

Технічні науки

УДК 004.4

Смачило Петро Ігорович

магістр

Національного університету «Львівська політехніка»

Смачило Петр Игоревич

магистр

Национального университета «Львовская политехника»

Smachylo Petro

Graduate Student of the

Lviv Polytechnic National University

Павич Наталія Ярославівна

кандидат технічних наук, доцент кафедри програмного забезпечення

Національний університет «Львівська політехніка»

Павыч Наталья Ярославовна

Кандидат технических наук, доцент кафедры программного обеспечения

Национальный университет «Львовская политехника»

Pavych Natalia

Candidate of Technic Sciences, Associate Professor of Software Department

Lviv Polytechnic National University

**АРХІТЕКТУРА ВЕБ-СИСТЕМИ ЗБОРУ БЛАГОДІЙНИХ КОШТІВ З
РЕКОМЕНДАЦІЙНИМ АЛГОРИТМОМ
АРХИТЕКТУРА ВЕБ-СИСТЕМЫ СБОРА БЛАГОТВОРИТЕЛЬНЫХ
СРЕДСТВ С РЕКОМЕНДАТЕЛЬНЫМ АЛГОРИТМОМ
ARCHITECTURE OF CHARITY FUNDRAISING WEB SYSTEM WITH
RECOMMENDATION ALGORITHM**

Анотація. В статті розглядається проектування системи збору благодійних коштів, що використовує різновид гібридного рекомендаційного алгоритму для надання рекомендацій елементів присутніх у базі даних.

Ключові слова: проектування, гібридний рекомендаційний алгоритм, надання рекомендацій, база даних.

Аннотация. В статье рассматривается проектирование системы сбора благотворительных средств, что использует разновидность гибридного рекомендательного алгоритма для предоставления рекомендаций элементов присутствующих в базе данных.

Ключевые слова: проектирования, гибридный рекомендательный алгоритм, предоставление рекомендаций, база данных.

Summary. The article examine the design of a charitable fundraising system that uses a variety of hybrid recommendation algorithm to provide recommendations for the elements present in the database.

Key words: design, hybrid recommendation algorithm, recommendations, database.

У наш час, коли все більше і більше проектів оцифровуються та перекваліфікуються на роботу онлайн, старі методи для збору благодійних коштів стають все менш ефективними. Користувачі, що відвідують ці портали бажають бачити простоту та мінімалізм у інтерфейсі, а також зручний функціонал. Але найголовніше для користувачів таких порталів – це прозорість їх діяльності. Благодійні проекти створені на порталі повинні давати користувачам змогу залишити коментар, прикріпити звітний файл, переглянути статистику. Це все допоможе створити довіру користувача до порталу, та позбавити його страху про втрату своїх коштів.

Так як концепція цього порталу передбачає створення кампаній самими користувачами, кількість кампаній може зрости дуже швидко і представлення їх посортованими по часу створення стане не ефективним. Багато кампаній може бути просто загубленими з часом і стати не доступними для користувача. Одним з рішень цієї проблеми, яке б могло дати сильний поштовх у розвитку онлайн систем для збору благодійних коштів може буде інтеграція рекомендаційного алгоритму у систему.

Основною метою проекту є збільшення об'єму зібраних коштів завдяки наданню користувачам рекомендацій елементів в залежності від їх діяльності у системі для збору благодійних коштів, а також дослідження рекомендаційних систем та методів їх побудови.

Системи, що використовують рекомендаційні системи, такі як Youtube та Amazon.com є схожі, обидві системи використовують сторінку персональних рекомендацій, а також проводиться аналіз історії переглядів користувача, та на основі цього аналізу непомітно пропонуються деякі елементи. Натомість можна вирізнити LinkedIn серед систем, алгоритм не фокусується лише на даних про поточного користувача, а застосовує підхід порівняння профілів та підбору подібних профілів. Використовуючи ці два підходи систем можна створити гібридний алгоритм.

На даний момент рекомендаційні алгоритми не широко застосовуються у сфері внесення благодійних коштів.

Використання рекомендаційного алгоритму у цій сфері може дозволити більш ефективно та швидко збирати кошти на проекти. У наступному джерелі[1] розглядається застосування рекомендаційного алгоритму у сумісній сфері – надання рекомендацій NGO проектів[2] волонтерам та надання рекомендацій волонтерів NGO. У джерелі описується, що рекомендаційний алгоритм полегшує задачу підтримки великої кількості даних, а вирішує проблему росту кількості користувачів, яким стає важче орієнтуватися при великій кількості даних.

