

УДК 004.415:791

ПРОЄКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КІНОТЕАТРУ

Ткач Є.С., Колесник О.Б.

e-mail: yevhen.tkach1@nure.ua

Харківський національний університет радіоелектроніки, каф. СТ
м. Харків, Україна

In today's rapidly developing information technology environment, there are growing requirements for automating business processes in the entertainment sector. Cinemas, as one of the most popular types of entertainment services, strive to provide visitors with convenient services for choosing movies, buying tickets, and receiving recommendations for new screenings. The implementation of recommendation systems in cinema web applications allows to improve the quality of service by offering viewers personalized offers based on their preferences. The use of modern data analytics technologies allows cinemas to effectively manage their screening program and respond to the needs of visitors.

У сучасних умовах швидкого розвитку інформаційних технологій зростають вимоги до автоматизації бізнес-процесів у сфері розваг. Кінотеатри, як один із найпопулярніших видів розважальних послуг, прагнуть забезпечити відвідувачів зручними сервісами для вибору фільмів, покупки квитків та отримання рекомендацій щодо нових сеансів. Упровадження рекомендаційних систем у вебзастосунки кінотеатрів дозволяє підвищити якість обслуговування, пропонуючи глядачам персоналізовані пропозиції залежно від їхніх вподобань. Застосування підходів, які враховують історію перегляду або оцінювання фільмів користувачем, сприяє покращенню лояльності клієнтів та збільшенню продажів. Використання сучасних технологій аналізу даних дає змогу кінотеатрам ефективно управляти власною програмою показів і реагувати на потреби відвідувачів. Таким чином, створення інформаційної системи кінотеатру з розвиненою рекомендаційною підсистемою є актуальним завданням, яке дозволяє оптимізувати роботу закладу та підвищити його конкурентоспроможність на ринку розваг.

Нині більшість кінотеатрів пропонують стандартні функції: афішу фільмів, розклад сеансів, онлайн-бронювання квитків, особистий кабінет користувача. Проте цим системам часто бракувало гнучких механізмів персоналізації. Розроблена нами система успішно розв'язує цю проблему завдяки інтегрованій рекомендаційній компоненті. У ній користувач не змушений самотійно переглядати весь каталог доступних фільмів, адже персональні пропозиції формуються на основі його попередніх оцінок, жанрових вподобань, історії замовлень або рейтингу улюблених акторів чи режисерів. Це не лише покращує враження відвідувачів, але й допомагає

кінотеатру продавати більше квитків завдяки точнішому влучанню в інтереси кожного глядача.

Систему реалізовано в триланковій архітектурі: фронтенд забезпечує зручну взаємодію з користувачем, бекенд опрацьовує логіку бронювання й рекомендацій, а база даних зберігає інформацію про фільми, користувачів і куплені квитки. На рівні клієнта застосовано фреймворк Angular, що уможлиблює динамічне відображення інформації та оновлення в реальному часі. Рівень бізнес-логіки реалізовано за допомогою ASP.NET Core. Поєднання .Net та Angular вважається найкращим поєднанням завдяки економічній ефективності, відокремленій кодовій базі та пластичності стеку інтерфейсу користувача [1]. Нарешті, поєднання цих двох фреймворків може підвищити безпеку, швидкість та ефективність додатку.

Також ця система доповнена спеціалізованим сервісом для обчислення рекомендацій на базі .NET for Apache Spark. Він надає .NET прив'язки для Spark, що дозволяє отримати доступ до API Spark через C# та F#. За допомогою .NET для Apache Spark стає можливим писати і виконувати користувацькі функції для Spark, написані на .NET [2]. Рекомендаційна підсистема містить контент-орієнтований модуль (фільтрація на основі вмісту), побудований на жанрових, акторських та інших характеристиках фільмів.

Існують два основні підходи до рекомендацій у кінотеатральних системах: контент-орієнтований та колаборативний. Перший аналізує вподобання конкретного користувача, враховуючи історію переглядів, жанри, рейтинги та улюблених акторів. Завдяки цьому система пропонує персоналізовані фільми, що точно відповідають його інтересам.

Контент-орієнтований підхід працює незалежно від активності інших користувачів і пояснює логіку своїх рекомендацій. Однак він може звужувати вибір, пропонуючи лише схожий контент. Колаборативний підхід, навпаки, аналізує вибір схожих користувачів, відкриваючи нові варіанти перегляду. Водночас він має суттєві недоліки: не працює для нових користувачів та фільмів, схильний до «масового ефекту» та маніпуляцій із популярністю [3].

