-rate), що означає, що запропонована модель потужніша у здатності ідентифікувати релевантні фільми користувачу. CF означає метод колаборативної фільтрації. CB репрезентує метод фільтрації на основі контенту, а SA – скорочення для системи аналізу тональності.

Також було порівняно час запуску на різній кількості вузлів та з різною кількістю даних. Експериментальні результати показані в табл. 3.3 та рисунку 3.2

Таблиця 3.3 – Час роботи гібридної рекомендаційної системи на Spark

| Кількість вузлів | Час роботи на повному наборі даних, сек | Час роботи на половині набору даних, сек |
|---|---|---|
| (1) | 463 | 246 |
| (2) | 276 | 151 |
| (3) | 197 | 114 |
| (4) | 153 | 92 |
| (5) | 101 | 63 |
| (6) | 86 | 56 |
| (7) | 75 | 48 |
| (8) | 65 | 44 |
| (9) | 58 | 41 |

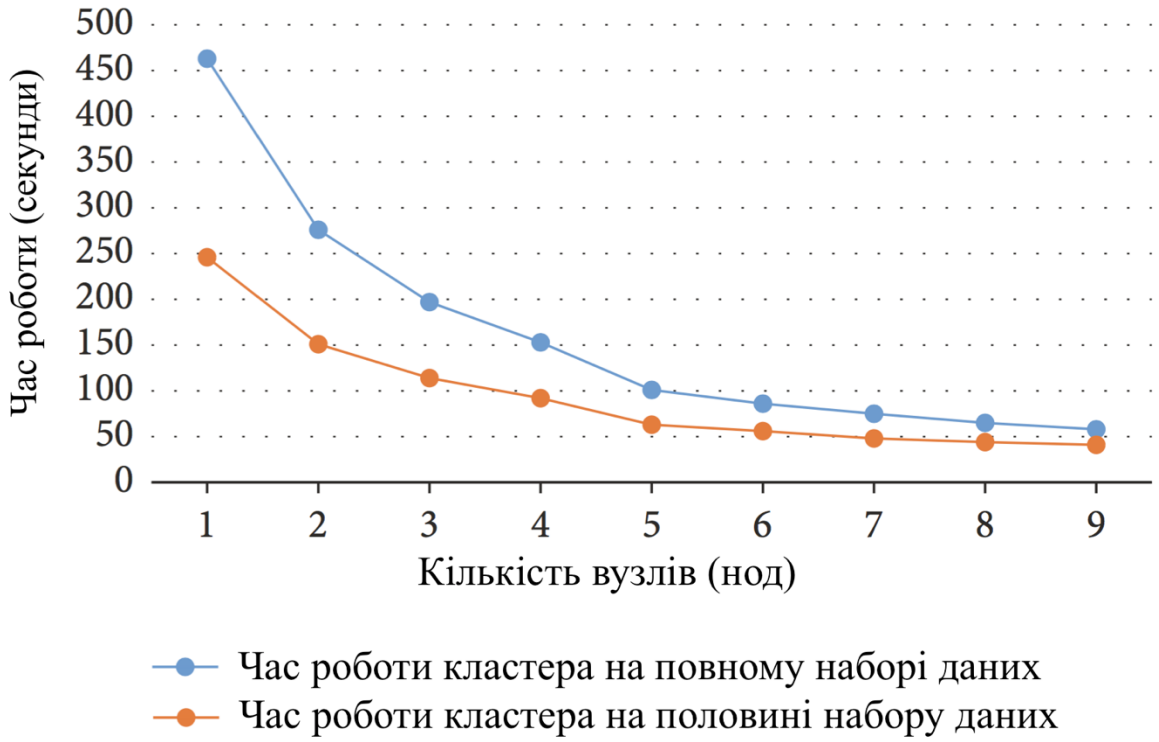

Рис. 3.2 – Час роботи гібридної рекомендаційної системи на Spark

По-перше, оскільки кількість вузлів у обчислювальному кластері збільшується, обчислювальна ефективність Spark зростає, і відповідний експериментальний результат показує, що час роботи зменшується. По-друге, коли запропонована модель застосовується у більших даних, прискорення обчислювальної ефективності краще. Результати показують, що запропонований нами спосіб добре працює як з точки зору точності, так і з точки зору ефективності. З одного боку, це може допомогти постачальника різноманітних послуг уникнути перенаправлення клієнтів через затримку інформації та рекомендації, надані користувачам мобільних послуг. З іншого боку, він може надавати допомогу для покращення швидкодії та досвіду користування системою.