На основі проведеного огляду систем, що використовують рекомендаційні алгоритми запроновано змішаний гібридний метод для використання у майбутній системі. Переваги методу є наступні: змішаний гібрид дозволяє уникнути проблеми із додаванням "нового продукту", не потребує даних від інших користувачів, може надавати рекомендацію користувачам з унікальними смаками, може рекомендувати нові і непопулярні елементи, може надавати пояснення, чому елемент був рекомендований, працює для будь якого елемента.

У випадку коли можливо робити велику кількість рекомендацій у реальному часі практично використовувати змішаний гібридний підхід[3], який використовує рекомендаційні підходи декількох типів. Цей підхід включає у себе метод контент типу, що працює на основі текстових описів елементів системи та спільної фільтрації про інтереси інших користувачів. Результати виконання обидвох підходів поєднуються у фінальній програмі. Змішаний підхід дає змогу оминати проблему додавання нового елемента: компонента, яка базується на контенті, може рекомендувати нові елементи, які ще не були оцінені користувачами.

Змішаний гібрид може включати такі проблеми:

- при поєднанні результатів підходів у одну сутність потрібні способи вирішення конфліктів;
 - змішаний метод припускає, що цінність різних методів є більш-менш рівною у просторі можливих елементів, але це не завжди так;
 - результати змішаного гібриду залежать від порядку виконання методів;
- потребує велику кількість ресурсів.

Змішаний гібридний метод можливо модифікувати наступним чином:

- окрім часто використовуваних підходів фільтрації на основі контенту та спільної фільтрації у змішаному гібриді, використовувати підхід на основі популярності;
- адаптувати змішаний метод для використання у системі збору коштів при рекомендації проектів;
- адаптувати метод поєднання результатів різних підходів відповідно до вимог користувачів системи.

Архітектура системи

У системі для збору благодійних коштів під час проектування запропоновано використати наступний архітектурний шаблон MVC. Модель-View-Controller[4] - це архітектурний шаблон, який розділяє додаток на три основні логічні компоненти: модель, вигляд та контролер. Кожен із цих компонентів створений для обробки конкретних аспектів розробки програми.