Для кінотеатральної системи кращим є контент-орієнтований підхід, адже він забезпечує персоналізацію, працює навіть у малих масштабах і не залежить від активності інших. Користувач отримує рекомендації одразу, без очікування накопичення даних про інших глядачів, що покращує його досвід і сприяє зростанню продажу квитків.

Фільтрація на основі вмісту створює профілі користувачів, вивчаючи особливості елементів, які користувачі раніше споживали, такі як ключові слова, теги або метадані. Ці ознаки зазвичай отримують за допомогою методів інтелектуального аналізу тексту, і вони відображають інтереси користувача, виділяючи спільні характеристики елементів, з якими

користувач позитивно взаємодіє. Рекомендуючи нові товари, система порівнює характеристики кожного кандидата з профілем користувача, рекомендуючи товари, які найбільше відповідають відомим уподобанням користувача.

Поширеною мірою відповідності є косинус подібності, який обчислюється шляхом ділення точкового добутку двох векторів (наприклад, вектора профілю користувача і вектора характеристик товару) на добуток їхніх величин.

Це дає значення від 0 до 1, якщо використовуються лише невід'ємні ознаки, причому вищі показники вказують на сильніший збіг контенту. Хоча фільтрація на основі вмісту може бути високоефективною, коли описи товарів багаті, вона може призвести до «надмірної спеціалізації», якщо профілі користувачів стають занадто вузькими, обмежуючи доступ до несподіваних або нових товарів [4].

Контент-орієнтований підхід розробленої інформаційної системи полягає в тому, що для кожного користувача формується так званий «профіль вподобань» із даними про ключові атрибути продукту (для кінотеатру це жанр, актори, режисери, ключові слова сюжету), які найбільше йому подобаються.

Кожен фільм описано аналогічним набором ознак. Порівняння цих двох векторів – профілю користувача і вектора атрибутів фільму – визначає ступінь близькості між ними. Чим вища схожість, тим більша ймовірність, що саме цей фільм зацікавить глядача.

Найпоширеніший спосіб дістатися до цієї близькості – скористатися косинусною мірою. Якщо:

$p(u) = p_1(u), p_2(u), \dots, p_n(u)$ – є вектором «смаків» користувача u ;

$v(i) = v_1(i), v_2(i), \dots, v_n(i)$ – є вектором ознак елемента i , тоді міра

подібності обчислюється за формулою:

$$\text{sim}(u, i) = \frac{\sum_{k=1}^n p_k(u) * v_k(i)}{\sqrt{\sum_{k=1}^n (p_k(u))^2} * \sqrt{\sum_{k=1}^n (v_k(i))^2}}.$$

Чим ближче це значення до одиниці, тим більша схожість між профілем користувача та контентом фільму. На основі отриманих оцінок формують рейтинг, який відображає найперспективніші для користувача варіанти передусім.

Впровадження розробленої системи дозволить значно підвищити рівень персоналізації обслуговування та ефективніше задовольняти вподобання глядачів, створюючи для них комфортніші умови вибору фільмів. Завдяки гнучкій рекомендаційній підсистемі користувачі отримуватимуть цільові пропозиції, що враховують їхні інтереси, історію переглядів і вподобання, що суттєво покращить їхній досвід і сприятиме

зростанню продажів кінотеатру. Це, у свою чергу, зміцнить конкурентні позиції кінотеатру на ринку розваг, забезпечивши йому стабільний розвиток та підвищену рентабельність.

Список використаних джерел:

1. Use .NET for Apache Spark - Azure Synapse Analytics. Microsoft Learn: Build skills that open doors in your career. URL: <https://learn.microsoft.com/en-us/azure/synapse-analytics/spark/spark-dotnet> (дата звернення: 22.02.2025).

2. Rami N. Why Angular + ASP.Net Core is the Most Preferred Combination for web app development?. Medium. URL: <https://nisarg-c-metric.medium.com/why-angular-asp-net-core-is-the-most-preferred-combination-for-web-app-development-30fa7a491c58> (дата звернення: 26.02.2025).

3. Kumawat T. Basics of Content Based and Collaborative Based Recommendation Engines. Medium. URL: <https://medium.com/@tejpal.abhyuday/basics-of-content-based-and-collaborative-based-recommendation-engines-4aa704a3139> (дата звернення: 03.03.2025).

4. Lops P., de Gemmis M., Semeraro G. Content-based recommender systems // State of the art and trends. 2011. P.73-105. DOI:10.1007/978-0-387-85820-3_3