## 3.4. Висновки

Було проведено емпіричний аналіз роботи рекомендаційної системи на базі даних реальни відгуків із сайту IMDB.

З метою проведення оцінки ефективності запропонованої у даному дослідженні моделі, було використано чотири критерії, які були відповідно розміщені у матриці помилок.

Так як точність і повнота – іноді непорівнювані метрики, у дослідженні було вирішено застосувати $F$-метрику – середньозважена гармонійний точність та повнота, яка надає можливість ефективніше оцінити роботу моделі з точки зру повнішої перспективи.

Аналіз тональності тексту дозволяє оптимізувати рекомендаційни йсписок фільмів. Відповідно, у дослідженні було доведено, що застосування фільтрації на основі контенту та колаборативної фільтрації допомогає удосконалити існуючу модель формування рекомендаційної системи.

Було проілюстровано, що модель, яку представлено у даному дослідженні, працює більше ефективно, ніж базові рекомендації з точки зору частоти TP (TP-rate). Це означає, що запропонована у дослідженні модель є потужнішою у своїй здатності ідентифікувати відповідні фільми користувачу.

# ВИСНОВКИ

Система рекомендації для мобільних пристроїв вимагає як точності, так і швидкодії. У цьому документі запропоновано рекомендаційна система, який базується на гібридному аналізі рекомендацій та тональності, для підвищення точності систем рекомендацій. Крім того, Spark використовується для покращення швидкодії системи.

Запропонований нами спосіб дозволяє користувачам легко та швидко отримувати корисні пропозиції фільмів. Рекомендація щодо фільмів – це комплексна задача, яка включає в себе різні види користувачів та різні види фільмів. Зважаючи на корисну інформацію, що її можна отримати з відгуків, опублікованих користувачами, колаборативна фільтрація вважається найпопулярнішою та найрозповсюдженішою технікою рекомендацій. Більше того, через характеристики рекомендацій фільмів, історія перегляду користувача є дуже важливою, тому було додано метод рекомендацій на основі вмісту для спільної роботи, щоб створити гібридну систему рекомендацій. Крім того, краще враховувати позитивну та негативну інформацію під час аналізу системи рекомендацій. Загалом, люди схильні вважати, що позитивні відгуки мають позитивний вплив, а негативні відгуки мають негативні ефекти.

Аналіз тональності допомагає покращити точність результатів рекомендацій. Крім того, як було показано в експериментальних результатах, необхідно використовувати розподілену систему для вирішення масштабованості та швидкодії системи рекомендацій.

Пропонована структура може бути вдосконалена у кількох аспектах. По-перше, цей метод можна перевірити з більшою кількістю наборів даних. Різні дані можуть використовуватися різним аналізом тональності, тому

модель може бути налаштована для застосування у більшій кількості ситуацій. По-друге, у процесі аналізу тональності неминуче впливають різні види суб'єктивних ідей, що створює вплив на результати певних несприятливих ефектів. Тому майбутня робота може бути зосереджена на усуненні індивідуальних характеристик, прихованих у відгуках від користувачів.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ


1. C. Basu, H. Hirsh, and W. Cohen, “Recommendation as clas- si cation: Using social and content-based information in rec- ommendation,” in Proceedings of the 1998 15th National Confer- ence on Arti cial Intelligence, AAAI, pp. 714–720, July 1998.
2. W. Xu, J. Wang, Z. Zhao, C. Sun, and J. Ma, “A Novel Intel- ligence Recommendation Model for Insurance Products with Consumer Segmentation,” Journal of Systems Science and Infor- mation, vol. 2, no. 1, pp. 16–28, 2014.
3. Y.Xu,X.Guo,J.Hao,J.Ma,R.Y.K.Lau,andW.Xu,”Com- bining social network and semantic concept analysis for person- alized academic researcher recommendation,” Decision Support Systems, vol. 54, no. 1, pp. 564–573, 2012.
4. D. Guo, Z. Zhao, W. Xu et al., “How to nd a comfortable bus route – Towards personalized information recommendation services,” Data Science Journal, vol. 14, article no. 14, 2015.
5. D. Guo, Y. Zhu, W. Xu, S. Shang, and Z. Ding, “How to nd appropriate automobile exhibition halls: Towards a personal- ized recommendation service for auto show,” Neurocomputing, vol. 213, pp. 95–101, 2016.
6. V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang, “Collabo- rative ltering meets mobile recommendation: a user-centered approach,” in Proceedings of the 24th AAAI Conference on Arti cial Intelligence, pp. 236–241, 2010.
7. V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang, “Towards mobile intelligence: learning from GPS history data for collaborative recommendation,” Arti cial Intelligence, vol. 184/185, pp. 17–37, 2012.

8. M.-H. Park, J.-H. Hong, and S.-B. Cho, “Location-based rec- ommendation system using Bayesian user’s preference model in mobile devices,” in Proceedings of the 4th International Confer- ence on Ubiquitous Intelligence and Computing, pp. 1130–1139, 2007.
9. C. Ono, M. Kurokawa, Y. Motomura, and H. Asoh, “A context- aware movie preference model using a bayesian network for recommendation and promotion,” in Proceedings of 11th Inter- national Conference on User Modeling, pp. 247–257, 2007.
10. M. Szomszor, C. Cattuto, H. Alani et al., “Folksonomies, the semantic web, and movie recommendation,” in Proceedings of 4th European Semantic Web Conference, pp. 1–14, 2007.
11. T. De Pessemier, T. Deryckere, and L. Martens, “Context aware recommendations for user-generated content on a social network site,” in Proceedings of the EuroITV’09 – 7th European Conference on European Interactive Television Conference, pp. 133–136, Belgium, June 2009.
12. J. L. Herlocker, J. Konstan, A. Borchers, and J. Riedl, “An algo- rithmic framework for performing collaborative ltering,” in Proceedings of the 22nd Annual International ACM SIGIR Con- ference on Research and Development in Information Retrieval (SIGIR ’99), pp. 230–237, Berkeley, Calif, USA, August 1999.
13. Y. Koren, “Collaborative ltering with temporal dynamics,” Communications of the ACM, vol. 53, no. 4, pp. 89–97, 2010.
14. T. Hofmann, “Collaborative ltering via gaussian probabilistic latent semantic analysis,” in Proceedings of the 26th International ACM SIGIR Conference on Research and Development in Infor- maion Retrieval (SIGIR), pp. 259–266, Toronto, Canada, 2003.

15. Y. Zhang, D. Zhang, M. M. Hassan, A. Alamri, and L. Peng, “CADRE: Cloud-Assisted Drug REcommendation Service for Online Pharmacies,” Mobile Networks and Applications, vol. 20, no. 3, pp. 348–355, 2015.
16. B. Marlin, “Modeling user rating pro les for collaborative ltering,” in Proceedings of the 16th International Conference on Neural Information Processing Systems, pp. 627–634, 2003.
17. R. Salakhutdinov and A. Mnih, “Probabilistic matrix factor- ization,” in Proceedings of the 20th International Conference on Neural Information Processing Systems, pp. 1257–1264, 2007.
18. R. Salakhutdinov, A. Mnih, and G. Hinton, “Restricted Boltz- mann machines for collaborative ltering,” in Proceedings of the 24th International Conference on Machine learning (ICML ’07), vol. 227, pp. 791–798, Corvallis, Oregon, June 2007.
19. Y. Koren, “Factorization meets the neighborhood: a multi- faceted collaborative ltering model,” in Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discov- ery and Data Mining (KDD ’08), pp. 426–434, New York, NY, USA, August 2008.
20. S. Rendle, “Factorization machines,” in Proceedings of the 10th IEEE International Conference on Data Mining, ICDM 2010, pp. 995–1000, Australia, December 2010.
21. Y.Zhen,W.-J.Li,andD.-Y.Yeung,”TagiCoFi:Taginformedcol- laborative ltering,” in Proceedings of the 3rd ACM Conference on Recommender Systems, RecSys’09, pp. 69–76, USA, October 2009.
22. G. Lekakos and P. Caravelas, “A hybrid approach for movie recommendation,” Multimedia Tools and Applications, vol. 36, no. 1-2, pp. 55–70, 2008.
23. Y. Zhang, Z. Tu, and Q. Wang, “TempoRec: Temporal-Topic Based Recommender for Social Network Services,” Mobile Networks and Applications, vol. 22, pp. 1182–1191, 2017.

24. S. Debnath, N. Ganguly, and P. Mitra, “Feature weighting in content based recommendation system using social network analysis,” in Proceedings of the 17th International Conference on World Wide Web (WWW ’08), pp. 1041-1042, Beijing, China, April 2008.
25. M. Nazim Uddin, J. Shrestha, and G.-S. Jo, “Enhanced content- based ltering using diverse collaborative prediction for movie recommendation,” in Proceedings of the 2009 1st Asian Confer- ence on Intelligent Information and Database Systems, ACIIDS 2009, pp. 132–137, Viet Nam, April 2009.
26. A. Gunawardana and C. Meek, “A uni ed approach to building hybrid recommender systems,” in Proceedings of the 3rd ACM Conference on Recommender Systems, RecSys’09, pp. 117–124, USA, October 2009.
27. K. Soni, R. Goyal, B. Vadera, and S. More, “A ree Way Hybrid Movie Recommendation Syste,” International Journal of Computer Applications, vol. 160, no. 9, pp. 29–32, 2017.
28. G. Ling, M. R. Lyu, and I. King, “Ratings meet reviews, a com- bined approach to recommend,” in Proceedings of the 8th ACM Conference on Recommender Systems, RecSys 2014, pp. 105–112, USA, October 2014.
29. Y. Zhang, M. Chen, D. Huang, D. Wu, and Y. Li, “IDoctor: personalized and professionalized medical recommendations. Wireless Communications and Mobile Computing based on hybrid matrix factorization,” Future Generation Computer Systems, vol. 66, pp. 30–35, 2017.
30. B. Pang, L. Lee, and S. Vaithyanathan, “ umbs up?: sentiment classication using machine learning techniques,” in Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing—Volume 10 (EMNLP ’02), pp. 79– 86, Association for Computational Linguistics, Stroudsburg, Pa, USA, July 2002.

31. P. D. Turney, “ umbs up or thumbs down?” in Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 417–424, Philadelphia, Pennsylvania, July 2002.
32. J. R. Priester and R. E. Petty, “ e Gradual reshold Model of Ambivalence: Relating the Positive and Negative Bases of Attitudes to Subjective Ambivalence,” Journal of Personality and Social Psychology, vol. 71, no. 3, pp. 431–449, 1996.
33. H. Li, J. Cui, B. Shen, and J. Ma, “An intelligent movie recom- mendation system through group-level sentiment analysis in microblogs,” Neurocomputing, vol. 210, pp. 164–173, 2016.
34. J. Sun, G. Wang, X. Cheng, and Y. Fu, “Mining a ective text to improve social media item recommendation,” Information Processing & Management, vol. 51, no. 4, pp. 444–457, 2015.
35. Q. Diao, M. Qiu, C.-Y. Wu, A. J. Smola, J. Jiang, and C. Wang, “Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS),” in Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD ’14), pp. 193–202, ACM, New York, NY, USA, August 2014.
36. Y.Zhang,”GroRec:AGroup-CentricIntelligentRecommender System Integrating Social, Mobile and Big Data Technologies,” IEEE Transactions on Services Computing, vol. 9, no. 5, pp. 786– 795, 2016.
37. D. Liu, W. Xu, W. Du, and F. Wang, “How to choose appropriate experts for peer review: An intelligent recommendation method in a big data context,” Data Science Journal, vol. 14, article no. 16, 2015.
38. W. Xu, J. Sun, J. Ma, and W. Du, “A personalized information recommendation system for RD project opportunity nding in big data contexts,” Journal of Network and Computer Applica- tions, vol. 59, pp. 362–369, 2016.

39. Y. Zhou, D. Wilkinson, R. Schreiber, and R. Pan, “Large-scale parallel collaborative ltering for the Net ix Prize,” in Proceedings of the 4th International Conference on Algorithmic Aspects in Information and Management, pp. 337–348, 2008.
40. Z.-D.ZhaoandM.-S.Shang,”User-based collaborative filtering recommendation algorithms on hadoop,” in Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining, WKDD 2010, pp. 478–481, ailand, January 2010.
41. J. Sun, W. Xu, J. Ma, and J. Sun, “Leverage RAF to nd domain experts on research social network services: A big data analytics methodology with MapReduce framework,” International Journal of Production Economics, vol. 165, pp. 185–193, 2015.
42. J. Jiang, J. Lu, G. Zhang, and G. Long, “Scaling-up item-based collaborative ltering recommendation algorithm based on Hadoop,” in Proceedings of the 7th IEEE World Congress on Services, pp. 490–497, IEEE, Washington, DC, USA, July 2011.
43. S. Panigrahi, R. K. Lenka, and A. Stitipragyan, “A Hybrid Distributed Collaborative Filtering Recommender Engine Using Apache Spark,” in Proceedings of the 7th International Conference on Ambient Systems, Networks and Technologies, ANT 2016 and the 6th International Conference on Sustainable Energy Information Technology, SEIT 2016, pp. 1000–1006, Spain, May 2016.
44. A. Wijayanto and E. Winarko, “Implementation of multi-criteria collaborative ltering on cluster using Apache Spark,” in Proceedings of the 2nd International Conference on Science and Technology-Computer, ICST 2016, pp. 177–181, Indonesia, October 2016.

45. R. Feldman and I. Dagan, “Knowledge discovery in textual databases (KDT,” in Proceedings of the First International Conference on Knowledge Discovery and Data Mining, pp. 112–117, 1995.
46. A. H. Tan, “Text mining: the state of the art and the challenges,” in Proceedings of the Paci c Asia Conference on Knowledge Discovery and Data Mining PAKDD 1999 Workshop on Knowledge Discovery from Advanced Databases, pp. 65–70, 1999.
47. A. Hotho, A. Nu¨rnberger, and G. Paaß, “A brief survey of text mining,” LDV Forum – GLDV Journal for Computational Linguistics and Language Technology, vol. 20, pp. 19–62, 2005.
48. H.-P.Zhang,H.-K.Yu,D.-Y.Xiong,andQ.Liu,”HHMM-based Chinese lexical analyzer ICTCLAS,” in Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing (SIGHAN ’03), pp. 184–187, Sapporo, Japan, July 2003.
49. Y.Wangand W.Xu,”LeveragingdeeplearningwithLDA-based text analytics to detect automobile insurance fraud,” Decision Support Systems, vol. 105, pp. 87–95, 2018.
50. F. Wu, Y. Huang, Y. Song, and S. Liu, “Towards building a high- quality microblog-speci c Chinese sentiment lexicon,” Decision Support Systems, vol. 87, pp. 39–49, 2016.
51. Y.Wang,W.Xu,andH.Jiang,”Usingtextminingandclustering to group research proposals for research project selection,” in Proceedings of the 48th Annual Hawaii International Conference on System Sciences, HICSS 2015, pp. 1256–1263, USA, January 2015.