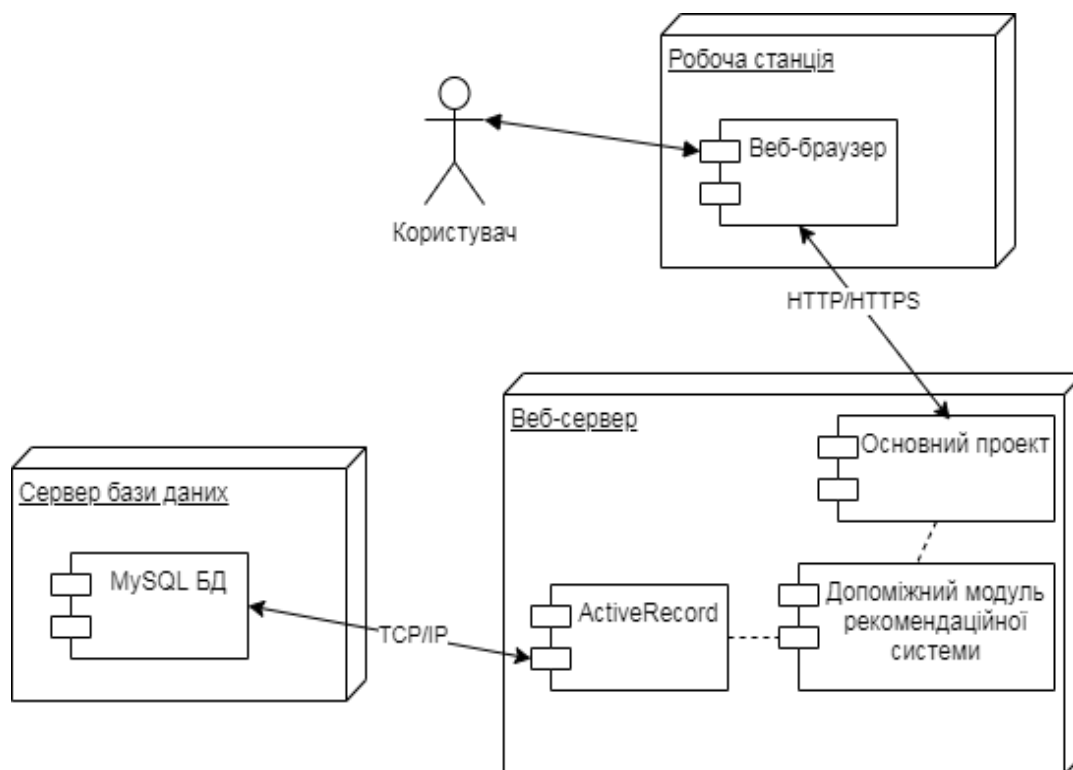


Рис. 1. Діаграма розгортання веб-системи

Модель є центральним компонентом архітектурного шаблону. Модель відповідає всій логіці, пов'язаній із даними, з якою працює користувач. У розробленій системі присутні наступні сутності моделей: News, Campaign, Comment, User, Privilege. Для задачі моніторингу роботи алгоритму додалася модель PerformanceData. Модель PerformanceData представляє дані про роботу рекомендаційних методів, включених у змішаний гібридний підхід. Модуль рекомендаційної системи безпосередньо працює з наступними моделями Campaign та User.

Контролер, приймає дані на вхід та перетворює їх у команди для моделі або вигляду. Він діє як інтерфейс між компонентами Модель і Вигляд для обробки всієї бізнес-логіки та вхідних запитів, маніпулює даними за допомогою компонента Модель і взаємодіє з виглядом, щоб зробити кінцевий результат. У системі присутні наступні контролери: NewsController, CampaignsController, UsersController, CommentsController. Рекомендаційний модуль безпосередньо використовується CampaignsController при виклику методу #index, що викликає фільтрацію даних для конкретного користувача та віддає дані опісля їх фільтрації для представлення. Також було додано контролер PerformanceDataController, що відповідає за витягування даних про роботу кожного з субпідходів та повернення цих статистичних даних для представлення.

Модуль рекомендаційної системи складається з таких класів:

- ContentTypRecommender - ідея фільтрації на основі вмісту полягає у позначенні продуктів за допомогою певних ключових слів, розумінні того, що подобається користувачеві, пошуку цих ключових слів у базі даних та рекомендації різних продуктів з однаковими атрибутами. Клас міститиме функції підгрування даних, створення TF-IDF вектору, вирахування косинуса подібності та результуючий метод.

- CollaborativeFilteringRecommender - це техніка, яка дозволяє відфільтрувати елементи, які можуть сподобатися користувачеві, на основі реакцій подібних користувачів. Метод працює, шукаючи велику групу людей і знаходячи менший набір користувачів зі смаками, схожими на конкретного користувача. Він розглядає предмети, які їм подобаються, та поєднує їх, щоб створити рейтинговий список пропозицій. Клас міститиме функції витягування найближчого сусіда, підгрування даних, побудови моделі, отримання очікуваного рейтингу та результуючий метод.
- PopularityBasedRecommender - в основному він використовує елементи, які зараз у тренді. Наприклад, якщо який-небудь товар, який зазвичай купує кожен новий користувач, існує ймовірність, що він може запропонувати цей товар користувачеві, який щойно зареєструвався. Клас міститиме функції витягування даних, зєднання даних, групування даних та результуючих метод
- MixedGybridRecommender – клас відповідає за поєднання та сортування даних отриманих з класів трьох рекомендаційних методів. Клас міститиме функції витягування даних контент рекомендатора, колаборативного та заснованого на популярності, фільтрації поєднаних даних та результуючий метод.
- ResultsLogger – клас відповідає за збір даних про роботу кожного рекомендатора та запис їх у No SQL Базу даних. Він включатиметься у кожен з класів.

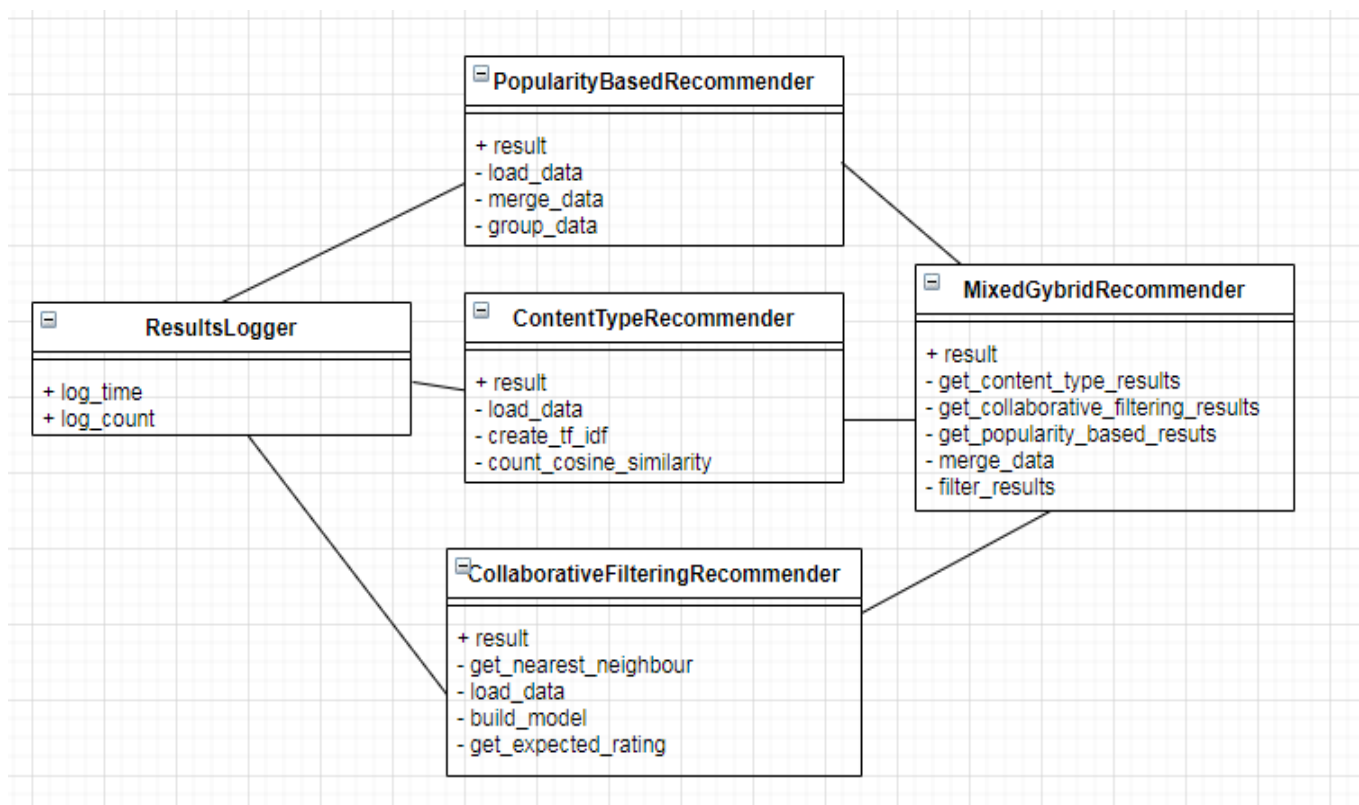


Рис. 2. Діаграма класів модуля рекомендаційної системи

Висновки. У статті запропоновано архітектуру веб-системи, та запропоновано діаграму класів для модулю рекомендаційної системи, представлено діаграми розгортання веб-системи та класів модуля рекомендаційної системи. Для вирішення завдання фільтрації проектів запропоновано гібридну систему рекомендацій, що поєднує рекомендації, що надходять від різних систем рекомендацій: на основі вмісту, спільний фільтруючий та на основі популярності.

Література

1. «Recommender System for volunteers in connection with NGO». URL: http://www.ijascse.org/volume-4-theme-based-issue-7/Recommender_system.pdf
2. «Non-governmental organization». URL: https://en.wikipedia.org/wiki/Nongovernmental_organizationRecommender_Systems_Survey_and_Experiments

3. «Mixed recommender system». URL:
<https://towardsdatascience.com/mixed-recommender-system-mf-matrix-factorization-with-item-similarity-based-cf-collaborative-544ddcedb330>
4. «MVC Framework - Introduction». URL:
https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm