

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

Навчально-науковий інститут прикладного системного аналізу

Кафедра математичних методів системного аналізу

До захисту допущено:

Завідувач кафедри

_____ Оксана ТИМОЩУК

« ____ » _____ 2026 р.

Дипломна робота

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Системний аналіз і управління»

спеціальності 124 «Системний аналіз»

**на тему: «Система підтримки прийняття рішень персоналізації
маркетингових стратегій в медіаіндустрії на основі онтологічного
моделювання та мережі морфологічних таблиць»**

Виконав:

Студент IV курсу, групи КА-22

Ковальчук Олександр Миколайович _____

Керівник:

Доцент кафедри ММСА, д.т.н.

Савченко Ілля Олександрович _____

Консультант з економічного розділу:

Доцент кафедри ЕК, к.е.н.

Рощина Надія Василівна _____

Консультант з нормоконтролю:

Асистент кафедри ММСА, д-р. філософії

Канцедал Георгій Олегович _____

Рецензент:

Доцент кафедри СП, к.т.н.

Булах Богдан Вікторович _____

Засвідчую, що у цій дипломній роботі
немає запозичень з праць інших авторів
без відповідних посилань.

Студент _____

Київ – 2026 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Навчально-науковий інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу

Рівень вищої освіти – перший (бакалаврський)
Спеціальність – 124 «Системний аналіз»
Освітня програма «Системний аналіз і управління»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Оксана ТИМОЩУК
«___» _____ 2026 р.

ЗАВДАННЯ
на дипломну роботу студенту
Ковальчуку Олександр Миколайовичу

1. Тема роботи «Система підтримки прийняття рішень персоналізації маркетингових стратегій в медіаіндустрії на основі онтологічного моделювання та мережі морфологічних таблиць», керівник роботи Савченко Ілля Олександрович, доктор технічних наук, затверджені наказом по університету від 27.05.2026 р. № 1944-с.
2. Термін подання студентом роботи «10» червня 2026 р.
3. Вихідні дані до роботи: інформаційні матеріали з відкритих джерел стосовно проблематики кіноіндустрії та вибору маркетингових стратегій, дані з відкритих датасетів для наповнення бази знань, та експертні оцінки, побудовані власноруч.
4. Зміст роботи: дослідження предметної області; теоретичні засади онтологічного моделювання та модифікованого методу морфологічного аналізу; програмна реалізація системи підтримки прийняття рішень; функціонально-вартісний аналіз програмного продукту.
5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): ієрархія класів онтології, мережа морфологічних таблиць, інтерфейс програмного продукту, результати функціонально-вартісного аналізу.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Функціонально-вартісний аналіз	Рощина Н.В., доц., к.е.н.		

7. Дата видачі завдання _____ 27.05.2026 _____

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Затвердження теми бакалаврської дипломної роботи	18.02.2026-20.04.2026	виконано
2	Огляд літератури, державних стандартів та дослідження предметної області дипломної роботи.	20.04.2026-26.04.2026	виконано
3	Створення онтологічної бази знань, розробка морфологічних таблиць та надання експертних оцінок.	27.04.2026-10.05.2026	виконано
4	Розробка програмного забезпечення для персоналізації маркетингових стратегій та аналіз результатів.	11.05.2026-20.05.2026	виконано
5	Написання четвертого розділу.	21.05.2026-25.05.2026	виконано
6	Оформлення пояснювальної записки, перевірка правильності структури та проходження нормоконтролю.	26.05.2026-02.06.2026	виконано
7	Остаточне оформлення роботи, підготовка презентації та доповіді.	03.06.2026-10.06.2026	виконано

Студент _____

Олександр КОВАЛЬЧУК

Керівник _____

Ілля САВЧЕНКО

РЕФЕРАТ

Дипломна робота складається з 101 сторінки, містить 21 рисунок, 14 таблиць, 4 додатки та 16 джерел.

СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ, МЕДІАІНДУСТРІЯ, ОНТОЛОГІЯ, МММА, МЕРЕЖА МОРФОЛОГІЧНИХ ТАБЛИЦЬ, ПЕРСОНАЛІЗАЦІЯ.

Метою роботи є розробка системи підтримки прийняття рішень для персоналізації маркетингових стратегій у медіаіндустрії на основі онтологічного моделювання та мережі морфологічних таблиць.

Об'єктом дослідження є процес прийняття рішень щодо персоналізації маркетингових стратегій та формату релізу кінофільмів.

Предметом дослідження є двоетапний модифікований метод морфологічного аналізу з нерівномірним розподілом оцінок та онтологічні моделі.

В роботі проаналізовано чинники вибору стратегій просування фільмів, спроектовано автоматизовану OWL-онтологію та мережу морфологічних таблиць. Проведено функціонально-вартісний аналіз для обґрунтування оптимального варіанту реалізації.

Результатом є вебзастосунок з інтерактивним інтерфейсом, реалізований мовою Python з використанням бібліотеки owlready2 та фреймворку Streamlit.

Подальший розвиток передбачає розширення онтології та інтеграцію алгоритмів збору ринкових даних у реальному часі.

ABSTRACT

Diploma thesis consists of 101 pages, contains 21 figures, 14 tables, 4 appendices, and 16 sources.

DECISION SUPPORT SYSTEM, MEDIA INDUSTRY, ONTOLOGY, MMMA, NETWORK OF MORPHOLOGICAL TABLES, PERSONALIZATION.

The aim of this work is to develop a decision support system for personalizing media industry marketing strategies based on ontological modeling and a network of morphological tables.

The object of study is the decision-making process regarding the personalization of marketing strategies and film release formats.

The subject of the study is a two-stage modified morphological analysis method with an uneven distribution of estimates and ontological models.

The work analyzes film promotion factors, designs an automated OWL ontology, and builds a network of morphological tables. A functional-value analysis justifies the optimal implementation option.

The result is a web application developed in Python using the owlready2 library and the Streamlit framework.

Future development includes expanding the ontology and integrating algorithms for real-time media market data collection.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ.....	9
1.1 Загальна характеристика медіаіндустрії та її видів.....	9
1.2 Специфіка маркетингу та просування у кіноіндустрії.....	11
1.3 Огляд підходів та систем підтримки прийняття рішень у медіаіндустрії..	14
1.4 Висновки до розділу 1	17
РОЗДІЛ 2 ТЕОРЕТИЧНІ ЗАСАДИ ОНТОЛОГІЧНОГО МОДЕЛЮВАННЯ ТА МОРФОЛОГІЧНОГО АНАЛІЗУ.....	18
2.1 Основи теорії онтологій та інженерії знань.....	18
2.1.1 Формальне представлення та концептуалізація.....	19
2.1.2 Базові елементи та мови опису онтологій.....	21
2.1.3 Підходи до проєктування онтологій.....	23
2.2 Модифікований метод морфологічного аналізу.....	25
2.3 Двоетапний МММА з нерівномірним розподілом початкових оцінок.....	28
2.4 Висновки до розділу 2.....	31
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ.....	32
3.1 Розробка онтологічної бази знань.....	32
3.1.1 Проєктування структури онтології.....	32
3.1.2 Автоматизоване наповнення онтології.....	36
3.2 Мережа морфологічних таблиць.....	39
3.3 Проєктування та реалізація програмного інтерфейсу.....	44
3.4 Програмна реалізація основного алгоритму та аналіз проміжних результатів.....	47
3.5 Висновки до розділу 3.....	53
РОЗДІЛ 4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ.....	54
4.1 Постановка задачі проєктування.....	54

	7
4.2 Обґрунтування функцій програмного продукту.....	55
4.3 Обґрунтування систем параметрів програмного продукту.....	58
4.4 Аналіз експертного оцінювання параметрів.....	61
4.5 Аналіз рівня якості варіантів реалізації функції.....	64
4.6 Економічний аналіз варіантів розробки програмного продукту.....	66
4.7 Вибір кращого варіанту ПП за техніко-економічним рівнем.....	72
4.8 Висновки до розділу 4.....	74
ВИСНОВКИ.....	75
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	76
ДОДАТОК А МОДУЛЬ НАПОВНЕННЯ ОНТОЛОГІЇ.....	79
ДОДАТОК Б МАТЕМАТИЧНЕ ЯДРО СППР.....	84
ДОДАТОК В ВЕБ-ІНТЕРФЕЙС.....	89
ДОДАТОК Г ПРЕЗЕНТАЦІЙНІ МАТЕРІАЛИ.....	92

ВСТУП

Сучасна кіноіндустрія є сферою з надзвичайно високими фінансовими ризиками та коротким життєвим циклом продукту. Глобальний перехід до цифрового маркетингу вимагає від брендів глибоко персоналізованого підходу, оскільки простої трансляції рекламних повідомлень вже недостатньо. Більшість існуючих маркетингових систем покладаються переважно на загальні статистичні дані, що не дозволяє повною мірою врахувати складні індустріальні закономірності.

В умовах, коли на вибір стратегії впливає значна кількість факторів, інтуїції маркетолога не завжди достатньо для прийняття обґрунтованих рішень. Тому створення систем підтримки прийняття рішень, що базуються на поєднанні об'єктивних баз знань та математичних методів якісного аналізу, є надзвичайно актуальним завданням для підвищення ефективності рекламних кампаній.

Метою даної роботи є розробка системи підтримки прийняття рішень для персоналізації маркетингових стратегій у медіаіндустрії на основі онтологічного моделювання та мережі морфологічних таблиць.

Об'єктом дослідження є процес прийняття рішень щодо вибору та персоналізації маркетингових стратегій і формату релізу в кіноіндустрії.

Предметом дослідження є двоетапний модифікований метод морфологічного аналізу з нерівномірним розподілом початкових оцінок та онтологічні моделі представлення знань.

У роботі застосовується апарат теорії онтологій для побудови бази знань кіноіндустрії, а також двоетапний модифікований метод морфологічного аналізу, який є потужним інструментом якісного аналізу в задачах передбачення та прийняття рішень за умов багатоваріантності.

РОЗДІЛ 1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Загальна характеристика медіаіндустрії та її видів

У сучасному світі термін «медіа» є складною, багатоаспектною категорією, яка охоплює технічні засоби створення, тиражування та поширення інформації, а також процеси обміну нею між авторами та аудиторією. Будь-яка медіатизована комунікація є опосередкованою, оскільки вона обов'язково передбачає використання певних фізичних носіїв або засобів передачі повідомлення (від друкованого тексту до цифрових алгоритмів) [1].

Прикладним та економічним аспектом функціонування медіа є медіаіндустрія. Її можна визначити як складну сукупність підприємств та організацій, метою яких є отримання прибутку, забезпечення стійкого розвитку та задоволення інформаційних, соціальних і розважальних потреб споживачів шляхом масового виробництва та дистрибуції медіапродукту. Найсуттєвішою характеристикою медіаіндустрії, що докорінно відрізняє її від інших галузей сучасної економіки, є функціонування на здвоєному ринку (ринку подвійного медіапродукту) [1].

Цей механізм передбачає, що медіапідприємства одночасно конкурують на двох ринках:

- 1) ринок товарів (змісту), де в якості товару виступає контент, який продається або надається масовій чи сегментованій аудиторії;
- 2) ринок рекламних послуг, де товаром виступає вже сама аудиторія (її увага та вільний час), доступ до якої медіакомпанії продають рекламодавцям [1].

З економічної точки зору, масова або чітко сегментована аудиторія є головною основою фінансування медіа. Щоб утримувати увагу споживачів, медіапродукт набуває складної двоїстої природи. З одного боку, це продукт духовного виробництва, що впливає на свідомість людини, а з іншого – товар,

що має свою споживчу та ринкову вартість. Товар медіапідприємств у сучасних умовах є комплексним і об'єднує в собі чотири ключові компоненти:

- 1) зміст (безпосередньо інформація);
- 2) думки (ідеї та аналітика);
- 3) розваги (емоційне залучення);
- 4) знання (освітній та пізнавальний аспект) [1].

Історично та технологічно медіаіндустрію поділяють на дві великі групи: традиційні та нові медіа. До традиційного сектору належать періодичні друковані видання, мовні ЗМІ (ефірне радіо та телебачення), книговидавничий бізнес, а також кіновиробництво та музична індустрія. Проте під впливом стрімкого розвитку цифрових та мережевих технологій сформувався сектор нових медіа (інтернет-видання, соціальні мережі, блоги, OTT-сервіси). Принциповими ознаками, що відрізняють їх від традиційних засобів комунікації, є:

- 1) дігитальність (цифровий характер контенту) та гнучкість об'єднання різних видів інформації на різних платформах;
- 2) інтерактивність, що забезпечує миттєвий зворотний зв'язок і дозволяє користувачеві бути одночасно споживачем і автором контенту (UGC);
- 3) персоналізація, тобто автоматична фільтрація та видача інформації відповідно до унікальних інтересів конкретного користувача;
- 4) потоковість споживання контенту і необмеженість у його обсягах та просторі [1].

В умовах жорсткої конкуренції за увагу аудиторії медіаіндустрія активно залучає інноваційні методи просування та розповсюдження контенту, серед яких ключове місце займають кросмедійність та трансмедійність. Кросмедіа передбачає розповсюдження контенту через різні платформи в офлайн і онлайн режимі з використанням різних медіаформатів (наприклад, одночасно з виходом фільму з'являється саундтрек на радіо та відеогра). Натомість трансмедійний контент – це інноваційна модель сторітелінгу, яка передбачає, що сама історія розвивається одночасно на декількох платформах.

Кожен елемент трансмедійного контенту є лише частиною загальної історії, що змушує споживача переходити від одного медіаканалу до іншого для отримання цілісного досвіду [1].

На сучасному етапі медіаіндустрія невіддільна від індустрії вільного часу та розваг. Вільний час людей – це найцінніший ресурс для медіаіндустрії, оскільки саме в цей час аудиторія споживає медіапродукти. Згідно з останніми дослідженнями світового медіаринку, зокрема звітом компанії Deloitte, традиційні медіакомпанії сьогодні змагаються з великими технологічними гігантами за обмежений час користувачів. Соціальні медіа та контент, створений незалежними авторами, поступово витісняють традиційні розваги: понад половина представників молодих поколінь зазначають, що контент у соціальних мережах є для них більш релевантним, ніж класичні телешоу чи фільми [2]. Це вимагає від медіапідприємств, зокрема в кіноіндустрії, безперервно еволюціонувати, шукати нові гібридні формати взаємодії з глядачем та впроваджувати персоналізовані цифрові стратегії [2].

1.2 Специфіка маркетингу та просування у кіноіндустрії

Сучасна кіноіндустрія є сферою з надзвичайно високими фінансовими ризиками та відносно коротким життєвим циклом продукту. Будь-який фільм потребує значних інвестицій ще до моменту виходу на екран, а його комерційний успіх безпосередньо залежить від того, наскільки ефективно побудована комунікація з цільовою аудиторією. Як і медіаіндустрія загалом, ринок кіно функціонує на здвоєному ринку, де виробники з одного боку пропонують глядачам унікальний контент, а з іншого – змагаються за їхній обмежений час та увагу в умовах надзвичайно високої конкуренції з іншими формами дозвілля [1].

Маркетингові стратегії у кіно зазнали радикальної еволюції в епоху глобалізації та цифровізації. Традиційно кіномаркетинг зосереджувався на перериванні аудиторії через конвенційні канали мовлення: створенні дорогих театральних трейлерів, купівлі прайм-тайму на телебаченні, радіо та використанні зовнішньої реклами на білбордах. Сьогодні ж акцент змістився з пасивної односторонньої комунікації на активне двостороннє залучення глядачів та побудову навколо фільму спільноти [3]. Загалом, інструменти просування сучасних фільмів можна класифікувати на три основні групи:

- 1) офлайн інструменти, до яких належить традиційна реклама (телебачення, друковані ЗМІ, зовнішня реклама) та різноманітні PR-заходи (прем'єри, прес-тури, мерчендайзинг);
- 2) онлайн інструменти, що охоплюють маркетинг у соціальних мережах (SMM), контент-маркетинг та цифрову таргетовану рекламу;
- 3) нестандартні інструменти, які включають залучення лідерів думок (Influencer-маркетинг), партизанський маркетинг та гейміфікацію [1, 3].

Соціальні медіа здійснили революцію у просуванні кіно. Провідні студії наразі виділяють до 50 % свого маркетингового бюджету на інтернет-кампанії, оскільки ці платформи дозволяють взаємодіяти з фанатами задовго до релізу стрічки. Особливої популярності набули платформи для створення коротких вертикальних відео, такі як TikTok, Instagram Reels та Snapchat. Формат відео тривалістю до 60 секунд є ідеальним для поширення тизерів, кадрів зі знімального майданчика та інтерв'ю з акторами, адаптованих для перегляду з мобільних пристроїв. Такі платформи заохочують створення користувацького контенту (UGC), танцювальних челенджів та мемів, що забезпечує органічне поширення інформації про фільм з використанням рекомендаційних алгоритмів [3].

Вагому роль у нестандартному просуванні відіграє маркетинг впливу (Influencer-маркетинг). На відміну від традиційної реклами, сучасні глядачі схильні більше довіряти блогерам та інфлюенсерам, з якими у них формується міцна парасоціальна взаємодія. Співпраця навіть з мікро- та нано-

інфлюенсерами дозволяє студіям досягати високого рівня залученості специфічних нішевих аудиторій, генеруючи щирі рекомендації та позитивний сарафанний маркетинг, що конвертується у значне зростання касових зборів у прем'єрні вихідні [3, 4].

Поряд зі зміною методів просування, цифровізація фундаментально трансформувала і канали дистрибуції фільмів. Традиційно дистрибуція була централізованою і відбувалася виключно в кінотеатрах або на фізичних носіях (DVD, Blu-Ray). Проте розвиток інтернету, технологій Web 3.0 та швидкісних мереж 5G призвів до масового поширення технології Video On Demand (VOD) та стрімкого розвитку OTT-платформ (Over-the-Top), таких як Netflix, Disney+, HBO, Hulu [5]. Це зумовило появу нових форматів випуску кінопродукту:

- 1) кіно-реліз, тобто традиційний вихід фільму виключно у прокат в кінотеатрах;
- 2) онлайн-ексклюзив, що передбачає випуск фільму безпосередньо на потокових OTT-сервісах, оминаючи великі екрани;
- 3) гібридний реліз, коли фільм одночасно (або з мінімальним вікном) з'являється як у кінотеатрах, так і на стримінгових сервісах.

Масовий перехід аудиторії до OTT-платформ зумовлений їхньою зручністю, гнучкістю та можливістю насолоджуватися контентом у будь-який час та в будь-якому місці. Водночас індустрія стикається і з серйозними викликами цифрової епохи, такими як піратство та монополізація ринку дистрибуції кількома великими технологічними гігантами, що ускладнює конкуренцію для менших гравців [5].

Щоб залишатися конкурентоспроможними в умовах надлишку контенту, кінокомпаніям доводиться переходити до дата-орієнтованих персоналізованих стратегій маркетингу. Використовуючи алгоритми машинного навчання та аналіз великих даних (історія переглядів, активність у соцмережах, пошукові запити), маркетологи відходять від широких демографічних припущень. Натомість створюються вузькоспеціалізовані рекламні матеріали для конкретних кластерів аудиторії. Наприклад, для однієї групи глядачів

монтується трейлер з акцентом на візуальні спецефекти, а для іншої – той самий фільм рекламується через призму драматичної історії чи романтичної лінії [3].

Більше того, дедалі частіше у маркетингу кіно інтегруються технології віртуальної та доповненої реальності. Вони пропонують глядачам ефект глибокого занурення у світ стрічки ще до її виходу в прокат. Це можуть бути віртуальні прем'єри за участю аватарів акторів, використання 3D-голографічних зображень персонажів при скануванні QR-кодів на постерах, або AR-квести в реальному середовищі міст. Такі інновації значно підвищують залученість аудиторії і створюють унікальний досвід взаємодії з медіапродуктом [3].

1.3 Огляд підходів та систем підтримки прийняття рішень у медіаіндустрії

Процес прийняття рішень у сучасній економіці, і зокрема в медіаіндустрії, є складною задачею, яка вимагає врахування великої кількості як кількісних, так і якісних факторів. Фундаментальним підходом до вирішення проблем багатокритеріального вибору є метод аналізу ієрархій (Analytic Hierarchy Process), розроблений американським математиком Томасом Сааті [6]. Цей метод дозволяє структурувати складну проблему у вигляді ієрархії: на вершині знаходиться головна мета, нижче розташовуються критерії та підкритерії, а на найнижчому рівні – альтернативні варіанти рішень. Основою методу Сааті є використання попарних порівнянь та суджень експертів. Ці порівняння здійснюються за допомогою абсолютної шкали суджень від 1 до 9, що дозволяє вимірювати нематеріальні фактори у відносних величинах [6]. Крім того, такий підхід дозволяє розглядати рішення з чотирьох різних позицій (модель BOCR): переваги (Benefits), можливості

(Opportunities), витрати (Costs) та ризики (Risks), що є вкрай важливим для медіабізнесу, де людська інтуїція та експертна оцінка контенту відіграють ключову роль [6].

Інформатизація суспільства призвела до активного впровадження комп'ютеризованих систем підтримки прийняття рішень (СППР). На сьогодні у кіноіндустрії значного поширення набули СППР, орієнтовані на кінцевого споживача, оскільки класичний поділ фільмів за жанрами є занадто загальним. Для вирішення цієї задачі застосовуються рекомендаційні СППР, побудовані на основі методу простого адитивного зважування (Simple Additive Weighting) [7]. Цей метод базується на зваженому середньому значенні: система обчислює оціночний бал для кожної альтернативи, перемножуючи масштабовані значення критеріїв на їхні вагові коефіцієнти, безпосередньо задані користувачем. В якості базових критеріїв для генерації рекомендацій використовуються рейтинг фільму, вікові обмеження, кількість жанрів та тривалість кінострічки [7].

З іншого боку, кіновідеовиробництво характеризується надзвичайно високими фінансовими ризиками, що створює потребу в аналітичних СППР для продюсерів та дистриб'юторів. Сучасні системи такого класу активно використовують алгоритми машинного навчання для аналізу великих масивів даних. Наприклад, існують розробки, що використовують алгоритм наївного байєсівського класифікатора (Naïve Bayes) для передбачення комерційного успіху фільмів та аналізу трендів [8]. Цей імовірнісний метод класифікації впливає з теореми Байєса і припускає умовну незалежність ознак. Він ефективно обробляє метадані фільмів та відгуки аудиторії, класифікуючи проєкти на «успішні» та «менш успішні» [8]. Завдяки таким СППР стейкхолдери можуть виявляти актуальні тенденції і приймати обґрунтовані стратегічні рішення [8].

Спираючись на ці наукові розробки, на сучасному ринку програмного забезпечення сформувався сегмент комерційних СППР, які пропонують кіностудіям готові інструменти для предиктивної аналітики. Під час

дослідження ринку комерційних рішень для кінобізнесу було виділено дві передові платформи.

1. Cinelytic [9] це аналітична платформа на базі штучного інтелекту, яка допомагає кіностудіям оцінювати комерційну цінність проекту ще на етапі сценарію, аналізуючи історичні дані прокату та популярність акторського складу.
2. Gower Street Analytics [10] це спеціалізована система, що фокусується на прогнозуванні касових зборів та оптимізації дат релізу фільмів, аналізуючи глобальні ринкові тенденції.

Незважаючи на високу ефективність, спільним недоліком існуючих академічних та комерційних систем є те, що вони спираються переважно на аналіз історичних та статистичних даних. Проте інноваційні медіапроекти стикаються з унікальними умовами просування, де минулої статистики може бути недостатньо. Виникає обґрунтована потреба в інструментах, здатних об'єднати об'єктивні характеристики контенту з якісними експертними судженнями, оскільки традиційні алгоритми машинного навчання не завжди здатні працювати з високим рівнем невизначеності на ранніх етапах планування.

Найефективнішим підходом для вирішення задач із нечіткою або неповною інформацією є морфологічний аналіз та його модифікації. Використання модифікованого методу морфологічного аналізу (МММА) дозволяє синтезувати нові рішення, враховуючи якісні показники, індустріальні обмеження та експертну інтуїцію. Застосування такого математичного апарату створює міцне підґрунтя для розробки СППР нового покоління, які будуть здатні генерувати найбільш реалістичні конфігурації рішень та допомагати в ухваленні стратегічних маркетингових планів в умовах високої ринкової невизначеності.

1.4 Висновки до розділу 1

У першому розділі було досліджено предметну область функціонування сучасної медіаіндустрії та специфіку маркетингу в кінематографі. Встановлено, що ринок кіно є сферою з надзвичайно високими фінансовими ризиками, який функціонує за принципами зведеного ринку. Доведено, що в епоху цифровізації, розвитку OTT-платформ та надлишку контенту традиційні масові рекламні кампанії втрачають свою ефективність. Натомість критичного значення набувають дата-орієнтовані персоналізовані стратегії, які активно використовують інструменти онлайн-просування, Influencer-маркетинг, гейміфікацію та новітні гібридні формати релізу.

Проведений огляд існуючих та комерційних систем підтримки прийняття рішень у кіноіндустрії показав їхню високу ефективність у прогнозуванні трендів. Однак було виявлено їхній спільний суттєвий недолік: існуючі СППР спираються переважно на обробку історичних статистичних масивів і не здатні повною мірою враховувати якісні експертні оцінки, нетипові маркетингові конфігурації та індустріальні обмеження, що виникають під час просування унікальних медіапроектів.

З огляду на це, обґрунтовано об'єктивну потребу у розробці СППР, яка дозволить гнучко поєднувати статистичні характеристики контенту з людською інтуїцією та експертним досвідом. Найбільш релевантним математичним апаратом для розв'язання цієї задачі в умовах високої ринкової невизначеності визначено двоетапний модифікований метод морфологічного аналізу (МММА) з нерівномірним розподілом початкових оцінок у поєднанні з базами знань на основі онтологічного моделювання. Теоретичні засади та процедура застосування цих методів будуть детально розглянуті у наступному розділі роботи.

РОЗДІЛ 2 ТЕОРЕТИЧНІ ЗАСАДИ ОНТОЛОГІЧНОГО МОДЕЛЮВАННЯ ТА МОРФОЛОГІЧНОГО АНАЛІЗУ

2.1 Основи теорії онтологій та інженерії знань

У сучасній інформатиці та інженерії знань онтології відіграють ключову роль у формалізованому представленні складних предметних областей. Загальновизнано, що основними цілями створення онтологій є забезпечення спільного розуміння структури інформації між людьми та програмними агентами, уможливлення повторного використання знань, явне визначення припущень предметної області та відокремлення доменних знань від операційних [11].

Поняття онтології еволюціонувало від філософської категорії до строгого інженерного артефакту, що має під собою математичне підґрунтя. Найвідоміше визначення, запропоноване Т. Грубером, стверджує, що онтологія – це явна специфікація концептуалізації [11]. Згодом це визначення було уточнено Н. Гуаріно: онтологія є логічною теорією, яка пояснює передбачуване значення формального словника, тобто його онтологічну прихильність до певної концептуалізації світу [12]. На практиці, в контексті сучасного розроблення інформаційних систем та Семантичного павутини (Semantic Web), онтологію часто визначають еквівалентно до бази знань, побудованої на основі дескрипційних логік [12].

Для того, щоб онтологія могла бути ефективно використана програмними агентами та автоматичними «ризонерами», вона повинна спиратися на строгу формальну семантику, що гарантує однозначність інтерпретації всіх концептів та відношень. У наступних підрозділах детально розглянуто математичні засади побудови онтологічних моделей, їхню логічну архітектуру та специфіку роботи з відкритими даними.

2.1.1 Формальне представлення та концептуалізація

Фундаментальною основою для формального представлення знань в онтологіях є логіка першого порядку та її розв'язні фрагменти – дескрипційні логіки. Будь-яке онтологічне моделювання починається з вибору філософської концептуалізації. В інженерії знань виділяють дві основні доктрини: емпіричну (терміни посилаються на реально існуючі сутності у світі, що не залежать від свідомості) та концептуальну (терміни посилаються на концепти, тобто психологічні чи абстрактні формальні сутності у свідомості) [12]. На рівні математичної логіки ці відмінності зводяться до побудови коректної моделі інтерпретації.

Універсум та функція інтерпретації Семантика онтологічних мов строго визначається через поняття інтерпретації I . Будь-яка інтерпретація складається з непорожньої множини Δ^I , яка називається універсумом (або доменом інтерпретації), та функції інтерпретації $\cdot^I$. Функція інтерпретації працює наступним чином:

- 1) кожному концепту (класу) вона зіставляє певну підмножину універсуму:
 $C^I \subseteq \Delta^I$;
- 2) кожній ролі (властивості або відношенню) зіставляється бінарне відношення на універсумі: $R^I \subseteq \Delta^I \times \Delta^I$;
- 3) кожному індивіду (конкретному об'єкту) зіставляється конкретний елемент з універсуму: $a^I \in \Delta^I$ [12].

Онтологія складається з набору аксіом, які накладають обмеження на інтерпретацію. Якщо функція інтерпретації задовольняє всі без винятку аксіоми онтології, то така інтерпретація називається моделлю. З точки зору семантики, модель є абстракцією певного стану світу. Оскільки аксіоми онтології зазвичай фіксують лише часткові знання про предметну область, завжди існує багато різних станів світу, які можуть задовольняти онтологію

[12]. Логічний наслідок з онтології вважається істинним виключно тоді, коли він є істинним у всіх можливих моделях, які задовольняють цю онтологію.

Для строгої формалізації знань архітектура бази знань у дескрипційній логіці структурно поділяється на складові, що відповідають за інтенціональні та екстенціональні знання.

1. Інтенціональні знання (TBox – Terminological Box) описують саму термінологію, структуру концептів та їхні властивості. Це знання про класи та їх підпорядкування. На цьому рівні формуються загальні правила та обмеження предметної області.
2. Екстенціональні знання (ABox – Assertional Box) описують конкретні факти про індивідів у домені, їх належність до класів та зв'язки між ними через ролі. Тобто це конкретні екземпляри [12].

Поділ на інтенціональну та екстенціональну частини дозволяє ефективно використовувати автоматичні ризонери, які окремо виконують перевірку логічної сумісності термінології та класифікацію конкретних індивідів.

Однією з найголовніших концептуальних відмінностей між онтологічними базами знань та класичними реляційними базами даних є механізм роботи з неповною інформацією. Реляційні бази даних функціонують за припущенням закритого світу: будь-який факт, який явно не збережений у базі як істинний, автоматично вважається хибним.

Натомість онтології спираються на припущення відкритого світу. Згідно з OWA (Open World Assumption), відсутність твердження в онтології не означає, що воно є хибним; це лише означає, що системі цей факт наразі невідомий [13]. Оскільки Інтернет та будь-які інтелектуальні системи є відкритими середовищами, ми не можемо припускати, що вся повнота інформації вже інтегрована в нашу базу [13].

Такий підхід захищає систему від поспішних висновків. Наприклад, якщо для об'єкта явно не вказано наявності певної властивості, ризонер розглядатиме всі можливі моделі світу – як ті, де об'єкт має цю властивість, так і ті, де не має. Відповідно, семантика таких логік є монотонною: додавання

нових знань в онтологію може лише звузити множину можливих моделей та розширити перелік логічних висновків, але воно ніколи не скасовує істинність тих фактів, які були логічно виведені раніше [12].

2.1.2 Базові елементи та мови опису онтологій

Для того щоб концептуалізація предметної області могла оброблятися та аналізуватися машинними алгоритмами, вона має бути формалізована за допомогою спеціалізованих мов та структурних елементів. Сучасні онтології, зокрема ті, що побудовані для екосистеми Semantic Web, спираються на три фундаментальні складові: класи, індивіди та властивості [11, 13].

Класи є центральним елементом більшості онтологій і описують загальні концепти в предметній області. З математичної точки зору, клас являє собою множину, яка містить певну кількість індивідів, що об'єднані спільними ознаками. Класи організовуються в ієрархічну структуру – таксономію. Наприклад, клас «Собака» може бути підкласом «Тварина», що логічно означає: всі члени множини «Собака» автоматично є членами множини «Тварина». В онтологіях виділяють примітивні та визначені класи. Примітивні класи описуються лише необхідними умовами. Визначені класи описуються необхідними та достатніми умовами: якщо будь-який об'єкт задовольняє ці умови, машина логічного виведення автоматично класифікує його як екземпляр цього класу [13].

Індивіди представляють конкретні об'єкти реального світу в досліджуваній предметній області. Якщо клас – це множина, то індивід – це конкретний елемент цієї множини. Ключовою особливістю онтологій є те, що для індивідів не діє припущення про унікальність імен. Це означає, що дві різні назви можуть посилатися на одного й того самого індивіда, якщо явно не вказано, що ці сутності є різними [12, 13]. Такий підхід є критично важливим

для інтеграції баз знань, де одні й ті самі об'єкти можуть фігурувати під різними ідентифікаторами.

Властивості визначають бінарні відношення між сутностями. На відміну від об'єктно-орієнтованого програмування, де атрибути інкапсульовані всередині класів, в онтологіях властивості є самостійними сутностями першого рівня і існують незалежно від класів, і поділяються на дві основні категорії.

1. Об'єктні властивості (Object Properties) встановлюють зв'язок між двома індивідами. Наприклад, властивість `hasChild` пов'язує індивіда класу `Person` з іншим індивідом класу `Person`. Для них можна задавати логічні характеристики: транзитивність, симетричність, рефлексивність, функціональність та наявність оберненої властивості (наприклад, `hasParent` як обернена до `hasChild`).
2. Властивості даних (Data Properties) пов'язують індивіда з конкретним значенням певного типу даних (наприклад, `xsd:string`, `xsd:integer`, `xsd:dateTime`). Вони є аналогами атрибутів у реляційних базах даних [13].
Для того щоб описані елементи склалися в єдину базу знань, використовуються стандартизовані мови опису онтологій.

Фундаментом для подання знань у мережі є RDF. Основна ідея RDF полягає в тому, що будь-яка інформація або знання розкладається на атомарні висловлювання у вигляді трійок: Суб'єкт – Предикат – Об'єкт [12]:

- 1) суб'єкт і об'єкт представляють сутності, а предикат – відношення між ними;
- 2) кожен ресурс унікально ідентифікується за допомогою глобального ідентифікатора (URI або IRI), що робить знання машинозчитуваними та придатними для об'єднання у глобальний семантичний граф [12];
- 3) суб'єктом може бути лише ресурс, предикатом – властивість, а об'єктом – як інший ресурс, так і літерал.

Попри гнучкість RDF, його виразної здатності недостатньо для складного логічного моделювання. Тому консорціум W3C розробив мову

OWL (Web Ontology Language), яка базується на дескрипційних логіках – розв'язному фрагменті логіки першого порядку [12, 13]. OWL розширює словник RDF і додає строгу математичну семантику.

Завдяки OWL стає можливим не просто зберігати факти, а й задавати складні обмеження, такі як квантори існування та загальності, точні кардинальності та теоретико-множинні операції над класами [13]. Сучасний стандарт мови, OWL 2, поділяється на кілька профілів (OWL 2 EL, OWL 2 QL, OWL 2 RL), кожен з яких оптимізований під специфічні задачі та пропонує різний компроміс між виразністю мови та швидкістю автоматичного логічного виведення [12]. Використання OWL дозволяє автоматичним ризонерам перевіряти базу знань на несуперечливість та виводити нові, неявно задані знання з уже існуючих фактів.

2.1.3 Підходи до проєктування онтологій

Проєктування онтології – це складний та ітеративний процес, який не має єдиної універсальної методології. Не існує єдиного «правильного» шляху моделювання предметної області, оскільки найкраще рішення завжди залежить від того, для яких саме застосунків створюється онтологія та які розширення передбачаються в майбутньому [11]. Для систематизації цього процесу в інженерії знань вироблено низку практичних підходів.

Одним із найефективніших способів визначити межі майбутньої онтології є формування списку питань компетентності. Питання компетентності – це набір типових запитів, на які база знань, побудована на основі цієї онтології, повинна вміти давати правильні відповіді [12]. Вони відіграють роль своєрідного лакмусового папірця на етапі тестування: чи містить розроблена онтологія достатньо інформації, класів та відношень, щоб вирішити покладені на неї завдання. Концепти, які є остаточними відповідями

на найдетальніші з цих питань, зазвичай стають конкретними індивідами в побудованій базі знань [11].

При безпосередньому формуванні таксономії та наповненні онтології виділяють три базові підходи.

1. Підхід «згори-донизу». Процес моделювання розпочинається з визначення найзагальніших концептів предметної області, які на наступних етапах послідовно спеціалізуються та розбиваються на більш конкретні підкласи. Наприклад, спочатку створюється клас «Вино», який потім розбивається на «Червоне вино» та «Біле вино», а ті деталізуються до конкретних сортів [11]. У ширшому розумінні інженерії знань цей підхід також тісно пов'язаний із повторним використанням фундаментальних або базових онтологій, таких як DOLCE, BFO чи GFO. Замість того, щоб щось вигадувати, розробник успадковує готові високорівневі категорії та загальнофілософські принципи моделювання, адаптуючи їх під свою конкретну область [12].
2. Підхід «знизу-вгору». Цей метод діє у зворотному напрямку: розробка розпочинається з визначення найбільш специфічних, конкретних класів, які згодом об'єднуються у концепти вищого рівня спільності [11]. Сучасна інженерія знань активно використовує цей підхід для автоматизованого або напіваавтоматизованого видобування знань із уже існуючих джерел. Вхідними даними для побудови онтології «знизу-вгору» можуть слугувати схеми існуючих реляційних баз даних, електронні таблиці, неформалізовані тезауруси або ж масиви неструктурованих текстів, що обробляються за допомогою методів обробки природної мови (NLP) [12].
3. Комбінований підхід. Є поєднанням двох попередніх стратегій і на практиці виявляється найзручнішим для багатьох розробників. Процес відштовхується від найуживаніших, ключових концептів середнього рівня, які є найбільш очевидними та описовими для предметної області [11]. Визначивши це «ядро», розробник одночасно рухається вгору,

узагальнюючи концепти, та вниз, деталізуючи їх до специфічних підкласів [11].

Жоден із цих підходів не є апіорі кращим за інший. Вибір стратегії залежить від структури наявних даних, особистого бачення розробника та специфіки розв'язуваної задачі.

2.2 Модифікований метод морфологічного аналізу

У задачах системного аналізу, де об'єкти характеризуються високим рівнем невизначеності, неповнотою інформації та великою кількістю альтернативних конфігурацій, класичних кількісних методів часто буває недостатньо. Найбільш ефективним підходом для розв'язання таких проблем є модифікований метод морфологічного аналізу (МММА) [14, 15].

Основним об'єктом МММА є морфологічна таблиця (МТ). Вона складається з певної кількості N характеристичних параметрів F_i , де $i \in \overline{1, N}$. Кожному параметру F_i відповідає множина n_i альтернатив $a_j^{(i)}$, де $j \in \overline{1, n_i}$. Головною вимогою до формування цих множин є їх повнота та взаємовиключність, щоб вони охоплювали всі можливі стани системи. Конфігурацією МТ називається вектор, що містить по одній альтернативі для кожного параметра: $\{a_{j_1}^{(1)}, a_{j_2}^{(2)}, \dots, a_{j_N}^{(N)}\}$.

Класичний МММА реалізується як двоетапна процедура. На першому етапі аналізуються неконтрольовані фактори та формуються сценарії, а на другому – синтезуються та оцінюються стратегії рішень [14, 15].

Перший етап МММА: аналіз сценаріїв. Процедура розпочинається з отримання початкових (незалежних) оцінок ймовірностей $p_j^{(i)}$ для кожної альтернативи. У класичному підході ці оцінки отримують шляхом безпосереднього експертного оцінювання, де кожній альтернативі експерт

надає суб'єктивну оцінку за шкалою Міллера [14]. Відповідність якісних характеристик кількісним наведено у таблиці 2.1.

Таблиця 2.1 – Шкала Міллера для оцінювання альтернатив МТ

Якісна характеристика	Кількісна характеристика
Практично неможливо	$[0 \div 0,1]$
Дуже мала ймовірність	$[0,1 \div 0,25]$
Мала ймовірність	$[0,25 \div 0,4]$
Середня ймовірність	$[0,4 \div 0,6]$
Велика ймовірність	$[0,6 \div 0,75]$
Дуже велика ймовірність	$[0,75 \div 0,9]$
Практично гарантовано	$[0,9 \div 1]$

Окрім експертного оцінювання, в умовах цифровізації початкові наближення також можуть бути отримані автоматизовано – шляхом текстового аналізу масивів новин чи звітів. Для цього розраховується відношення сумарного ступеня впевненості згадок певної альтернативи у текстах до загальної кількості текстів [16].

Оскільки в реальних системах параметри завжди взаємопов'язані, для врахування цих зв'язків будується числова матриця взаємної узгодженості. Кожній парі альтернатив $a_{j_1}^{(i_1)}$ та $a_{j_2}^{(i_2)}$ різних параметрів присвоюється оцінка $c_{i_1 j_1, i_2 j_2}$ у діапазоні $[-1; 1]$ [14] (Таблиця. 2.2). При використанні текстового аналізу ці коефіцієнти можуть розраховуватись статистично як кореляція спільної частоти згадувань двох альтернатив у текстових фрагментах [16].

Таблиця 2.2 – Оцінки матриці взаємної узгодженості

Оцінка	Тлумачення
-1	Альтернативи повністю неузгоджені. Конфігурація неможлива.
(-1; 0)	Неузгоджені. Вибір однієї зменшує ймовірність вибору іншої.
0	Незалежні. Вибір однієї не впливає на вибір іншої.
(0; 1)	Узгоджені. Вибір однієї збільшує ймовірність вибору іншої.
1	Повністю узгоджені. Вибір однієї тягне за собою вибір іншої.

Після побудови матриці вибори альтернатив перестають бути незалежними. Для знаходження уточнених ймовірностей $p_j^{(i)}$ розв'язується нелінійна система рівнянь Байєса [14, 15]. У загальному вигляді для всіх параметрів вона записується так:

$$\begin{cases} p_1^{(1)} = \sum \sum \dots \sum P(\{a_1^{(1)}, a_{j_2}^{(2)}, \dots, a_{j_N}^{(N)}\} | a_{j_2}^{(2)}) \cdot p_{j_2}^{(2)}; \\ p_{n_N}^{(N)} = \sum \sum \dots \sum P(\{a_{j_1}^{(1)}, a_{j_2}^{(2)}, \dots, a_{n_N}^{(N)}\} | a_{j_1}^{(1)}) \cdot p_{j_1}^{(1)} \\ \sum_{j=1}^{n_1} p_j^{(1)} = 1; \quad \dots \quad \sum_{j=1}^{n_N} p_j^{(N)} = 1. \end{cases}$$

Розв'язавши цю систему, отримують оновлену МТ сценаріїв, що відображає реальні ймовірності виникнення станів системи з урахуванням усіх прихованих зв'язків.

Другий етап МММА: синтез стратегій. Метою другого етапу є розрахунок оцінок результативності кожної альтернативи з МТ стратегій в умовах сценаріїв, сформованих на першому етапі.

Відмінність полягає в тому, що параметри стратегій залежать від параметрів сценаріїв одностороннім зв'язком [14, 15]. Оскільки вибір альтернатив стратегій залежить від рішення суб'єкта, замість поняття «ймовірність» використовується величина «очікуваної результативності» (ефективності) E .

Будується матриця односторонньої узгодженості. Для кожної альтернативи стратегії $a_{j_{N+}}^{(i_{N+})}$ розраховується її умовна результативність $E(a_{j_{N+}}^{(i_{N+})} | V_k)$ за умови реалізації конкретного сценарію V_k [14]. Після цього повна очікувана результативність елемента стратегії обчислюється шляхом перебору всіх можливих сценаріїв та їхніх ймовірностей $P(V_k)$, формула:

$$E(a_{j_{N+}}^{(i_{N+})}) = \sum_k E(a_{j_{N+}}^{(i_{N+})} | V_k) \cdot P(V_k)$$

Значення $E(a_{j_{N+}}^{(i_{N+})})$ показує очікувану ефективність конкретної дії чи рішення з огляду на всю сукупність можливих реалізацій зовнішнього середовища. Отримані оцінки використовуються для ранжування альтернатив і побудови загальної «ідеальної» стратегії [14].

2.3 Двоетапний МММА з нерівномірним розподілом початкових оцінок

У класичному застосуванні модифікованого методу морфологічного аналізу початкові наближення ймовірностей альтернатив зазвичай

встановлюються експертами безпосередньо для досліджуваної системи і часто приймаються як рівномірні або визначаються за шкалою Міллера [14, 15].

Проте при розв'язанні задачі персоналізації маркетингових стратегій виникає специфічна ситуація: початкові оцінки пріоритетності задаються зовнішнім фактором, а саме профілем конкретного глядача, і є суттєво нерівномірними. Оскільки ці оцінки формуються виключно на основі суб'єктивних побажань користувача, їх пряме використання в системі рівнянь Байєса може призвести до генерації нереалістичних індустріальних конфігурацій.

Для вирішення цієї проблеми і математичного зв'язування профілю користувача з параметрами медіапродукту запроваджується попередній розрахунковий етап. За своїм математичним апаратом він є аналогічним до етапу синтезу стратегій класичного МММА, але з нетиповою зміною ролей: за таблицю сценаріїв приймаються характеристики користувача, а за таблицю стратегій – об'єктивні параметри цільової системи.

За допомогою матриці зв'язків між глядачем та фільмом розраховується умовна результативність R для альтернативи $a_j^{(i)}$ за умови реалізації конкретного профілю користувача V_k . Розрахунок здійснюється за формулою:

$$R(a_j^{(i)} | V_k) = \frac{p_j'^{(i)} \cdot \prod_{m=1}^N (c_{mj_m,ij} + 1)}{\sum_{k=1}^{n_i} (p_k'^{(i)} \cdot \prod_{m=1}^N (c_{mj_m,ik} + 1))}$$

Головна відмінність від класичного підходу полягає в тому, що базова оцінка $p_j'^{(i)}$ береться безпосередньо з нерівномірних оцінок споживача, а також доповнюється базовими частотами появи характеристик, видобутими з онтологічної бази знань, а не приймається як рівномірна константа.

Після розрахунку умовної результативності визначається повна очікувана результативність шляхом перебору всіх можливих конфігурацій. Отримані збалансовані значення приймаються як коректні початкові

ймовірності $p_j^{(i)}$ для запуску першого етапу класичного МММА. На цьому етапі за допомогою системи рівнянь Байєса [14] алгоритм вирішує внутрішні індустріальні конфлікти і відсіює нереалістичні комбінації. Зрештою, на другому етапі МММА найбільш імовірні конфігурації фільмів виступають сценаріями для синтезу кінцевих маркетингових стратегій.

2.4 Висновки до розділу 2

У розділі досліджено теоретичні засади онтологічного моделювання та модифікованого методу морфологічного аналізу, що є підґрунтям для розробки системи підтримки прийняття рішень.

Доведено, що для формалізації складної предметної області доцільно використовувати онтології. Застосування дескрипційних логік та концепції відкритого світу дозволяє створити машинозчитувану базу знань, здатну працювати з неповною інформацією та логічно виводити нові факти.

Обґрунтовано застосування двоетапного МММА для задач із високим рівнем невизначеності. Використання морфологічних таблиць, матриць узгодженості та нелінійних рівнянь Байєса дозволяє відсіяти індустріально неможливі сценарії і синтезувати ефективні стратегії.

Ключовою адаптацією методу стала розробка попереднього розрахункового етапу для роботи з нерівномірним розподілом початкових оцінок. Використання математичного апарату умовної результативності дозволяє коректно збалансувати суб'єктивні вподобання з об'єктивними даними кіноіндустрії. У підсумку, симбіоз онтологій та МММА формує надійне аналітичне ядро цільової системи.

РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

3.1 Розробка онтологічної бази знань

Розроблена в межах роботи онтологічна база знань виступає ключовим компонентом цільової системи підтримки прийняття рішень. Вона виконує дві стратегічні функції: по-перше, слугує формалізованою основою для подальшого проєктування мережі морфологічних таблиць; по-друге, виступає єдиним структурованим вмістилищем об'ємних масивів інформації про фільми, їхні характеристики та учасників знімального процесу. Такий підхід дозволяє органічно поєднати теоретичну модель кіноіндустрії з реальними даними для подальшої автоматизації роботи алгоритму СППР.

3.1.1 Проектування структури онтології

Основою для функціонування системи підтримки прийняття рішень для персоналізації маркетингових стратегій у медіаіндустрії є розроблена онтологічна база знань. Дана онтологія дозволяє формалізовано описати ключові сутності кіноіндустрії та інтегрувати профіль користувача з характеристиками медіапродукту й доступними рішеннями щодо просування.

В онтології розроблено розгалужену ієрархію, що включає всі необхідні концепти для формування мережі морфологічних таблиць. Нижче наведені основні класи онтології.

1. **Movie**: центральний концепт, що представляє фільми.
2. **Person**: описує учасників процесу та цільову аудиторію. Він поділяється на творців контенту (**Actor**, **Director**) та безпосередньо користувачів

(Viewer). Для забезпечення логічної цілісності класи Actor та Director визначені як логічно непересічні з класом Viewer.

3. Атрибути та параметри класифікації: набір допоміжних класів, що виступають параметрами для морфологічних таблиць. До них належать AgeCategory, BudgetCategory, DurationCategory, FameLevel, PlotType, ReleaseFormat, DigitalActivityLevel та PromotionStrategy.
4. Спеціалізовані ієрархії: клас Genre деталізується на конкретні жанри (Action, Comedy, Drama, History, Horror, Romance, Sci-Fi/Fantasy), а клас PromotionStrategy має розгалуження на OnlinePromotion, OfflinePromotion та NonStandartPromotion.

Дерево ієрархії розроблених класів онтології наведено на рисунку 3.1.

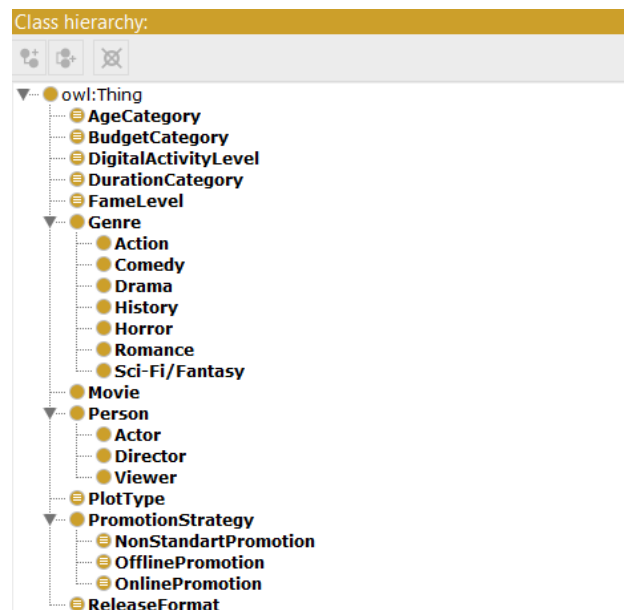


Рисунок 3.1 – Дерево ієрархії класів онтології

Для встановлення семантичних зв'язків між створеними класами було визначено розгалужену систему властивостей об'єктів.

1. HasContributor: базова властивість для зв'язку фільму з його творчою командою. Має деталізовані підвластивості hasActor та hasDirector.
2. HasAttribute: дозволяє описувати параметри самого фільму через підвластивості hasBudget, hasDuration, hasGenre, hasPlotType та isSuitableFor.

3. Опис користувача: розроблено окрему групу властивостей для профілювання глядача – `hasAge`, `hasDigitalActivity`, `hasGenrePreference` та `prefersReleaseFormat`. Окремо виділено властивість `isSusceptibleTo` та її зворотну пару `influencesViewer`.
4. Рівень відомості: для розрахунків у СППР впроваджено специфічні властивості `hasOverallCastFame` та `hasDirectorFameLevel`.
5. `HasStrategy`: пов'язує фільм із маркетинговими рішеннями через `hasPromotion` та `hasRelease`.

Загальну ієрархію властивостей об'єктів розробленої бази знань продемонстровано на рисунку 3.2.

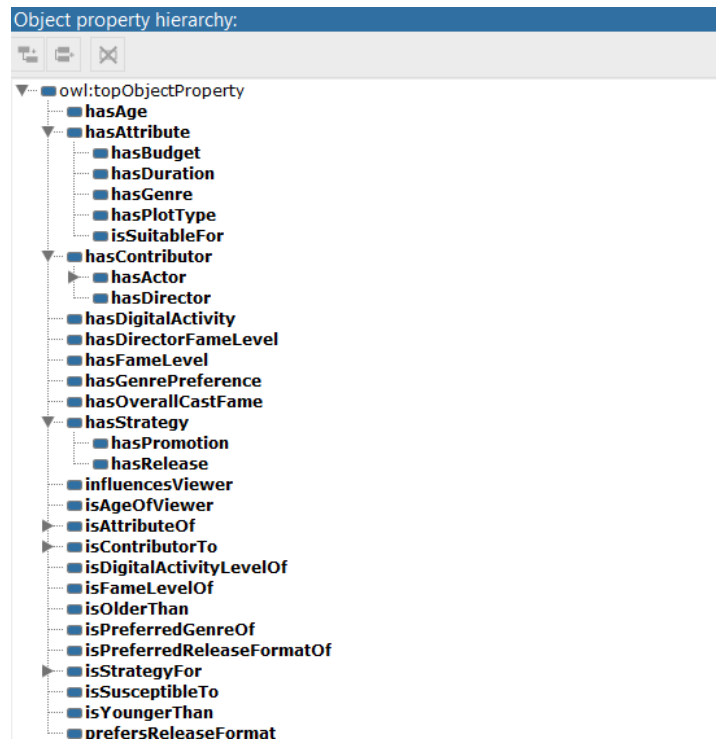


Рисунок 3.2 – Ієрархія властивостей об'єктів онтологічної бази знань

Для забезпечення цілісності даних та коректної роботи машини логічного виведення, об'єктним властивостям було призначено жорсткі характеристики.

1. Функціональність: математичний апарат модифікованого методу морфологічного аналізу вимагає, щоб кожен параметр конфігурації мав рівно одну альтернативу. Щоб гарантувати цю умову, ключові

властивості були визначені як функціональні. До них належать `hasBudget`, `hasDuration`, `hasPlotType`, `hasPromotion`, `hasRelease`, `isSuitableFor`, `hasAge`, `hasDigitalActivity`, `hasOverallCastFame` та `hasDirectorFameLevel`. Це унеможливлює логічні конфлікти під час роботи СППР.

2. Транзитивність: призначена для властивостей `isOlderThan` та `isYoungerThan`. Це дозволяє системі розуміти вікову ієрархію: якщо категорія `Adult` старша за `Teenager`, а `Teenager` старший за `Kid`, система автоматично будує логічний ланцюжок висновків.
3. Обернені властивості: кожна об'єктна властивість має свою логічну пару (наприклад, `hasBudget` – `isBudgetOf`, `hasGenre` – `isGenreOf`), що дозволяє підтримувати двосторонній зв'язок між об'єктами бази знань і здійснювати пошук у будь-якому напрямку.

Налаштовані характеристики та логічні обмеження для об'єктних властивостей зображено на рисунку 3.3.

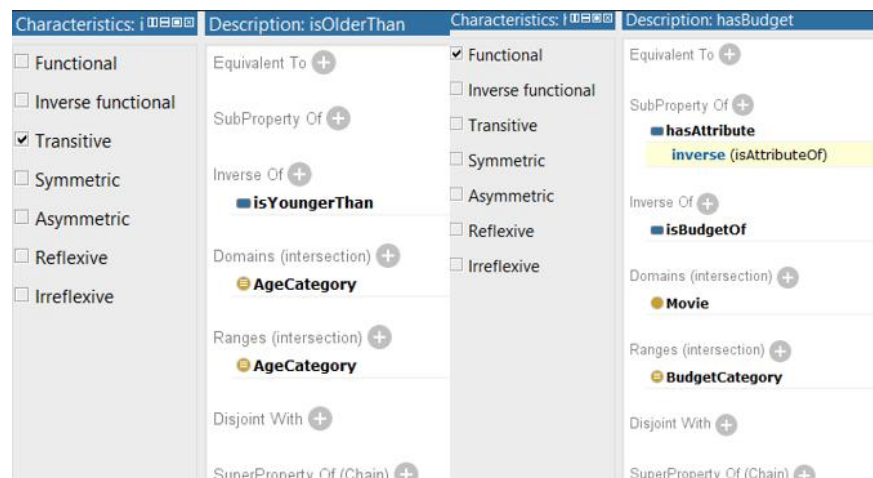


Рисунок 3.3 – Характеристики та логічні обмеження об'єктної властивості

Для формування альтернатив морфологічних таблиць параметричні класи були обмежені чітким переліком індивідів за допомогою конструкції `owl:oneOf`.

Наприклад, клас `BudgetCategory` еквівалентний виключно множині індивідів `{CategoryA, CategoryB, LowBudget}`. Подібним чином визначені

DigitalActivityLevel (HighDigitalActivity, LowDigitalActivity, MediumDigitalActivity), DurationCategory, PlotType (Adaptation, Original, Remake), FameLevel (Debut, Famous, RisingStar) та формати релізу (Hybrid, Streaming, Theatrical).

Приклад опису визначеного класу-переліку та його еквівалентностей у середовищі розробки наведено на рисунку 3.4.

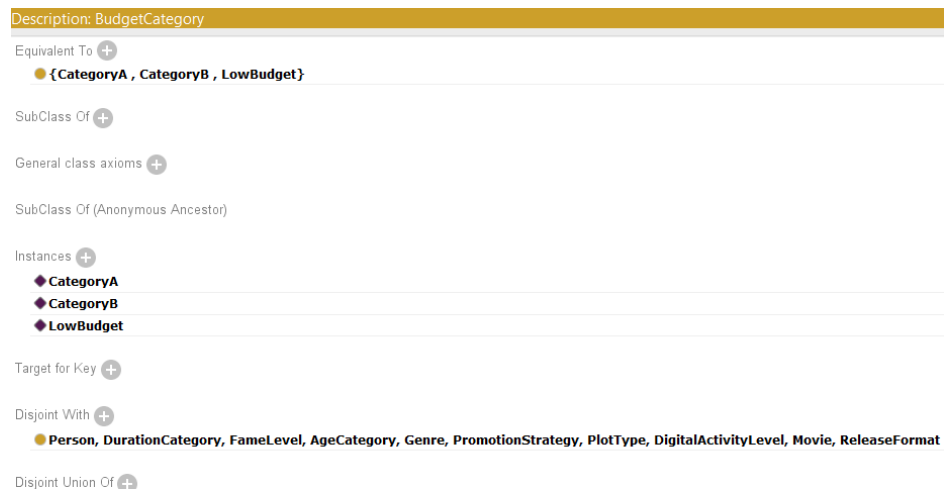


Рисунок 3.4 – Опис визначеного класу-переліку та його еквівалентностей

3.1.2 Автоматизоване наповнення онтології

Для забезпечення розробленої системи підтримки прийняття рішень репрезентативним обсягом даних було створено програмний модуль автоматизованого парсингу та наповнення бази знань. Модуль реалізовано мовою програмування Python із використанням бібліотеки pandas для обробки великих масивів даних та owlready2 для безпосередньої взаємодії з онтологією формату OWL.

Для отримання об'єктивних характеристик кінофільмів було використано п'ять відкритих некомерційних датасетів платформи IMDb (title.basics, title.ratings, title.crew, name.basics, title.principals). З метою оптимізації використання оперативної пам'яті під час обробки

наймасивнішого файлу акторського складу застосовано метод ітеративного читання даних блоками по 200 000 рядків.

Для забезпечення максимальної різноманітності конфігурацій під час подальшого морфологічного аналізу, алгоритм формує стратифіковану вибірку з 200 повнометражних фільмів у часовому проміжку з 1990 по 2026 роки. Фільми сортуються за кількістю оцінок глядачів та рівномірно розподіляються на три сегменти: найпопулярніші світові блокбастери, фільми середньої популярності та маловідомі нішеві стрічки.

Під час роботи скрипт зчитує відфільтровані дані, формує унікальні ідентифікатори об'єктів та автоматично визначає для них значення характеристик, що відповідають параметрам морфологічної таблиці. Заповнення ключових характеристик здійснюється за допомогою нижче наведених алгоритмів.

1. Бюджет: оскільки відкриті датасети IMDb не містять фінансової інформації, скрипт здійснює мережеві запити до API сервісу The Movie Database (TMDb). Отримані реальні цифри бюджету автоматично класифікуються відповідно до альтернатив: CategoryA (понад 40 млн дол.), CategoryB (від 2 до 40 млн дол.) та LowBudget (менше 2 млн дол.). У разі відсутності даних в API спрацьовує розроблена статистична евристика: фільми з понад 350 тис. голосів отримують CategoryA, з понад 40 тис. голосів – CategoryB, а решта класифікуються як LowBudget.
2. Рівень відомості: відомість творчої команди вираховується об'єктивно на основі аналізу найуспішніших робіт особи та максимальної кількості їхніх оцінок. Для відповідності параметрам морфологічної таблиці пороги розділено: режисери отримують статус Famous за наявності фільму з понад 200 тис. голосів, а інакше RisingStar, тоді як для акторів діє трирівнева шкала: Famous (> 400 тис. голосів), RisingStar (> 50 тис. голосів) та Debut для новачків або осіб з меншою кількістю оцінок.
3. Динамічне формування властивостей: задля підготовки даних до роботи алгоритму СППР, безпосередньо в середовищі owlready2 динамічно

створюються нові функціональні властивості об'єктів `FunctionalProperty`: `hasOverallCastFame` та `hasDirectorFameLevel`. Це дозволяє агрегувати рівень відомості всього акторського складу та режисера і прив'язати їх безпосередньо до індивіда фільму, що критично важливо для генерації однозначних конфігурацій морфологічної таблиці.

Результатом роботи програмного модуля є збережена копія вихідної онтології, яка містить повністю сформовану `AVox`-частину бази знань, а саме 200 екземплярів фільмів та пов'язаних з ними акторів і режисерів. Для кожного створеного фільму коректно призначено необхідні функціональні та екзистенціальні властивості згідно зі спроектованою структурою.

Фрагмент заповненої онтології зі згенерованими індивідами класів продемонстровано на рисунку 3.5, а результат автоматизованого наповнення властивостей для конкретного індивіда-фільму – на рисунку 3.6.



Рисунок 3.5. Результат автоматизованого заповнення онтології індивідами

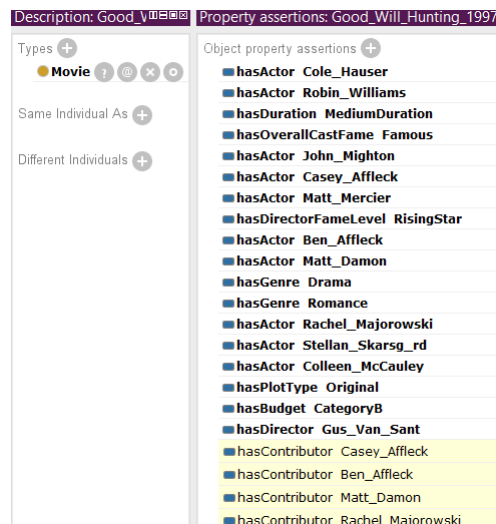


Рисунок 3.6. Результат автоматизованого наповнення властивостей індивіда фільму в онтології

3.2 Мережа морфологічних таблиць

Для забезпечення роботи алгоритму системи підтримки прийняття рішень з персоналізації маркетингових стратегій було спроектовано мережу морфологічних таблиць. В основу математичних розрахунків системи покладено двоетапний модифікований метод морфологічного аналізу з нерівномірним розподілом початкових оцінок. Головною метою побудови такої мережі є врахування впливу характеристик глядача на ймовірність вибору певних конфігурацій фільмів, а згодом – на очікувану результативність застосування конкретних маркетингових інструментів. Розроблена мережа складається з трьох взаємопов'язаних морфологічних таблиць: таблиці профілю користувача, таблиці характеристик фільму та таблиці маркетингових рішень.

Перша морфологічна таблиця (таблиця 3.1) описує профіль цільового користувача. Вона містить наступні характеристичні параметри: жанрові вподобання, вік, рівень цифрової активності та улюблена платформа перегляду.

Таблиця 3.1 – Морфологічна таблиця профілю користувача

Характеристичні параметри			
Жанрові вподобання	Вік	Цифрова активність	Платформа перегляду
1	2	3	4
1.1 Екшн	2.1 Дорослий (18+)	3.1 Висока	4.1 Кінотеатр
1.2 Комедія	2.2 Підліток (10-17)	3.2 Середня	4.2 Стримінг
1.3 Драма	2.3 Дитина (3-9)	3.3 Низька	4.3 Гібрид
1.4 Фантастика/Фентезі			
1.5 Хоррор			
1.6 Історичний			
1.7 Романтика			

Друга таблиця (таблиця 3.2) формалізує характеристики самого фільму. Вона складається з таких параметрів: жанр, рівень відомості режисера, хронометраж, бюджетна категорія, тип сюжету та загальний рівень відомості акторського складу.

Таблиця 3.2 – Морфологічна таблиця характеристик фільму

Характеристичні параметри					
Жанр	Режисер	Хрон.	Бюджет	Сюжет	Актори
1	2	3	4	5	6
1.1 Екшн	2.1 Відомий	3.1 Короткий	4.1 Великий	5.1 Ориг.	6.1 Відомі
1.2 Комедія	2.2 Маловід.	3.2 Середній	4.2 Середній	5.2 Адаптація	6.2 Наб. поп.
1.3 Драма		3.3 Довгий	4.3 Малий	5.3 Ремейк	6.3 Дебют
1.4 Фантаст./Фентезі					
1.5 Хоррор					
1.6 Історія					
1.7 Романтика					

Третя таблиця (таблиця 3.3) містить альтернативи рішень щодо маркетингової кампанії та релізу, які ухвалює дистриб'ютор. Її параметрами є: офлайн інструменти, онлайн інструменти, нестандартні інструменти просування та формат релізу.

Таблиця 3.3 – Морфологічна таблиця маркетингових стратегій та релізу

Рішення				
Реклама	Офлайн інструменти	Онлайн інструменти	Нестандартні інструменти	Реліз
7	8	9	10	11
7.1 Офлайн	8.1 Традиційна реклама	9.1 SMM	10.1 Influencer-маркетинг	11.1 Кіно-реліз
7.2 Онлайн	8.2 PR-заходи	9.2 Контент-маркетинг	10.2 Партизанський маркетинг	11.2 Онлайн-ексклюзив
7.3 Нестандартна		9.3 Цифрова реклама	10.3 Гейміфікація	11.3 Гібридний

Ідея функціонування СППР розбита на три послідовні етапи роботи алгоритму, де створена онтологія відіграє роль фундаментальної бази знань. Для врахування зв'язків між параметрами морфологічних таблиць застосовуються числові матриці взаємозв'язків. Відповідно до методології модифікованого методу морфологічного аналізу, кожній парі альтернатив різних параметрів присвоюється експертна оцінка в діапазоні від -1 до 1.

Водночас, не всі параметри мають прямий причинно-наслідковий вплив один на одного. Для пар альтернатив, між якими логічний зв'язок очевидно відсутній, експертна оцінка в матрицях приймалася рівною нулю.

Першим етапом є адаптація початкових оцінок від «Користувача» до «Фільму». На цьому кроці система бере профіль конкретного глядача з таблиці 3.1 як вхідний сценарій. За допомогою матриці зв'язків між глядачем та фільмом, алгоритм автоматично перераховує початкові оцінки: математично підвищує ймовірності для тих характеристик фільму, що підходять цьому

користувачу, та занижує для тих, що йому не цікаві. Матрицю зв'язків між профілем користувача та характеристиками фільму наведено на рисунку 3.7.

Користувач	Ф1.1	Ф1.2	Ф1.3	Ф1.4	Ф1.5	Ф1.6	Ф1.7	Ф2.1	Ф2.2	Ф3.1	Ф3.2	Ф3.3	Ф4.1	Ф4.2	Ф4.3	Ф5.1	Ф5.2	Ф5.3	Ф6.1	Ф6.2	Ф6.3
K1.1	1	-0.3	-0.3	0.4	-0.1	0.2	-0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K1.2	-0.3	1	0.1	0.2	-0.2	0	0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K1.3	-0.3	0.1	1	-0.2	-0.3	0.5	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K1.4	0.4	0.2	-0.2	1	0.2	0	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K1.5	-0.1	-0.3	-0.3	0.2	1	0	-0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K1.6	0.2	0	0.5	-0.1	0	1	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K1.7	-0.5	0.4	0.6	0.1	-0.5	0.3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K2.1	0.2	0.2	0.7	0.1	0.2	0.6	0.4	0.5	-0.2	0	0.4	0.5	0.4	0.3	-0.2	0.5	0.3	0.2	0.5	0.3	0.1
K2.2	0.7	0.4	0.1	0.7	0.5	-0.2	0.1	-0.2	0.4	0.3	0.5	-0.4	0.6	0.3	-0.4	0.3	0.4	0.5	0.4	0.6	-0.2
K2.3	-0.3	0.7	-0.5	0.5	-0.9	-0.5	0.1	-0.4	0.1	0.8	0.3	-0.8	0.3	0.6	-0.5	0.6	0.5	0.2	0.1	0.2	0.5
K3.1	0.4	0.2	0.1	0.6	0.4	-0.1	0.2	0.2	0.2	0.2	0.5	0.4	0.5	0.4	0.1	0.4	0.5	0.5	0.4	0.5	0.2
K3.2	0.3	0.4	0.4	0.3	0.2	0.3	0.4	0.4	0.2	0.4	0.6	0.3	0.4	0.5	0.3	0.5	0.4	0.3	0.5	0.4	0.2
K3.3	0.1	0.3	0.7	0.1	0.1	0.6	0.5	0.6	-0.2	0.1	0.4	0.6	0.3	0.4	0.2	0.6	0.2	0.1	0.6	0.2	0.1
K4.1	0.8	0.3	0.2	0.8	0.5	0.2	0.3	0.6	-0.3	0.1	0.4	0.7	0.8	0.3	-0.5	0.5	0.4	0.3	0.7	0.2	0.1
K4.2	0.2	0.5	0.4	0.3	0.4	0.4	0.5	-0.2	0.6	0.5	0.6	0.1	0.2	0.7	0.5	0.4	0.5	0.6	0.2	0.7	0.4
K4.3	0.5	0.4	0.3	0.6	0.4	0.3	0.4	0.4	0.3	0.3	0.5	0.4	0.5	0.5	0.2	0.5	0.4	0.4	0.5	0.4	0.2

Рисунок 3.7 – Матриця зв'язків між профілем користувача та характеристиками фільму

Другим етапом є аналіз внутрішніх зв'язків морфологічної таблиці «Фільм». СППР вирішує внутрішні конфлікти самої кіноіндустрії, перевіряючи, наскільки згенерований на першому етапі конструкт фільму є реалістичним. Ця матриця має вигляд напівматриці, оскільки внутрішні параметри однаково впливають один на одного, що продемонстровано на рисунку 3.8.

Фільм	Ф1.1	Ф1.2	Ф1.3	Ф1.4	Ф1.5	Ф1.6	Ф1.7	Ф2.1	Ф2.2	Ф3.1	Ф3.2	Ф3.3	Ф4.1	Ф4.2	Ф4.3	Ф5.1	Ф5.2	Ф5.3
Ф2.1	0.4	0.2	-0.1	0.4	-0.2	0.5	0.3											
Ф2.2	0.2	0.3	0.3	-0.3	0.3	0.1	0.5											
Ф3.1	0.2	0.5	0.25	-0.2	0.5	0.1	0.4	0.1	0.6									
Ф3.2	0.5	0.2	0.4	0.3	0.2	0.5	0.2	0.5	0.5									
Ф3.3	0.1	-0.5	-1	0.4	-0.1	0.2	-0.3	0.6	-0.5									
Ф4.1	0.8	-0.3	0.1	0.8	-0.5	0.6	-0.2	0.8	-0.8	-0.6	0.3	0.8						
Ф4.2	0.3	0.4	0.5	0.3	0.3	0.4	0.5	0.3	0.4	0.2	0.6	0.3						
Ф4.3	-0.7	0.3	0.4	-0.7	0.7	0	0.4	-0.6	0.7	0.6	0.2	-0.6						
Ф5.1	0.5	0.4	0.3	0.15	0.3	0.5	0.2	0.2	0.5	0.1	0.5	0.2	0	0	0			
Ф5.2	0.3	-0.2	-0.1	0.7	-0.3	0.3	0.4	0.4	0.2	0	0.5	0.4	0.6	0.3	-0.5			
Ф5.3	0.4	-0.2	-0.3	0.3	0.3	0.2	-0.2	0.5	0.1	0.2	0.4	0.3	0.4	0.4	-0.2			
Ф6.1	0.5	0.3	0.1	0.2	-0.3	0.7	-0.1	0.8	-0.5	-0.2	0.5	0.4	0.8	0.4	-0.8	0.3	0.4	0.5
Ф6.2	0.4	0.3	0.3	0.1	0.2	0.4	0.2	0.2	0.5	0.2	0.6	0.2	0.2	0.7	0.2	0.5	0.3	0.3
Ф6.3	0.1	0.5	0.3	-0.2	0.4	-0.3	0.6	-0.7	0.8	0.6	0.2	-0.5	-0.8	0.2	0.8	0.4	0.1	0.1

Рисунок 3.8 – Матриця внутрішньої узгодженості характеристик фільму

Третім етапом відбувається безпосередній розрахунок найкращих стратегій просування та релізу. Це фінальний розрахунок, метою якого є визначення очікуваної результативності кожної альтернативи з МТ рішень (таблиця 3.3). Система використовує найбільш імовірні фільми з другого етапу і пропускає їх через матрицю зв'язків маркетингових інструментів, яку зображено на рисунку 3.9.

Фільм	C7.1	C7.2	C7.3	C8.1	C8.2	C9.1	C9.2	C9.3	C10.1	C10.2	C10.3	C11.1	C11.2	C11.3
Ф1.1	0.5	0.7	0.4	0.4	0.5	0.6	0.4	0.6	0.3	0.4	0.5	0.8	-0.2	0.4
Ф1.2	0.3	0.6	0.3	0.3	0.3	0.7	0.5	0.6	0.6	0.5	0.2	0.3	0.6	0.4
Ф1.3	0.4	0.4	0.1	0.5	0.4	0.4	0.5	0.3	0.2	0.2	0	0.5	0.4	0.4
Ф1.4	0.5	0.7	0.6	0.4	0.5	0.7	0.5	0.6	0.5	0.6	0.7	0.8	-0.1	0.5
Ф1.5	0.2	0.6	0.4	0.2	0.3	0.6	0.4	0.5	0.5	0.6	0.3	0.4	0.5	0.4
Ф1.6	0.6	0.3	0.1	0.7	0.6	0.3	0.5	0.3	0.1	0.1	0	0.6	0.2	0.4
Ф1.7	0.4	0.5	0.2	0.4	0.4	0.6	0.5	0.5	0.4	0.3	0.1	0.4	0.5	0.4
Ф2.1	0.6	0.5	0.3	0.7	0.7	0.5	0.6	0.5	0.4	0.3	0.2	0.7	0.1	0.5
Ф2.2	0.2	0.6	0.4	0.2	0.3	0.7	0.5	0.6	0.6	0.5	0.3	0.2	0.7	0.4
Ф3.1	0.2	0.6	0.4	0.2	0.2	0.7	0.4	0.6	0.6	0.5	0.4	0.2	0.7	0.4
Ф3.2	0.5	0.5	0.3	0.5	0.5	0.5	0.5	0.5	0.4	0.4	0.3	0.6	0.3	0.5
Ф3.3	0.6	0.3	0.2	0.7	0.6	0.3	0.4	0.3	0.2	0.2	0.1	0.7	0.1	0.4
Ф4.1	0.7	0.6	0.5	0.8	0.7	0.6	0.5	0.7	0.5	0.6	0.6	0.9	-0.3	0.5
Ф4.2	0.4	0.5	0.3	0.5	0.5	0.6	0.5	0.5	0.5	0.4	0.3	0.5	0.4	0.6
Ф4.3	0.1	0.6	0.3	0.1	0.2	0.8	0.6	0.6	0.7	0.6	0.2	0.1	0.8	0.3
Ф5.1	0.4	0.6	0.4	0.4	0.5	0.6	0.6	0.6	0.5	0.5	0.3	0.4	0.5	0.5
Ф5.2	0.5	0.5	0.3	0.6	0.5	0.5	0.5	0.5	0.4	0.3	0.4	0.6	0.3	0.5
Ф5.3	0.6	0.4	0.2	0.7	0.6	0.4	0.4	0.4	0.3	0.2	0.2	0.7	0.2	0.4
Ф6.1	0.6	0.5	0.2	0.7	0.7	0.6	0.5	0.5	0.4	0.3	0.2	0.7	0.1	0.5
Ф6.2	0.3	0.6	0.4	0.3	0.4	0.7	0.6	0.6	0.7	0.5	0.3	0.4	0.6	0.5
Ф6.3	0.1	0.5	0.3	0.1	0.2	0.7	0.5	0.5	0.6	0.6	0.2	0.1	0.8	0.3

Рисунок 3.9 – Матриця зв'язків між характеристиками фільму та маркетинговими стратегіями

Для фінального коригування результатів додатково використовуються параметри самого користувача з першого етапу. Наприклад, для блокбастера і глядача з високою цифровою активністю алгоритм виділить найвищі бали для цифрової реклами та гібридного релізу. Відповідну матрицю зв'язків між маркетинговими стратегіями та профілем користувача подано на рисунку 3.10.

Стратегія	K1.1	K1.2	K1.3	K1.4	K1.5	K1.6	K1.7	K2.1	K2.2	K2.3	K3.1	K3.2	K3.3	K4.1	K4.2	K4.3
C7.1	0.4	0.2	0.6	0.3	0.1	0.6	0.5	0.8	-0.2	-0.5	-0.6	0.2	0.8	0.6	-0.2	0.2
C7.2	0.5	0.5	0.3	0.7	0.6	0.2	0.4	0.2	0.8	0.6	0.9	0.5	-0.3	0.3	0.8	0.5
C7.3	0.6	0.4	0.1	0.6	0.5	0.1	0.2	0.1	0.7	0.4	0.7	0.3	-0.2	0.4	0.5	0.5
C8.1	0.3	0.2	0.5	0.2	0.1	0.6	0.4	0.8	-0.2	-0.4	-0.6	0.2	0.8	0.7	-0.3	0.2
C8.2	0.4	0.3	0.4	0.3	0.2	0.5	0.5	0.6	0.1	0	-0.2	0.4	0.6	0.5	0.1	0.3
C9.1	0.4	0.6	0.2	0.6	0.5	0.1	0.3	0.1	0.9	0.5	0.9	0.4	-0.5	0.2	0.8	0.5
C9.2	0.3	0.4	0.4	0.5	0.4	0.3	0.4	0.4	0.6	0.4	0.6	0.6	0.2	0.3	0.6	0.4
C9.3	0.5	0.5	0.3	0.7	0.6	0.2	0.4	0.3	0.8	0.7	0.8	0.5	-0.2	0.4	0.7	0.6
C10.1	0.4	0.5	0.1	0.5	0.4	0.1	0.3	0.1	0.8	0.6	0.8	0.4	-0.4	0.3	0.7	0.5
C10.2	0.3	0.4	0.2	0.4	0.4	0.2	0.3	0.2	0.6	0.3	0.6	0.5	-0.1	0.4	0.5	0.4
C10.3	0.5	0.3	-0.2	0.7	0.5	-0.2	0.1	-0.1	0.8	0.7	0.9	0.2	-0.6	0.2	0.7	0.4
C11.1	0.8	0.4	0.5	0.7	0.6	0.4	0.4	0.6	0.3	-0.2	-0.2	0.3	0.6	0.9	-0.8	0.1
C11.2	-0.3	0.6	0.4	0.4	0.4	0.3	0.6	-0.1	0.8	0.6	0.9	0.4	-0.4	-0.8	0.9	0.2
C11.3	0.4	0.5	0.4	0.6	0.4	0.4	0.4	0.4	0.6	0.4	0.5	0.6	0.2	0.1	0.2	0.9

Рисунок 3.10 – Матриця зв'язків між маркетинговими стратегіями та профілем користувача

Таким чином, розроблена мережа морфологічних таблиць дозволяє усунути проблему статичних рішень і забезпечує гнучкий, математично обґрунтований підхід до персоналізації рекламних кампаній у медіаіндустрії.

3.3 Проєктування та реалізація програмного інтерфейсу

Під час проєктування програмного інтерфейсу головною задачею було створення зручного та інтуїтивно зрозумілого середовища для особи, що приймає рішення. Інтерфейс програмного продукту реалізовано за допомогою веб-фреймворку Streamlit на мові Python. Використання цієї технології дозволило створити інтерактивний веб-додаток, який забезпечує швидкий доступ до аналітики через браузер.

Структурно програмний інтерфейс розділено на дві основні зони: бокову панель налаштувань для введення вхідних даних та головне вікно для виведення результатів.

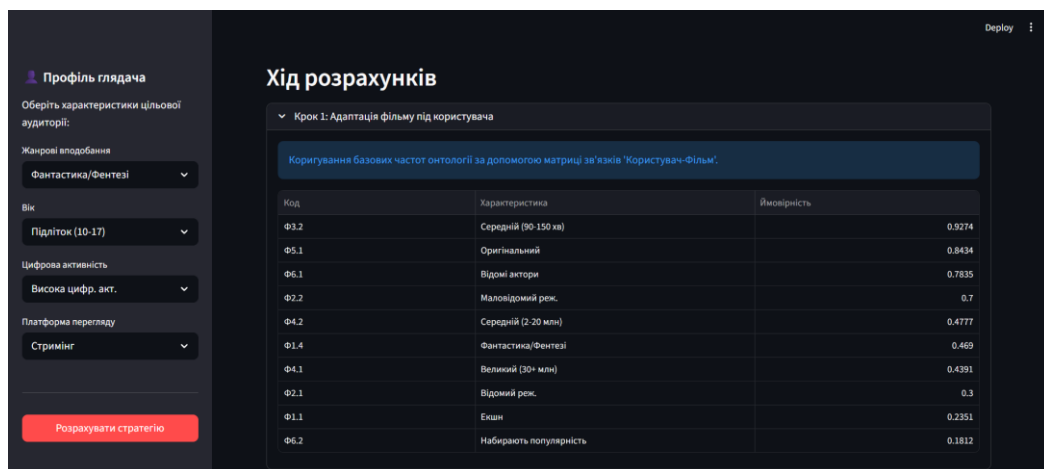
У боковій панелі реалізовано меню «Профіль глядача», яке можна побачити на рисунку 3.11. Введення даних здійснюється за допомогою випадających списків, де користувач обирає характеристики цільової аудиторії за чотирма критеріями: жанрові вподобання, вік, рівень цифрової активності та платформа перегляду. Використання випадających списків із заздалегідь визначеними альтернативами гарантує коректність вхідних даних, оскільки користувач може обрати лише існуючі варіанти. Це унеможливорює введення помилкових чи порожніх значень, що є важливою складовою стабільності системи.

Рисунок 3.11 – Бокова панель для введення параметрів профілю користувача та вікно інтерфейсу

У головному вікні програми до моменту початку розрахунків відображається інформаційне повідомлення-підказка з проханням обрати характеристики глядача у меню зліва та натиснути кнопку розрахунку. Це забезпечує зрозумілу навігацію для нових користувачів.

Після натискання головної кнопки «Розрахувати стратегію», яка розміщена внизу бокової панелі, додаток звертається до математичного ядра системи. Завдяки оптимізації алгоритмів МММА розрахунки виконуються швидко, після чого у головному вікні програми послідовно відображаються три блоки результатів, що відповідають логічним етапам роботи алгоритму.

Спочатку система виводить результати першого розрахункового етапу – скориговані ймовірності альтернатив морфологічної таблиці «Фільм», які видно на рисунку 3.12. Це дозволяє особі, що приймає рішення, наочно побачити, як саме змінилися базові оцінки характеристик медіапродукту під впливом обраного профілю глядача.



Хід розрахунків

Крок 1: Адаптація фільму під користувача

Коригування базових частот онтології за допомогою матриці зв'язків 'Користувач-Фільм'.

Код	Характеристика	Ймовірність
Ф3.2	Середній (90-150 мн)	0.9274
Ф5.1	Оригінальний	0.8434
Ф6.1	Відомі актори	0.7835
Ф2.2	Маловідомий реж.	0.7
Ф4.2	Середній (2-20 млн)	0.4777
Ф1.4	Фантастика/Фентезі	0.469
Ф4.1	Великий (30+ млн)	0.4391
Ф2.1	Відомий реж.	0.3
Ф1.1	Екшн	0.2351
Ф6.2	Набирають популярність	0.1812

Рисунок 3.12 – Відображення скоригованих ймовірностей характеристик фільму

Нижче програма демонструє результати другого етапу. На екран виводиться інформаційне повідомлення про загальну кількість знайдених реалістичних конфігурацій, а також список із топ-5 найбільш імовірних конфігурацій фільмів, що продемонстровано на рисунку 3.13. Цей блок слугує

індикатором того, що алгоритм успішно відсіяв індустріально несумісні варіанти за допомогою матриці внутрішньої узгодженості.

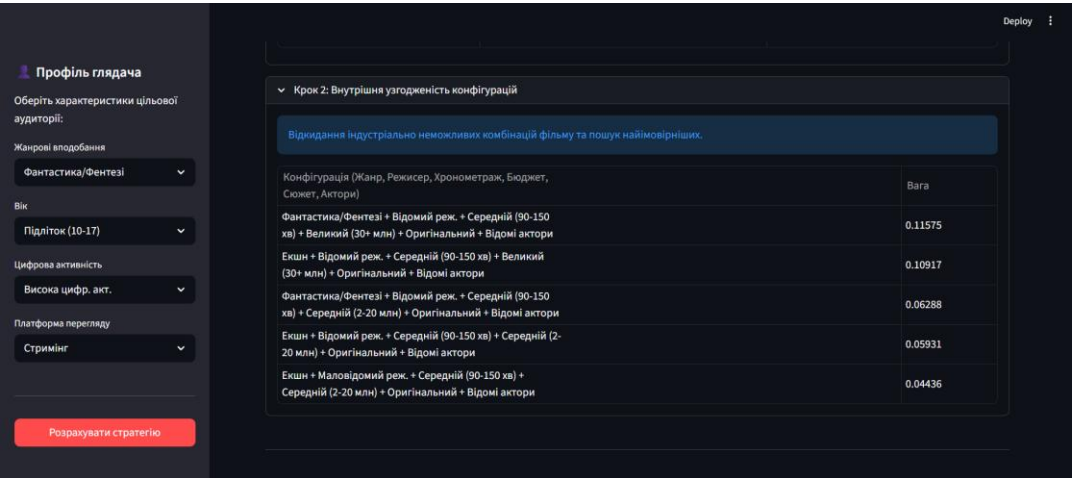


Рисунок 3.13 – Відображення кількості відфільтрованих конфігурацій та найімовірніших фільмів

Нарешті, як підсумок роботи системи, у головному вікні генерується фінальний результат третього етапу – рейтинг очікуваної результативності маркетингових стратегій на рисунку 3.14. Дані подаються бар чартів, де найбільш ефективні інструменти просування та формати релізу для заданого користувача мають найбільше значення.

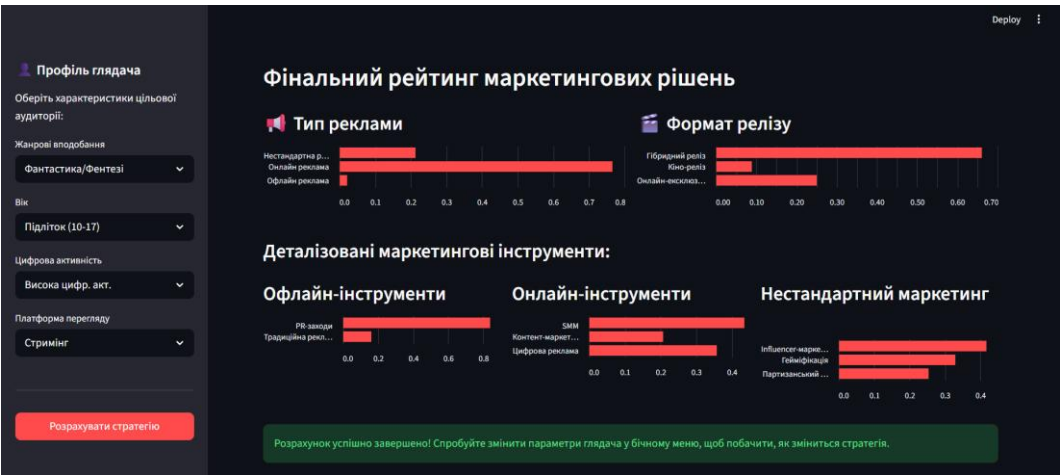


Рисунок 3.14 – Фінальний рейтинг маркетингових стратегій у головному вікні програми

Такий підхід до поетапного виведення інформації робить процес прийняття рішень прозорим, оскільки користувач бачить не лише кінцеву рекомендацію, але й проміжні логічні кроки системи.

3.4 Програмна реалізація основного алгоритму та аналіз проміжних результатів

Для демонстрації покрокової роботи розробленої СППР та аналізу її математичного ядра, змодельємо ситуацію для глядача з наступним цільовим профілем:

- 1) жанрові вподобання: Екшн;
- 2) вікова категорія: Дорослий (18+ років);
- 3) цифрова активність: Висока;
- 4) платформа перегляду: Кінотеатр.

Запуск алгоритму ініціює послідовне виконання розрахункових етапів, кожен з яких супроводжується виведенням відповідних проміжних результатів на екран.

На початку роботи програма не використовує стандартний для класичного методу рівномірний розподіл початкових оцінок. Замість цього функція парсингу автоматично зчитує OWL-файл розробленої онтології, що містить екземпляри 200 реальних кінофільмів. Програма підраховує частоти появи кожної альтернативи і формує об'єктивний вектор базових ймовірностей для кожної з 6 характеристик фільму.

Таблиця 3.4 – Ймовірності характеристик фільму на основі даних з онтології

Жанр		Режисер		Хронометраж		Бюджет		Сюжет		Актор	
Е	0.235	В	0.6	К	0.05	В	0.39	О	0.875	В	0.825
К	0.16	М	0.4	С	0.825	С	0.395	А	0.080	Н	0.11
Д	0.04			Д	0.125	М	0.215	Р	0.065	Д	0.065
Ф	0.265										
Х	0.08										
І	0.1										
Р	0.12										

На наступному етапі базові оцінки з онтології коригуються під заданого дорослого глядача за допомогою матриці «Користувач – Фільм». Алгоритм перераховує початкові оцінки: він збільшує ймовірності для тих характеристик фільму, що підходять цьому глядачу, та зменшує для тих, що йому не цікаві, використовуючи формулу умовної результативності. Наприклад, оскільки обрано жанр «Екшн» та вік «Дорослий», математично суттєво зростає ймовірність жанрів екшн та драма, довгого хронометражу та середнього чи великого бюджету, тоді як ймовірність дитячих чи суто романтичних форматів знижується. Отримані проміжні результати виводяться у першому блоці головного вікна програми.

Таблиця 3.5 – Скориговані ймовірності характеристик фільму з врахуванням профілю користувача

Жанр		Режисер		Хронометраж		Бюджет		Сюжет		Актор	
Е	0.419	В	0.865	К	0.022	В	0.589	О	0.895	В	0.893
К	0.062	М	0.135	С	0.826	С	0.373	А	0.071	Н	0.078
Д	0.019			Д	0.152	М	0.038	Р	0.034	Д	0.029
Ф	0.347										
Х	0.054										
І	0.061										
Р	0.039										

Отримані на попередньому етапі оцінки проходять перевірку через матрицю внутрішніх взаємозв'язків у межах таблиці «Фільм». Програма генерує абсолютно всі можливі комбінації характеристик фільмів, загальна кількість яких у повному морфологічному просторі становить 2268 варіантів. Якщо матриця вказує на повну індустріальну несумісність між параметрами, добуток для цієї конфігурації автоматично занулюється, і вона відкидається як неможлива. В результаті відсіювання система виводить інформаційне повідомлення про загальну кількість знайдених реалістичних конфігурацій фільмів з ймовірністю більшою за нуль, та демонструє топ-5 найбільш імовірних з них, що проілюстровано на рисунку 3.15.

Крок 2: Внутрішня узгодженість конфігурацій	
Відкидання індустріально неможливих комбінацій фільму та пошук найімовірніших.	
Конфігурація (Жанр, Режисер, Хронометраж, Бюджет, Сюжет, Актори)	Вартість
Екшн + Відомий реж. + Середній (90-150 хв) + Великий (30+ млн) + Оригінальний + Відомі актори	0.29652
Фантастика/Фентезі + Відомий реж. + Середній (90-150 хв) + Великий (30+ млн) + Оригінальний + Відомі актори	0.13034
Екшн + Відомий реж. + Середній (90-150 хв) + Середній (2-20 млн) + Оригінальний + Відомі актори	0.09386
Історичний + Відомий реж. + Середній (90-150 хв) + Великий (30+ млн) + Оригінальний + Відомі актори	0.04669
Екшн + Відомий реж. + Довгий (150+ хв) + Великий (30+ млн) + Оригінальний + Відомі актори	0.04412

Рисунок 3.15 – Відображення кількості відфільтрованих конфігурацій та найімовірніших збірок фільму

На фінальному кроці відбувається розрахунок найкращих стратегій просування та релізу, виведені на рисунку 3.16. Для кожної з реалістичних конфігурацій фільму програма перевіряє ефективність стратегії через матрицю «Фільм – Стратегія». Щоб зробити результат максимально точним, цей бал додатково коригується двома поправками.

1. Враховується безпосередній вплив профілю глядача через матрицю «Стратегія – Користувач». Оскільки наш глядач віддає перевагу кінотеатрам, альтернатива «Кіно-реліз» отримує найвищий стимулюючий бал, а онлайн-ексклюзив навпаки жорстко знижується. Водночас, висока цифрова активність паралельно підвищує бали для цифрової реклами та SMM.
2. Застосовується ієрархічне зважування, а саме ефективність конкретного інструменту домножується на вагу його батьківської категорії.



Рисунок 3.16 – Фінальний рейтинг маркетингових стратегій у головному вікні програми

Для більш детального аналізу зведемо головні результати розрахунків очікуваної результативності, згенеровані системою, у таблицю 3.6.

Таблиця 3.6 – Значення очікуваної результативності стратегій

Група рішень	Маркетинговий інструмент	Очікувана результативність
Тип реклами	Онлайн реклама	0,57
	Нестандартна реклама	0,23
	Офлайн реклама	0,20
Формат релізу	Кіно	0,72
	Гібрид	0,28
	Онлайн	0,00
Офлайн-інструменти	PR-заходи	0,65
	Традиційна реклама	0,35
Онлайн-інструменти	Цифрова реклама	0,42
	SMM	0,38
	Контент-маркетинг	0,20
Нестандартний маркетинг	Influencer	0,40
	Партизанський	0,37
	Гейміфікація	0,23

Отримані результати демонструють високу точність та адаптивність математичної моделі. Для дорослого глядача, який віддає перевагу кінотеатрам, алгоритм закономірно виділив беззаперечним лідером формат «Кіно-реліз», залишивши гібридний формат на другому місці, а онлайн-ексклюзив отримав найнижчі показники.

Що стосується глобальних маркетингових стратегій, система розставила пріоритети наступним чином: найбільший показник очікуваної результативності отримала «Онлайн реклама», на другому місці опинилася «Нестандартна реклама», а «Офлайн реклама» показала найнижчий результат. Деталізований розгляд маркетингових інструментів підтверджує цю логіку.

Для даного профілю найкраще спрацюють цифрова реклама та SMM, PR-заходи та Influencer-маркетинг.

Отже, результати розрахунків СППР аналітично підтверджують дієвість вибору рішень. Замість покладання лише на інтуїцію, система математично доводить, що для дорослого глядача з високою цифровою активністю, який любить кінотеатри, оптимальним є фокус на цифрові інструменти комунікації та PR-заходи з ексклюзивним кінотеатральним релізом.

3.5 Висновки до розділу 3

У даному розділі було здійснено практичну реалізацію розробленої системи підтримки прийняття рішень для персоналізації маркетингових стратегій у медіаіндустрії.

Створено онтологічну базу знань, структуру якої автоматизовано наповнено репрезентативною вибіркою з 200 реальних кінофільмів, що дозволило сформулювати об'єктивні базові частоти для подальших обчислень. На основі спроектованої мережі з трьох взаємопов'язаних морфологічних таблиць програмно реалізовано алгоритм двоетапного модифікованого методу морфологічного аналізу з нерівномірним розподілом початкових оцінок.

Для забезпечення зручної взаємодії з особою, яка приймає рішення, розроблено мінімалістичний та інтуїтивно зрозумілий програмний інтерфейс. Покроковий аналіз проміжних та фінальних результатів для тестового профілю глядача підтвердив коректність роботи математичного ядра: система успішно адаптує базові оцінки під користувача, ефективно відсіює індустріально неможливі конфігурації фільмів та формує логічно обґрунтований рейтинг результативності стратегій просування і форматів релізу.

Отже, розроблений програмний продукт повністю відповідає поставленим завданням та доводить дієвість поєднання онтологічного моделювання з методами морфологічного аналізу.

РОЗДІЛ 4 ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ

У даному розділі проводиться техніко-економічне обґрунтування розробки програмної системи підтримки прийняття рішень для персоналізації маркетингових стратегій у медіаіндустрії. Програмний продукт складається з трьох взаємопов'язаних модулів: модуля автоматизованого наповнення онтологічної бази знань, математичного ядра СППР на основі двоетапного модифікованого методу морфологічного аналізу та веб-інтерфейсу користувача. Для вибору оптимального варіанту реалізації застосовується апарат функціонально-вартісного аналізу (ФВА).

Функціонально-вартісний аналіз – технологія, що дозволяє оцінити реальну вартість продукту незалежно від організаційної структури компанії. ФВА проводиться з метою виявлення резервів зниження витрат за рахунок ефективніших варіантів розробки та кращого співвідношення між споживчою вартістю виробу й витратами на його виготовлення.

4.1 Постановка задачі проектування

Метою даного етапу є створення програмної системи підтримки прийняття рішень, яка автоматизує процес вибору маркетингових стратегій і форматів релізу кінофільмів на основі профілю цільового глядача. Продукт повинен забезпечувати можливість формалізованого представлення знань кіноіндустрії у вигляді онтологічної бази знань, а також реалізацію двоетапного модифікованого методу морфологічного аналізу для генерації персоналізованих рекомендацій.

Технічні вимоги до програмного продукту є наступні:

- 1) наявність онтологічної бази знань з реальними даними про кінофільми;
- 2) реалізація двоетапного МММА з нерівномірним розподілом початкових оцінок;

- 3) коректна обробка матриць взаємної узгодженості між параметрами;
- 4) формування рейтингу маркетингових стратегій на виході системи;
- 5) інтерактивний інтерфейс для введення профілю глядача та перегляду результатів;
- 6) використання відкритих даних IMDb/TMDB для наповнення бази знань;
- 7) можливість розширення та модернізації системи в майбутньому.

4.2 Обґрунтування функцій програмного продукту

Головна функція F_0 – розробка СППР для персоналізації маркетингових стратегій у медіаіндустрії на основі онтологічного моделювання та мережі морфологічних таблиць. На основі F_0 виділено три основні підфункції.

F_1 – побудова та наповнення бази знань:

- 1) (а) автоматизоване наповнення OWL-онтології (owlready2, IMDb, TMDB API);
- 2) (б) ручне формування реляційної бази даних SQLite/PostgreSQL.

F_2 – реалізація математичного алгоритму рекомендацій:

- 1) (а) двоетапний МММА з нерівномірним розподілом початкових оцінок;
- 2) (б) алгоритм простого адитивного зважування (SAW).

F_3 – реалізація інтерфейсу користувача:

- 1) (а) streamlit – інтерактивний веб-додаток;
- 2) (б) flask та HTML-шаблони – класичний веб-застосунок.

Варіанти реалізації відображено у морфологічній карті (рис. 4.1).

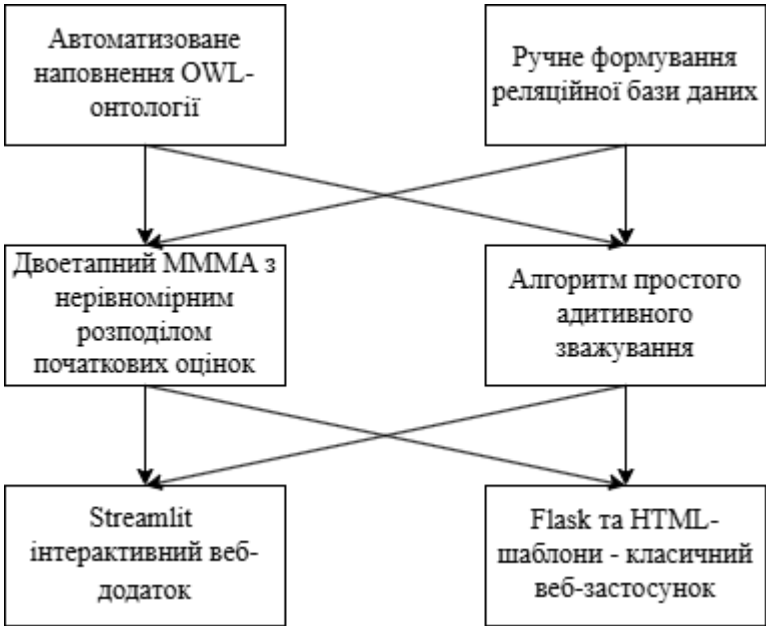


Рисунок 4.1 – Морфологічна карта

Дана морфологічна карта відображає всі можливі комбінації варіантів для наведених основних функцій. На основі даної карти будується позитивно-негативна матриця, яка наведена в таблиці 4.1.

Таблиця 4.1 – Позитивно-негативна матриця варіантів реалізації функцій

Функція	Варіант	Переваги	Недоліки
F1	А	Автоматичне наповнення 200 реальних фільмів, підтримка логічного виведення OWL, OWA для неповних даних, розширюваність онтології.	Тривалий час первинного наповнення через API-запити, складність налагодження owlready2.
	Б	Проста схема даних, висока швидкість SQL-запитів, легка інтеграція з Python.	Відсутній механізм логічного виведення, немає підтримки семантичних зв'язків між сутностями, ручне наповнення.

Продовження таблиці 4.1

F2	A	Враховує взаємозалежності параметрів через матриці узгодженості, відсіює нереалістичні конфігурації, коректно обробляє нерівномірні початкові оцінки з онтології.	Вищий час розробки, потребує налаштування чотирьох матриць зв'язків.
	B	Проста реалізація, мінімум параметрів налаштування.	Не враховує взаємозалежності параметрів, не відсіює нереалістичні комбінації, неможливо обробити нерівномірні початкові оцінки.
F3	A	Мінімальний час розробки інтерфейсу, вбудована підтримка діаграм і таблиць, нативна інтеграція з pandas/numpy без HTTP-шаблонів.	Обмежена гнучкість дизайну порівняно з Flask та HTML.
	B	Повна свобода проектування UI, широкі можливості маршрутизації.	Значно більший обсяг коду для інтерфейсу, потребує розробки HTML-шаблонів та CSS окремо.

На основі аналізу позитивно-негативної матриці зроблено вибір варіантів.

Функція F_1 : Обрано варіант A – OWL-онтологія. Автоматизоване наповнення через IMDb та TMDb API з підтримкою логічного виведення та принципу відкритого світу є ключовою вимогою алгоритму МММА, який потребує об'єктивних базових частот, обчислених на реальних даних.

Функція F_2 : Обрано варіант A – двоетапний МММА. Метод є принципово відмінним від SAW: він враховує взаємозалежності параметрів через чотири матриці зв'язків і відсіює індустріально нереалістичні комбінації. Повний морфологічний простір МТ Фільм складає 2 268 конфігурацій, коректна обробка яких є можливою лише засобами МММА.

Функція F_3 : Обрано варіант А – Streamlit. Завдяки нативній підтримці pandas DataFrame та вбудованих горизонтальних діаграм, цей фреймворк дозволяє відображати п'ять груп стратегій без необхідності розробляти HTML-шаблони.

Сформовано два варіанти реалізації:

$$F_1(a) - F_2(a) - F_3(a)$$

$$F_1(b) - F_2(b) - F_3(b)$$

4.3 Обґрунтування систем параметрів програмного продукту

Для кількісного оцінювання варіантів реалізації визначено систему параметрів, які максимально повно відображають споживчі властивості СППР.

X1 – потенційний обсяг програмного коду: характеризує складність реалізації та зусилля на подальшу підтримку. Менший обсяг при повній функціональності є кращим показником.

X2 – якість відсіювання нереалістичних конфігурацій: відображає адекватність математичної моделі – відсоток конфігурацій МТ Фільм, оцінка яких системою відповідає реальним кінематографічним закономірностям, підтвердженим даними онтологічної БЗ. Більше значення є кращим.

X3 – час розрахунку рекомендацій: визначає швидкодію системи від натискання кнопки до виведення повного рейтингу стратегій. Менший час є кращим.

X4 – функціональна повнота системи: кількість незалежно оцінюваних груп маркетингових рішень, що система здатна сформувати одночасно. Більша кількість є кращою.

Гірші, середні та кращі показники параметрів встановлюються на основі вимог замовника та умов, що описують використання програмного продукту, як наведено в таблиці 4.2.

Таблиця 4.2 – Основні параметри програмного продукту

Назва параметра	Умовне позначення	Одиниці виміру	Значення параметра		
			гірше	середнє	краще
Потенційний обсяг коду	X1	рядків коду	3 200	2 000	1 100
Якість відсіювання конфігурацій	X2	%	50	70	85
Час розрахунку рекомендацій	X3	с	10	4	1
Функціональна повнота	X4	груп рішень	2	3	5

За заданими значеннями в таблиці 4.2 будуються графіки характеристики параметрів (рисунок 4.2 – 4.5).

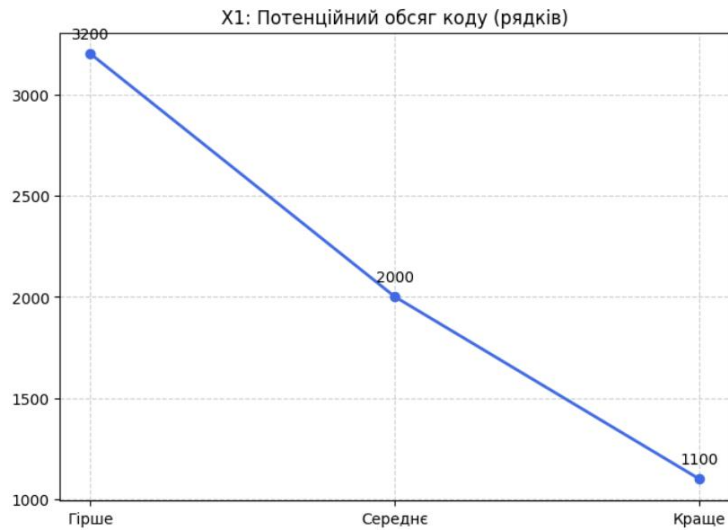


Рисунок 4.2 – X1, потенційний обсяг коду



Рисунок 4.3 – Х2, якість відсіювання конфігурацій

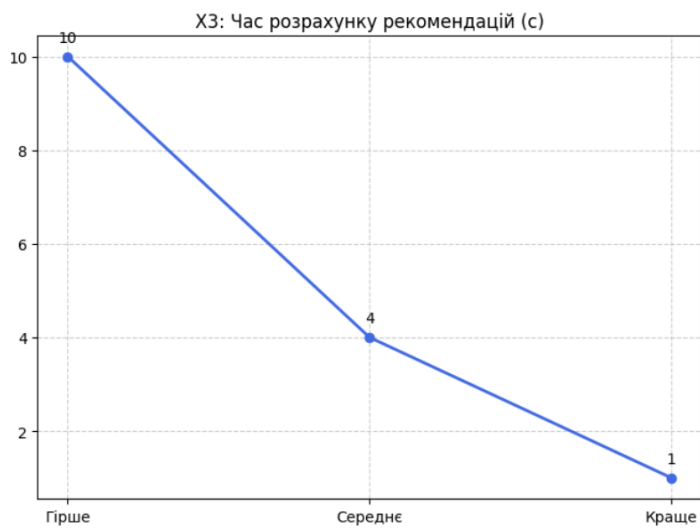


Рисунок 4.4 – Х3, час розрахунку рекомендацій



Рисунок 4.5 – Х4, функціональна повнота

4.4 Аналіз експертного оцінювання параметрів

Значимість кожного параметра визначається методом попарного порівняння. Оцінку проводить експертна комісія із семи осіб – фахівців у галузі системного аналізу, математичного моделювання та інформаційних систем. Кожен експерт незалежно присвоїв ранги параметрам від 1 до 4, де 1 – найважливіший параметр. Результати зведено у таблицю 4.3.

Таблиця 4.3 – Результати ранжування параметрів

Позначення параметра	Ранг параметра за оцінкою експерта							Сума рангів R_i	Відхилення Δ_i	Δ_i^2
	1	2	3	4	5	6	7			
X1	4	4	3	4	4	3	4	26	8,5	72,25
X2	1	1	1	1	2	1	1	8	-9,5	90,25
X3	2	3	2	2	1	2	2	14	-3,5	12,25
X4	3	2	4	3	3	4	3	22	4,5	20,25
Загальна кількість	10	10	10	10	10	10	10	70	0	195

Перевіримо достовірність експертних оцінок.

Загальна сума рангів:

$$R_i = \sum_{j=1}^N r_{ij} R_{ij} = \frac{Nn(n+1)}{2} = 70,$$

де N – число експертів;

n – кількість параметрів.

Середня сума рангів:

$$T = \frac{1}{n} R_j = 17,5$$

Відхилення суми рангів кожного параметра від середньої:

$$\Delta_i = R_i - T$$

Загальна сума квадратів відхилення:

$$S = \sum_{i=1}^N \Delta_i^2 = 195$$

Коефіцієнт узгодженості:

$$W = \frac{12S}{N^2(n^3 - n)} = \frac{12 \cdot 195}{7^2 \cdot (4^3 - 4)} \approx 0,796 > W_k = 0,67$$

Ранжування можна вважати достовірним, тому що знайдений коефіцієнт узгодженості перевищує нормативний, котрий дорівнює 0,67.

На основі результатів проводиться попарне порівняння параметрів (таблиця 4.4).

Таблиця 4.4 – Попарне порівняння параметрів

Параметри	Експерти							Кінцева оцінка	Числове значення
	1	2	3	4	5	6	7		
X1 і X2	<	<	<	<	<	<	<	<	0,5
X1 і X3	<	>	<	<	<	<	<	<	0,5
X1 і X4	>	>	>	>	>	>	>	>	1,5
X2 і X3	>	>	>	>	=	>	>	>	1,5
X2 і X4	>	>	>	>	>	>	>	>	1,5
X3 і X4	>	>	<	>	<	>	>	>	1,5

Числове значення, що визначає ступінь переваги i -го параметра над j тим a_{ij} визначається за формулою:

$$a_{ij} = \begin{cases} 1,5, & \text{при } X_i > X_j \\ 1,0, & \text{при } X_i = X_j \\ 0,5, & \text{при } X_i < X_j \end{cases}$$

З отриманих числових оцінок переваги складаємо матрицю $A = \|a_{ij}\|$ та розраховуємо вагомість K_{bi} за формулами:

$$K_{bi} = \frac{b_i}{\sum_{i=1}^n b_i}$$

$$b_i = \sum_{j=1}^N a_{ij}$$

Відносні оцінки розраховуються декілька разів доти, поки наступні значення не будуть незначно відрізнятися від попередніх (менше 2%). На другому і наступних кроках відносні оцінки розраховуються за наступними формулами:

$$K_{bi} = \frac{b'_i}{\sum_{i=1}^n b'_i}$$

$$b'_i = \sum_{j=1}^N a_{ij} b_j$$

Розрахунки вагомості параметрів зображено в таблиці 4.5.

Таблиця 4.5 – Розрахунок вагомості параметрів

Параметри x_i	Параметри x_j				Перша ітерація		Друга ітерація		Третя ітерація	
	X1	X2	X3	X4	b_i	K_{bi}	b'_i	K'_{bi}	b''_i	K''_{bi}
X1	1,0	0,5	0,5	1,5	3,5	0,219	12,25	0,208	44,63	0,207
X2	1,5	1,0	1,5	1,5	5,5	0,344	21,25	0,360	78,13	0,362
X3	1,5	0,5	1,0	1,5	4,5	0,281	16,25	0,276	59,38	0,275
X4	0,5	0,5	0,5	1,0	2,5	0,156	9,25	0,157	33,75	0,156
Всього					16	1,000	59	1,000	216	1,000

Різниця між другою та третьою ітераціями не перевищує 2%, тому результати третьої ітерації є остаточними.

4.5 Аналіз рівня якості варіантів реалізації функції

Оцінюємо якість реалізації кожної основної функції окремо. Записуємо показники технічного рівня якості варіантів реалізації програмного продукту в таблицю 4.6 та обчислюємо коефіцієнт технічного рівня за формулою:

$$K_K(j) = \sum_{i=1}^n K_{bi,j} B_{i,j},$$

де K_{Bi} – коефіцієнт вагомості i -го параметра;
 B_i – оцінка i -го параметра в балах.

Для варіанту 1: обсяг коду $\approx 1\,350$ рядків (три взаємопов'язаних модулі), якість відсіювання $\approx 82\%$, час розрахунку $\approx 1,5$ с, кількість груп рішень 5.

Для варіанту 2: обсяг коду $\approx 2\,800$ рядків через HTML-шаблони і схему БД, якість відсіювання $\approx 50\%$, час розрахунку $\approx 0,7$ с, кількість груп рішень 3.

Таблиця 4.6 – Розрахунок показників рівня якості варіантів реалізації

Основні функції	Варіант реалізації	Параметр	Абсолютне значення	Бальна оцінка параметра	Коефіцієнт вагомості параметра	Внесок рівня якості i
F1-F3	1	X1	1 350	78	0,207	16,15
		X2	82%	85	0,362	30,77
		X3	1,5 с	88	0,275	24,20
		X4	5 груп	95	0,156	14,82
F1-F3	2	X1	2 800	25	0,207	5,18
		X2	50%	35	0,362	12,67
		X3	0,7 с	96	0,275	26,40
		X4	3 групи	45	0,156	7,02

За формулою визначаємо рівень якості кожного варіанту:

$$K_{K1} = 16,15 + 30,77 + 24,20 + 14,82 = 85,94;$$

$$K_{K2} = 5,18 + 12,67 + 26,40 + 7,02 = 51,27.$$

Спираючись на отримані результати, можна зробити висновок, що перший варіант є кращим, оскільки коефіцієнт технічного рівня більше. Таким чином, для подальшої реалізації обрано його.

4.6 Економічний аналіз варіантів розробки програмного продукту

Для визначення вартості розробки програмного продукту спочатку проведемо розрахунок трудомісткості.

Обидва варіанти включають у себе такі окремі завдання:

- 1) розробка та наповнення бази знань;
- 2) реалізація математичного ядра алгоритму рекомендацій;
- 3) розробка інтерфейсу користувача та тестування системи.

Завдання 1 за ступенем новизни відноситься до групи А, завдання 2 – до групи Б, завдання 3 – до групи В. За складністю алгоритми, що використовуються в завданнях, належать до груп 2-3.

Для реалізації завдання 1 використовується довідкова та технічна інформація щодо побудови онтологій, структури датасетів IMDb та TMDb API. Завдання 2 використовує матричні дані узгодженостей і результати парсингу онтології. Завдання 3 передбачає розробку інтерактивного веб-інтерфейсу на основі отриманих результатів.

Загальна трудомісткість обчислюється за формулою:

$$T_0 = T_P \cdot K_P \cdot K_{СК} \cdot K_M \cdot K_{СТ} \cdot K_{СТ.М},$$

де T_P – трудомісткість розробки ПП;

K_P – поправочний коефіцієнт;

$K_{СК}$ – коефіцієнт на складність вхідної інформації;

K_M – коефіцієнт рівня мови програмування;

$K_{СТ}$ – коефіцієнт використання стандартних модулів;

$K_{СТ.М}$ – коефіцієнт стандартного математичного забезпечення.

Для першого завдання, виходячи з норм часу для завдань ступеню новизни А та групи складності алгоритму 2, трудомісткість становить $T_P = 42$ людино-дні. Поправочний коефіцієнт $K_P = 1,1$. Коефіцієнт складності вхідної

інформації $K_{СК} = 1$. Оскільки при розробці використовується мова Python ($K_M = 0,85$) та стандартні модулі ($K_{СТ} = 0,9$), загальна трудомісткість першого завдання для варіанту 1:

$$T_1(I) = 42 \cdot 1,1 \cdot 1 \cdot 0,85 \cdot 0,9 = 35,34 \text{ людино-днів}$$

Для варіанту 2 трудомісткість менша: $T_P = 28$, $K_{П} = 1,0$, $K_{СТ} = 1,0$:

$$T_1(II) = 28 \cdot 1,0 \cdot 1 \cdot 0,85 \cdot 1,0 = 23,80 \text{ людино-днів.}$$

Для другого завдання варіант 1 – двоетапний МММА з чотирма матрицями зв'язків: $T_P = 48$, $K_{П} = 1,0$, $K_{СТ} = 0,8$:

$$T_2(I) = 48 \cdot 1,0 \cdot 1 \cdot 0,85 \cdot 0,8 = 32,64 \text{ людино-днів.}$$

Варіант 2 – алгоритм SAW, значно простіший математично: $T_P = 22$, $K_{П} = 0,9$, $K_{СТ} = 1,0$:

$$T_2(II) = 22 \cdot 0,9 \cdot 1 \cdot 0,85 \cdot 1,0 = 16,83 \text{ людино-днів.}$$

Для третього завдання варіант 1 – Streamlit, не потребує HTML-шаблонів: $T_P = 25$, $K_{П} = 0,9$, $K_{СТ} = 0,8$:

$$T_3(I) = 25 \cdot 0,9 \cdot 1 \cdot 0,85 \cdot 0,8 = 15,30 \text{ людино-днів.}$$

Варіант 2 – Flask та HTML-шаблони, значно більший обсяг розробки інтерфейсу: $T_P = 42$, $K_{П} = 1,0$, $K_{СТ} = 0,9$:

$$T_3(II) = 42 \cdot 1,0 \cdot 1 \cdot 0,85 \cdot 0,9 = 32,13 \text{ людино-днів.}$$

Складаємо трудомісткість завдань для кожного варіанту:

$$T_I = (35,34 + 32,64 + 15,30) \cdot 8 = 666,24 \text{ людино-годин.}$$

$$T_{II} = (23,80 + 16,83 + 32,13) \cdot 8 = 582,08 \text{ людино-годин.}$$

У розробці беруть участь науковий керівник дипломної роботи з місячним окладом 36 000 грн. та студент-розробник з місячним окладом 21 000 грн.. Визначаємо середню погодинну ставку за формулою:

$$C_{\text{ч}} = \frac{M}{T_m \cdot t},$$

де M – місячний оклад працівників;

T_m – кількість робочих днів на місяць;

t – кількість робочих годин на день.

$$C_{\text{ч}} = \frac{21000 + 36000}{2 \cdot 21,1 \cdot 8} = 168,84 \text{ грн./год}$$

Заробітна плата розробника, з урахуванням додаткової, $K_D = 1,2$, визначається за формулою:

$$C_{\text{ЗП}} = C_{\text{ч}} \cdot T_i \cdot K_D,$$

де $C_{\text{ч}}$ – величина погодинної оплати праці;

T_i – трудомісткість відповідного завдання;

K_D – норматив, який враховує додаткову заробітну плату.

Заробітна плата за варіантами становить:

$$\text{I. } C_{3\Pi} = 168,84 \cdot 666,24 \cdot 1,2 = 134985,59 \text{ грн.}$$

$$\text{II. } C_{3\Pi} = 168,84 \cdot 582,08 \cdot 1,2 = 117934,18 \text{ грн.}$$

Відрахування на єдиний соціальний внесок становить 22%:

$$\text{I. } C_{\text{ВІД}} = 134985,59 \cdot 0,22 = 29696,83 \text{ грн.}$$

$$\text{II. } C_{\text{ВІД}} = 117934,18 \cdot 0,22 = 25945,52 \text{ грн.}$$

Визначимо витрати на оплату однієї машино-години. Одна ЕОМ обслуговує студента-розробника з окладом 21 000 грн. із коефіцієнтом зайнятості $K_3 = 0,2$:

$$C_{\Gamma} = 12 \cdot M \cdot K_3 = 12 \cdot 21000 \cdot 0,2 = 50400 \text{ грн.}$$

З урахуванням додаткової заробітної плати:

$$C_{3\Pi}(\text{ЕОМ}) = C_{\Gamma} \cdot (1 + K_{\text{Д}}) = 50400 \cdot (1 + 0,2) = 60480 \text{ грн.}$$

Відрахування на єдиний соціальний внесок:

$$C_{\text{ВІД}}(\text{ЕОМ}) = 60480 \cdot 0,22 = 13305,6 \text{ грн.}$$

Амортизаційні відрахування розраховуємо при нормі амортизації 25% та вартості ноутбука 40 000 грн:

$$C_{\text{А}} = K_{\text{ТМ}} \cdot K_{\text{А}} \cdot \text{Ц}_{\text{ПР}} = 1,15 \cdot 0,25 \cdot 40000 = 11500 \text{ грн.,}$$

де $K_{\text{ТМ}}$ – коефіцієнт, який враховує витрати на транспортування та монтаж;
 $K_{\text{А}}$ – річна норма амортизації;

$C_{\text{ПР}}$ – договірна ціна приладу.

Витрати на ремонт та профілактику при $K_{\text{Р}} = 5\%$:

$$C_{\text{Р}} = K_{\text{ТМ}} \cdot C_{\text{ПР}} \cdot K_{\text{Р}} = 1,15 \cdot 40000 \cdot 0,05 = 2300 \text{ грн.},$$

де $K_{\text{Р}}$ – відсоток витрат на поточні ремонти.

Ефективний річний фонд часу роботи ПК за рік:

$$T_{\text{ЕФ}} = (D_{\text{К}} - D_{\text{В}} - D_{\text{С}} - D_{\text{Р}}) \cdot t_{\text{З}} \cdot K_{\text{В}} = (242) \cdot 8 \cdot 0,85 = 1645,6 \text{ год.},$$

де $D_{\text{К}}$ – календарна кількість днів у році;

$D_{\text{В}}$ – вихідні дні;

$D_{\text{С}}$ – святкові дні;

$D_{\text{Р}}$ – дні планових ремонтів;

$t_{\text{З}}$ – кількість робочих годин у день;

$K_{\text{В}}$ – коефіцієнт використання приладу у часі.

Витрати на оплату електроенергії розраховуємо за формулою (тариф для споживачів II класу напруги складає 13,64 грн/кВт·год, потужність $N_{\text{С}} = 0,065$ кВт):

$$C_{\text{ЕЛ}} = T_{\text{ЕФ}} \cdot N_{\text{С}} \cdot K_{\text{З}} \cdot C_{\text{ЕН}} = 1645,6 \cdot 0,065 \cdot 0,2 \cdot 13,64 = 291,74 \text{ грн.},$$

де $N_{\text{С}}$ – середньоспоживча потужність приладу;

$K_{\text{З}}$ – коефіцієнт зайнятості приладу;

$C_{\text{ЕН}}$ – тариф за 1 кВт·год електроенергії.

Накладні витрати складають 67% від вартості ЕОМ:

$$C_H = C_{\text{ПР}} \cdot 0,67 = 40000 \cdot 0,67 = 26800 \text{ грн.}$$

Тоді, річні експлуатаційні витрати будуть:

$$C_{\text{ЕКС}} = C_{\text{ЗП}} + C_{\text{ВІД}} + C_A + C_P + C_{\text{ЕЛ}} + C_H$$

$$\begin{aligned} C_{\text{ЕКС}} &= 60480 + 13305,6 + 11500 + 2300 + 291,74 + 26800 \\ &= 114677,34 \text{ грн.} \end{aligned}$$

Собівартість однієї машино-години ЕОМ дорівнює:

$$C_{\text{М-Г}} = \frac{C_{\text{ЕКС}}}{T_{\text{ЕФ}}} = \frac{114677,34}{1645,6} = 69,69 \text{ грн/год.}$$

Оскільки всі роботи, пов'язані з розробкою програмного продукту, ведуться на ЕОМ, витрати на оплату машинного часу залежно від обраного варіанту реалізації становлять:

$$C_M = C_{\text{М-Г}} \cdot T_i$$

$$\text{I. } C_M = 69,69 \cdot 666,24 = 46\,430,27 \text{ грн.}$$

$$\text{II. } C_M = 69,69 \cdot 582,08 = 40\,565,15 \text{ грн.}$$

Накладні витрати складають 67% від заробітної плати:

$$C_H = C_{\text{ЗП}} \cdot 0,67$$

$$\text{I. } C_H = 134985,59 \cdot 0,67 = 90440,35 \text{ грн.}$$

$$\text{II. } C_H = 117934,18 \cdot 0,67 = 79015,90 \text{ грн.}$$

Отже, вартість розробки програмного продукту за варіантами становить:

$$C_{\text{ПП}} = C_{\text{ЗП}} + C_{\text{ВІД}} + C_{\text{М}} + C_{\text{Н}}$$

$$\text{I. } C_{\text{ПП}} = 134985,59 + 29696,83 + 46430,27 + 90440,35 = 301553,04 \text{ грн.}$$

$$\text{II. } C_{\text{ПП}} = 117934,18 + 25945,52 + 40565,15 + 79015,90 = 263460,75 \text{ грн.}$$

4.7 Вибір кращого варіанту ПП за техніко-економічним рівнем

Розрахуємо коефіцієнт техніко-економічного рівня за формулою:

$$K_{\text{ТЕР}j} = \frac{K_{kj}}{C_{\text{ПП}j}}$$

$$K_{\text{ТЕР}1} = \frac{85,94}{301553,04} = 2,850 \cdot 10^{-4}$$

$$K_{\text{ТЕР}2} = \frac{51,27}{263460,75} = 1,946 \cdot 10^{-4}$$

Як бачимо, найбільш ефективним є перший варіант реалізації програмного продукту з коефіцієнтом техніко-економічного рівня $K_{\text{ТЕР}1} = 2,850 \cdot 10^{-4}$. Незважаючи на дещо більшу вартість розробки, перший варіант забезпечує значно вищий технічний рівень завдяки застосуванню OWL-онтології та повноцінного двоетапного МММА.

Після виконання функціонально-вартісного аналізу програмного продукту, що розробляється, можна зробити висновок, що з розглянутих варіантів реалізації оптимальним є перший. Цей варіант передбачає використання OWL-онтологічної бази знань з автоматизованим наповненням, математичного ядра СППР на основі двоетапного МММА з чотирма матрицями зв'язків, а також інтерактивного веб-інтерфейсу на базі Streamlit. Такий підхід забезпечує оптимальне співвідношення між функціональними

можливостями системи, якістю відсіювання нереалістичних конфігурацій та вартістю розробки.

4.8 Висновки до розділу 4

У даному розділі було проведено функціонально-вартісний аналіз програмного продукту – системи підтримки прийняття рішень для персоналізації маркетингових стратегій у медіаіндустрії.

Головну функцію системи було декомпозиовано на три підфункції: побудова бази знань, реалізація математичного алгоритму та розробка інтерфейсу. Для порівняння сформовано два комплексні варіанти. Перший порівнювався з другим.

Для оцінювання обрано ключові параметри: потенційний обсяг коду, якість відсіювання конфігурацій, час розрахунку та функціональна повнота. За результатами експертного оцінювання, достовірність якого підтверджено коефіцієнтом конкордації $W = 0,796$, найвищу вагомість отримав параметр X2, що відображає ефективність фільтрації алгоритмом МММА. Розрахунки довели значну перевагу першого варіанту за рівнем якості.

Економічний аналіз виконано з урахуванням сучасних вимог: встановлено диференційовані оклади та застосовано тариф на електроенергію 13,64 грн/кВт·год для II класу напруги. Повна вартість розробки першого варіанту склала 301 553,04 грн.

Остаточний вибір підтверджено розрахунком коефіцієнта техніко-економічного рівня: $K_{\text{ТЕР1}} = 2,850 \cdot 10^{-4}$, що перевищує показник конкурента. Отже, реалізація СППР на базі OWL-онтології, МММА та Streamlit забезпечує найкраще співвідношення технічного рівня та вартості, що повністю обґрунтовує доцільність прийнятих інженерних рішень.

ВИСНОВКИ

В рамках даної дипломної роботи було розроблено систему підтримки прийняття рішень для персоналізації маркетингових стратегій у медіаіндустрії.

У процесі розробки було виконано повний життєвий цикл проектування. Було проаналізовано предметну область, виявлено основні чинники, що впливають на вибір стратегій просування та формату релізу кінофільмів. Були детально описані основні ідеї та наведено використання теорії онтологій і двоетапного МММА для розв'язання задач з високим рівнем ризиків та невизначеності.

За структуру системи було взято мережу взаємопов'язаних морфологічних таблиць, що об'єднує профіль користувача, характеристики фільму та маркетингові рішення. Для забезпечення об'єктивності початкових оцінок було спроектовано онтологічну базу знань у форматі OWL, яка автоматизовано наповнюється стратифікованою вибіркою реальних фільмів. Для програмної реалізації обчислювального алгоритму використано мову програмування Python. Для розробки інтерактивного графічного інтерфейсу використано фреймворк Streamlit.

Отримані результати показали, що поєднання онтологічного моделювання та МММА з нерівномірним розподілом оцінок дійсно дає точні результати, ефективно відсіює нереалістичні конфігурації фільмів і дозволяє формувати логічно обґрунтовані персоналізовані рекомендації.

Подальший розвиток даного програмного продукту може включати в себе розширення онтологічної бази знань новими параметрами, а також інтеграцію алгоритмів збору даних про тренди медіаринку в реальному часі. При реалізації такого сценарію можна досягти часткової або навіть повної автоматизації роботи СППР та динамічного оновлення рекомендацій системи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Григорова З. В., Сухорукова О. А., Кваско А. В., Шендерівська Л. П. Основи медіабізнесу: підручник. Київ: КПІ ім. Ігоря Сікорського, 2021. 323 с.
2. Arkenberg C. et al. 2025 Media and Entertainment Outlook. *Deloitte Insights*. 2025. 23 April. URL: <https://www.deloitte.com/us/en/insights/industry/technology/technology-media-telecom-outlooks/2025-media-entertainment-outlook.html> (Last accessed: 03.06.2026).
3. Xue D. A Study of Evolution of Film Marketing in the Digital Age. Applied Psychology and Marketing Management International Conference (APMM 2024), Singapore, May 17–19, 2024: *SHS Web of Conferences*. 2024. Vol. 193. Art. 04003. DOI: 10.1051/shsconf/202419304003.
4. Kim J., Kim M. Rise of Social Media Influencers as a New Marketing Channel: Focusing on the Roles of Psychological Well-Being and Perceived Social Responsibility among Consumers. *International Journal of Environmental Research and Public Health*. 2022. Vol. 19. Art. 2362. DOI: 10.3390/ijerph19042362.
5. Irfandianto M. E., Suprpto A. Q., Irwansyah. From Cinema to Small Screen: The Revolution of Film Distribution with OTT Streaming. *International Journal for Multidisciplinary Research (IJFMR)*. 2024. Vol. 6, Issue 3. Art. IJFMR240321558. URL: <https://www.ijfmr.com/papers/2024/3/21558.pdf> (Last accessed: 03.06.2026).
6. Saaty T. L. Decision making with the analytic hierarchy process. *International Journal of Services Sciences*. 2008. Vol. 1, No. 1. P. 83–98.
7. Sukarta A. B., Hidayat M. M., Zainal R. F. Decision Support System for Movie Recommendations Based on Multi User Preferences Using the Simple

- Additive Weighting Method. *Journal of Electrical Engineering and Computer Sciences*. 2022. Vol. 7, No. 2. P. 1285–1292. DOI: 10.54732/jeecs.v7i2.22.
8. Herwandi R. R., Wibowo A. P. W. Decision Support System for Movie Trend Analysis in 2025 Using the Naïve Bayes Algorithm. *Brilliance: Research of Artificial Intelligence*. 2025. Vol. 5, No. 2. P. 1063–1069. DOI: 10.47709/brilliance.v5i2.7092.
 9. Cinelytic: AI-driven Entertainment Analytics. URL: <https://www.cinelytic.com/> (Last accessed: 03.06.2026).
 10. Gower Street Analytics. URL: <https://gower.st/> (Last accessed: 03.06.2026).
 11. Noy N. F., McGuinness D. L. Ontology Development 101: A Guide to Creating Your First Ontology: Technical Report KSL-01-05. Stanford, CA: Stanford University, Knowledge Systems Laboratory, 2001. 25 p. URL: https://protege.stanford.edu/publications/ontology_development/ontology101.pdf (Last accessed: 03.06.2026).
 12. Keet C. M. An Introduction to Ontology Engineering. Version 1.5. 2020. 306 p. URL: <https://people.cs.uct.ac.za/~mkeet/OEbook/> (Last accessed: 03.06.2026).
 13. DeBellis M. A Practical Guide to Building OWL Ontologies Using Protégé 5.5 and Plugins. Edition 3.0. 2021. 90 p. URL: <https://www.michaeldebellis.com/post/new-protege-pizza-tutorial> (Last accessed: 03.06.2026).
 14. Савченко І. О. Методологічне і математичне забезпечення розв’язання задач передбачення на основі модифікованого методу морфологічного аналізу. *Системні дослідження та інформаційні технології*. 2011. № 3. С. 18–27.
 15. Панкратова Н. Д., Гайко Г. І., Савченко І. О. Розвиток підземної урбаністики як системи альтернативних проектних конфігурацій: монографія. Київ, 2019.

16. Savchenko I. Estimation of Morphological Tables Using Text Analysis Results. *Computer Science Journal of Moldova*. 2016. Vol. 24, No. 2 (71). P. 148–156.

ДОДАТОК А МОДУЛЬ НАПОВНЕННЯ ОНТОЛОГІЇ

```

import os
import re
import time
import pandas as pd
from owlready2 import *
import requests

TMDB_API_KEY = "922e0660a8a4ba7ffe0a67494dfde0d5"
BASE_DIR = os.path.dirname(os.path.abspath(__file__))

FILE_BASICS = os.path.join(BASE_DIR, 'title.basics.tsv')
FILE_CREW = os.path.join(BASE_DIR, 'title.crew.tsv')
FILE_NAMES = os.path.join(BASE_DIR, 'name.basics.tsv')
FILE_PRINCIPALS = os.path.join(BASE_DIR, 'title.principals.tsv')
FILE_RATINGS = os.path.join(BASE_DIR, 'title.ratings.tsv')

ONTOLOGY_FILE = os.path.join(BASE_DIR, 'ura2.owl')
OUTPUT_FILE = os.path.join(BASE_DIR, 'final_updated_ura2mim.owl')

df_ratings = pd.read_csv(FILE_RATINGS, sep='\t', na_values='\\N', low_memory=False)
df_ratings.set_index('tconst', inplace=True)
df_basics = pd.read_csv(FILE_BASICS, sep='\t', na_values='\\N', low_memory=False)

df_all = df_basics[(df_basics['titleType'] == 'movie') & (df_basics['startYear'] >=
1990) & (df_basics['startYear'] <= 2026)].copy()
df_all = df_all.merge(df_ratings[['numVotes']], left_on='tconst', right_index=True,
how='inner')
df_sorted = df_all.sort_values('numVotes', ascending=False)

df_movies = pd.concat([
    df_sorted.head(67),
    df_sorted.iloc[2000 : 2067],
    df_sorted.iloc[8000 : 8066]
]).copy()
target_tconsts = set(df_movies['tconst'].unique())

df_crew_filtered = pd.read_csv(FILE_CREW, sep='\t', na_values='\\N', low_memory=False)
df_crew_filtered = df_crew_filtered[df_crew_filtered['tconst'].isin(target_tconsts)]

chunks = []
for chunk in pd.read_csv(FILE_PRINCIPALS, sep='\t', na_values='\\N', low_memory=False,
chunks=200000):

```

```

        filtered = chunk[(chunk['tconst'].isin(target_tconsts)) &
        (chunk['category'].isin(['actor', 'actress']))]
        chunks.append(filtered)
    df_principals = pd.concat(chunks)

    relevant_nconsts = set(df_principals['nconst'].unique())
    for d_list in df_crew_filtered['directors'].dropna():
        relevant_nconsts.update(d_list.split(','))

    df_names = pd.read_csv(FILE_NAMES, sep='\t', na_values='\N', low_memory=False)
    df_names = df_names[df_names['nconst'].isin(relevant_nconsts)].copy()
    df_names.set_index('nconst', inplace=True)
    df_movies = df_movies.merge(df_crew_filtered[['tconst', 'directors']], on='tconst',
    how='left')

    onto_path = f"file://{ONTOLOGY_FILE}".replace("\\", "/")
    onto = get_ontology(onto_path).load()

    def fetch_tmdb_budget(imdb_id):
        if not TMDB_API_KEY: return 0
        try:
            url =
f"https://api.themoviedb.org/3/find/{imdb_id}?api_key={TMDB_API_KEY}&external_source=i
mdb_id"
            res = requests.get(url, timeout=5).json()
            if res.get('movie_results'):
                t_id = res['movie_results'][0]['id']
                m_data =
requests.get(f"https://api.themoviedb.org/3/movie/{t_id}?api_key={TMDB_API_KEY}",
timeout=5).json()
                return m_data.get('budget', 0)
        except: pass
        return 0

    def classify_budget(val, votes, title):
        if val > 0:
            if val >= 40000000: return onto.CategoryA
            if val >= 2000000: return onto.CategoryB
            return onto.LowBudget
        else:
            if votes > 350000: return onto.CategoryA
            if votes >= 40000: return onto.CategoryB
            return onto.LowBudget

    def get_pro_fame(nconst, is_director=False):
        if nconst not in df_names.index: return onto.RisingStar if is_director else
        onto.Debut

```

```

row = df_names.loc[nconst]
known = [t for t in str(row['knownForTitles']).split(',') if t and t != 'nan']
max_v = 0
for tid in known:
    if tid in df_ratings.index:
        v = df_ratings.loc[tid, 'numVotes']
        if v > max_v: max_v = v
if is_director: return onto.Famous if max_v > 200000 else onto.RisingStar
return onto.Famous if max_v > 400000 else (onto.RisingStar if max_v > 50000 else
onto.Debut)

def determine_main_genre_by_specificity(genres_str):
    if pd.isna(genres_str): return "Drama"
    genres = [g.strip().lower() for g in genres_str.split(',')]
    mapped_genres = []
    for g in genres:
        if g in ["history", "biography", "documentary"]:
mapped_genres.append("History")
        elif g in ["sci-fi", "fantasy", "animation"]: mapped_genres.append("Sci-
Fi/Fantasy")
        elif g in ["horror"]: mapped_genres.append("Horror")
        elif g in ["romance"]: mapped_genres.append("Romance")
        elif g in ["comedy", "family", "musical"]: mapped_genres.append("Comedy")
        elif g in ["action", "adventure", "thriller", "mystery", "crime", "western"]:
mapped_genres.append("Action")
        elif g in ["drama"]: mapped_genres.append("Drama")
        else: mapped_genres.append(g.title())
    specificity_priority = ["History", "Horror", "Sci-Fi/Fantasy", "Romance",
"Comedy", "Action", "Drama"]
    for priority in specificity_priority:
        if priority in mapped_genres: return priority
    return mapped_genres[0] if mapped_genres else "Drama"

with onto:
    class hasOverallCastFame(ObjectProperty, FunctionalProperty):
        domain = [onto.Movie]; range = [onto.FameLevel]
    class hasDirectorFameLevel(ObjectProperty, FunctionalProperty):
        domain = [onto.Movie]; range = [onto.FameLevel]
    class hasMainGenre(ObjectProperty, FunctionalProperty):
        domain = [onto.Movie]; range = [onto.Genre]

for m in list(onto.Movie.instances()): destroy_entity(m)

for idx, row in df_movies.iterrows():
    movie = onto.Movie(f"{re.sub(r'^a-zA-Z0-9', '_',
str(row['primaryTitle']))}_{int(row['startYear'])}")
    movie.label = [f"{row['primaryTitle']} ({int(row['startYear'])}")]

```

```

main_genre_target_name = determine_main_genre_by_specificity(row['genres'])
if pd.notna(row['genres']):
    for g in row['genres'].split(','):
        g_raw = g.strip()
        if g_raw.lower() in ["history", "biography", "documentary"]: target =
"History"
        elif g_raw.lower() in ["sci-fi", "fantasy", "animation"]: target = "Sci-
Fi/Fantasy"
        elif g_raw.lower() in ["horror"]: target = "Horror"
        elif g_raw.lower() in ["romance"]: target = "Romance"
        elif g_raw.lower() in ["comedy", "family", "musical"]: target = "Comedy"
        elif g_raw.lower() in ["action", "adventure", "thriller", "mystery",
"crime", "western"]: target = "Action"
        elif g_raw.lower() in ["drama"]: target = "Drama"
        else: target = g_raw.title()
        g_cls = onto[target]
        if g_cls:
            g_ind = onto.search_one(name=target, is_instance_of=g_cls)
            if not g_ind:
                try: g_ind = g_cls(target)
                except: g_ind = onto[target]
            if g_ind and g_ind not in movie.hasGenre: movie.hasGenre.append(g_ind)
            if target == main_genre_target_name: movie.hasMainGenre = g_ind
            if not movie.hasMainGenre and movie.hasGenre: movie.hasMainGenre =
movie.hasGenre[0]
        if pd.notna(row['directors']):
            d_id = str(row['directors']).split(',')[0]
            if d_id in df_names.index:
                d_ind = onto.Director(re.sub(r'^a-zA-Z0-9', '_',
str(df_names.loc[d_id]['primaryName'])))
                fame = get_pro_fame(d_id, True)
                d_ind.hasFameLevel = fame
                movie.hasDirector = d_ind
                movie.hasDirectorFameLevel = fame
            rt = float(row['runtimeMinutes']) if pd.notna(row['runtimeMinutes']) else 100
            movie.hasDuration = onto.ShortDuration if rt < 90 else (onto.MediumDuration if rt
<= 150 else onto.LongDuration)
            movie.hasBudget = classify_budget(fetch_tmdb_budget(row['tconst']),
row['numVotes'], row['primaryTitle'])
            t_low, g_low = str(row['primaryTitle']).lower(), str(row['genres']).lower()
            if any(m in t_low for m in ["based on", "novel", "book", "true story"]) or
"biography" in g_low:
                movie.hasPlotType = onto.Adaptation
            elif any(m in t_low for m in ["remake", "sequel", "part"]) or
bool(re.search(r'\b(II|III|IV|V|VI|[2-9])\b', str(row['primaryTitle']))):
                movie.hasPlotType = onto.Remake
            else: movie.hasPlotType = onto.Original

```

```

scores = []
for _, a_row in df_principals[df_principals['tconst'] ==
row['tconst']].iterrows():
    a_id = a_row['nconst']
    if a_id in df_names.index:
        a_ind = onto.Actor(re.sub(r'^a-zA-Z0-9', '_',
str(df_names.loc[a_id]['primaryName'])))
        fame = get_pro_fame(a_id, False)
        a_ind.hasFameLevel = fame
        movie.hasActor.append(a_ind)
        scores.append(3.0 if fame == onto.Famous else (1.5 if fame ==
onto.RisingStar else 0.0))
    if scores:
        avg = sum(scores) / len(scores)
        movie.hasOverallCastFame = onto.Famous if avg > 1.2 else (onto.RisingStar if
avg > 0.4 else onto.Debut)
    else: movie.hasOverallCastFame = onto.Debut

onto.save(file=OUTPUT_FILE, format="rdxml")

```

ДОДАТОК Б МАТЕМАТИЧНЕ ЯДРО СППР

```

import numpy as np
import pandas as pd
from itertools import product, combinations
from collections import defaultdict, Counter
import re
import os

BASE_DIR = os.path.dirname(os.path.abspath(__file__))

PATH_OWL = os.path.join(BASE_DIR, "final_updated_ura2mim.owl")
PATH_USER_MOVIE = os.path.join(BASE_DIR, "Користувач - Фільм.csv")
PATH_MOVIE_INT = os.path.join(BASE_DIR, "Фільм (Внутрішня).csv")
PATH_MOVIE_STRAT = os.path.join(BASE_DIR, "Фільм - Стратегія.csv")
PATH_STRAT_USER = os.path.join(BASE_DIR, "Стратегія - Користувач.csv")

USER_PARAMS = {
    'K1': ['K1.1', 'K1.2', 'K1.3', 'K1.4', 'K1.5', 'K1.6', 'K1.7'],
    'K2': ['K2.1', 'K2.2', 'K2.3'],
    'K3': ['K3.1', 'K3.2', 'K3.3'],
    'K4': ['K4.1', 'K4.2', 'K4.3'],
}

USER_LABELS = {
    'K1.1': 'Екшн', 'K1.2': 'Комедія', 'K1.3': 'Драма', 'K1.4': 'Фантастика/Фентезі',
    'K1.5': 'Хоррор', 'K1.6': 'Історичний', 'K1.7': 'Романтика',
    'K2.1': 'Дорослий (18+)', 'K2.2': 'Підліток (10-17)', 'K2.3': 'Дитина (3-9)',
    'K3.1': 'Висока цифр. акт.', 'K3.2': 'Середня цифр. акт.', 'K3.3': 'Низька цифр. акт.',
    'K4.1': 'Кінотеатр', 'K4.2': 'Стримінг', 'K4.3': 'Гібрид',
}

MOVIE_PARAMS = {
    'Ф1': ['Ф1.1', 'Ф1.2', 'Ф1.3', 'Ф1.4', 'Ф1.5', 'Ф1.6', 'Ф1.7'],
    'Ф2': ['Ф2.1', 'Ф2.2'],
    'Ф3': ['Ф3.1', 'Ф3.2', 'Ф3.3'],
    'Ф4': ['Ф4.1', 'Ф4.2', 'Ф4.3'],
    'Ф5': ['Ф5.1', 'Ф5.2', 'Ф5.3'],
    'Ф6': ['Ф6.1', 'Ф6.2', 'Ф6.3'],
}

MOVIE_LABELS = {
    'Ф1.1': 'Екшн', 'Ф1.2': 'Комедія', 'Ф1.3': 'Драма', 'Ф1.4': 'Фантастика/Фентезі',
    'Ф1.5': 'Хоррор', 'Ф1.6': 'Історичний', 'Ф1.7': 'Романтика',

```

```

    'Ф2.1': 'Відомий реж.', 'Ф2.2': 'Маловідомий реж.',
    'Ф3.1': 'Короткий (60-90 хв)', 'Ф3.2': 'Середній (90-150 хв)', 'Ф3.3': 'Довгий
(150+ хв)',
    'Ф4.1': 'Великий (30+ млн)', 'Ф4.2': 'Середній (2-20 млн)', 'Ф4.3': 'Малий (<2
млн)',
    'Ф5.1': 'Оригінальний', 'Ф5.2': 'Адаптація', 'Ф5.3': 'Ремейк',
    'Ф6.1': 'Відомі актори', 'Ф6.2': 'Набирають популярність', 'Ф6.3': 'Дебютанти',
}

```

```

STRAT_PARAMS = {
    'C7': ['C7.1', 'C7.2', 'C7.3'],
    'C8': ['C8.1', 'C8.2'],
    'C9': ['C9.1', 'C9.2', 'C9.3'],
    'C10': ['C10.1', 'C10.2', 'C10.3'],
    'C11': ['C11.1', 'C11.2', 'C11.3'],
}

```

```

STRAT_LABELS = {
    'C7.1': 'Офлайн реклама', 'C7.2': 'Онлайн реклама', 'C7.3': 'Нестандартна
реклама',
    'C8.1': 'Традиційна реклама', 'C8.2': 'PR-заходи',
    'C9.1': 'SMM', 'C9.2': 'Контент-маркетинг', 'C9.3': 'Цифрова реклама',
    'C10.1': 'Influencer-маркетинг', 'C10.2': 'Партизанський маркетинг', 'C10.3':
'Гейміфікація',
    'C11.1': 'Кіно-реліз', 'C11.2': 'Онлайн-ексклюзив', 'C11.3': 'Гібридний реліз',
}

```

```

def load_csv(path: str) -> pd.DataFrame:
    df = pd.read_csv(path, sep=';', index_col=0, encoding='utf-8-sig')
    df = df.fillna(0)
    df.index = df.index.astype(str).str.strip()
    df.columns = df.columns.astype(str).str.strip()
    return df

```

```

def cell(df: pd.DataFrame, row_code: str, col_code: str) -> float:
    r_found, c_found = None, None
    for r in df.index:
        if str(r).startswith(row_code):
            r_found = r; break
    for c in df.columns:
        if str(c).startswith(col_code):
            c_found = c; break
    if r_found is not None and c_found is not None:
        return float(df.at[r_found, c_found])
    return 0.0

```

```

ONT_GENRE_MAP = {'Action': 'Φ1.1', 'Comedy': 'Φ1.2', 'Drama': 'Φ1.3', 'Sci-
Fi/Fantasy': 'Φ1.4', 'Horror': 'Φ1.5', 'History': 'Φ1.6', 'Romance': 'Φ1.7'}
ONT_DIR_MAP = {'Famous': 'Φ2.1', 'RisingStar': 'Φ2.2', 'Debut': 'Φ2.2'}
ONT_DUR_MAP = {'ShortDuration': 'Φ3.1', 'MediumDuration': 'Φ3.2', 'LongDuration':
'Φ3.3'}
ONT_BUD_MAP = {'CategoryA': 'Φ4.1', 'CategoryB': 'Φ4.2', 'LowBudget': 'Φ4.3'}
ONT_PLOT_MAP = {'Original': 'Φ5.1', 'Adaptation': 'Φ5.2', 'Remake': 'Φ5.3'}
ONT_CAST_MAP = {'Famous': 'Φ6.1', 'RisingStar': 'Φ6.2', 'Debut': 'Φ6.3'}

def parse_ontology_frequencies(owl_path: str) -> dict:
    with open(owl_path, 'r', encoding='utf-8') as f:
        content = f.read()

    def extract_values(prop_name):
        return re.findall(rf'{prop_name} rdf:resource="[^"]*#[^"]+', content)

    genres = extract_values('hasMainGenre')
    budgets = extract_values('hasBudget')
    durations = extract_values('hasDuration')
    plots = extract_values('hasPlotType')
    cast_fame = extract_values('hasOverallCastFame')
    dir_fame = extract_values('hasDirectorFameLevel')

    freqs = defaultdict(Counter)
    for g in genres:
        code = ONT_GENRE_MAP.get(g)
        if code: freqs['Φ1'][code] += 1
    for d in dir_fame:
        code = ONT_DIR_MAP.get(d)
        if code: freqs['Φ2'][code] += 1
    for d in durations:
        code = ONT_DUR_MAP.get(d)
        if code: freqs['Φ3'][code] += 1
    for b in budgets:
        code = ONT_BUD_MAP.get(b)
        if code: freqs['Φ4'][code] += 1
    for p in plots:
        code = ONT_PLOT_MAP.get(p)
        if code: freqs['Φ5'][code] += 1
    for c in cast_fame:
        code = ONT_CAST_MAP.get(c)
        if code: freqs['Φ6'][code] += 1

    base_probs = {}
    for param, alts in MOVIE_PARAMS.items():
        total = sum(freqs[param].values()) or 1
        for alt in alts:

```

```

        base_probs[alt] = freqs[param].get(alt, 0) / total
    return base_probs

def step1_personalize_movie(user_profile: list, base_probs: dict, df_user_movie:
pd.DataFrame) -> dict:
    adapted = {}
    for param, alts in MOVIE_PARAMS.items():
        total = 0.0
        temp = {}
        for alt in alts:
            multiplier = 1.0
            for u_alt in user_profile:
                c = cell(df_user_movie, u_alt, alt)
                multiplier *= (1.0 + c)
            temp[alt] = base_probs[alt] * multiplier
            total += temp[alt]
        for alt in alts:
            adapted[alt] = temp[alt] / total if total > 0 else 1.0 / len(alts)
    return adapted

def step2_movie_consistency(adapted_probs: dict, df_movie_internal: pd.DataFrame,
top_k: int = 50) -> list:
    data_alts = [MOVIE_PARAMS[p] for p in sorted(MOVIE_PARAMS.keys())]
    all_configs = list(product(*data_alts))
    records = []
    total_w = 0.0
    for config in all_configs:
        p_prod = float(np.prod([adapted_probs[a] for a in config]))
        c_mult = 1.0
        for a_i, a_j in combinations(config, 2):
            val = cell(df_movie_internal, a_i, a_j)
            if val == 0.0: val = cell(df_movie_internal, a_j, a_i)
            c_mult *= (1.0 + val)
        w = p_prod * c_mult
        if w > 0:
            total_w += w
            records.append((config, w))
    valid = [(cfg, w / total_w) for cfg, w in records]
    valid.sort(key=lambda x: x[14], reverse=True)
    return valid[:top_k]

def step3_synthesize_strategies(valid_configs: list, user_profile: list,
df_movie_strat: pd.DataFrame, df_strat_user: pd.DataFrame) -> dict:
    raw = {}
    all_strat_codes = [c for p in sorted(STRAT_PARAMS.keys()) for c in
STRAT_PARAMS[p]]
    for strat_code in all_strat_codes:

```

```

    eff_movie = 0.0
    for movie_config, p_s in valid_configs:
        mult = 1.0
        for alt in movie_config:
            mult *= (1.0 + cell(df_movie_strat, alt, strat_code))
        eff_movie += p_s * mult
    user_mult = 1.0
    for u_alt in user_profile:
        user_mult *= (1.0 + cell(df_strat_user, strat_code, u_alt))
    raw[strat_code] = eff_movie * user_mult
    return raw

def normalize_by_groups(scores: dict) -> dict:
    groups = defaultdict(list)
    for code in scores:
        grp = code[:3] if code[1:3].isdigit() else code[:2]
        groups[grp].append(code)
    normalized = {}
    for grp, codes in groups.items():
        total = sum(scores[c] for c in codes)
        for c in codes:
            normalized[c] = scores[c] / total if total > 0 else 0.0
    return normalized

def run_sppd(user_profile: list, verbose: bool = True) -> pd.DataFrame:
    df_user_movie = load_csv(PATH_USER_MOVIE)
    df_movie_int = load_csv(PATH_MOVIE_INT)
    df_movie_strat = load_csv(PATH_MOVIE_STRAT)
    df_strat_user = load_csv(PATH_STRAT_USER)
    base_probs = parse_ontology_frequencies(PATH_OWL)
    adapted = step1_personalize_movie(user_profile, base_probs, df_user_movie)
    valid_configs = step2_movie_consistency(adapted, df_movie_int, top_k=50)
    raw_scores = step3_synthesize_strategies(valid_configs, user_profile,
    df_movie_strat, df_strat_user)
    norm_scores = normalize_by_groups(raw_scores)
    records = []
    for param, alts in sorted(STRAT_PARAMS.items()):
        for alt in alts:
            records.append({'Параметр': param, 'Код': alt, 'Альтернатива':
    STRAT_LABELS.get(alt, alt), 'Ймовірність': round(norm_scores.get(alt, 0.0), 4)})
    return pd.DataFrame(records)

if __name__ == "__main__":
    profiles = [{"name": "Підліток-геймер", "profile": ['K1.1', 'K2.1', 'K3.1',
    'K4.1']}]
    for i, p in enumerate(profiles):
        df = run_sppd(p['profile'], verbose=True)

```

ДОДАТОК В ВЕБ-ІНТЕРФЕЙС

```

import streamlit as st
import pandas as pd
import time
import sppr

st.set_page_config(page_title="СППР: Кіноіндустрія", page_icon="🎬", layout="wide")

st.title("СППР: Персоналізація стратегії релізу")
st.markdown("Система підтримки прийняття рішень на основі модифікованого методу морфологічного аналізу")

st.sidebar.header("👤 Профіль глядача")
st.sidebar.markdown("Оберіть характеристики цільової аудиторії:")

selected_profile = []

PARAM_FULL_NAMES = {
    'K1': 'Жанрові вподобання',
    'K2': 'Вік',
    'K3': 'Цифрова активність',
    'K4': 'Платформа перегляду'
}

for param_key, param_alts in sppr.USER_PARAMS.items():
    options = {alt: sppr.USER_LABELS.get(alt, alt) for alt in param_alts}
    param_title = PARAM_FULL_NAMES.get(param_key, param_key)
    selected = st.sidebar.selectbox(
        label=param_title,
        options=list(options.keys()),
        format_func=lambda x, o=options: o[x]
    )
    selected_profile.append(selected)

st.sidebar.divider()

if st.sidebar.button("Розрахувати стратегію", type="primary",
use_container_width=True):
    st.header("Хід розрахунків")
    with st.spinner("Завантаження матриць та онтологічної БЗ..."):
        try:
            df_user_movie = sppr.load_csv(sppr.PATH_USER_MOVIE)
            df_movie_int = sppr.load_csv(sppr.PATH_MOVIE_INT)

```

```

df_movie_strat = sppr.load_csv(sppr.PATH_MOVIE_STRAT)
df_strat_user = sppr.load_csv(sppr.PATH_STRAT_USER)
base_probs = sppr.parse_ontology_frequencies(sppr.PATH_OWL)
time.sleep(0.5)
except Exception as e:
    st.error(f"Помилка завантаження файлів: {e}")
    st.stop()

with st.expander("Крок 1: Адаптація фільму під користувача", expanded=True):
    st.info("Коригування базових частот онтології за допомогою матриці зв'язків  
'Користувач-Фільм'.")
    adapted = sppr.step1_personalize_movie(selected_profile, base_probs,
df_user_movie)
    df_adapted = pd.DataFrame(list(adapted.items()), columns=['Код',
'Ймовірність'])
    df_adapted['Характеристика'] = df_adapted['Код'].map(sppr.MOVIE_LABELS)
    df_adapted = df_adapted.sort_values(by="Ймовірність",
ascending=False).head(10)
    st.dataframe(df_adapted[['Код', 'Характеристика', 'Ймовірність']],
use_container_width=True, hide_index=True)

with st.expander("Крок 2: Внутрішня узгодженість конфігурацій", expanded=True):
    st.info("Відкидання індустріально неможливих комбінацій фільму та пошук  
найімовірніших.")
    valid_configs = sppr.step2_movie_consistency(adapted, df_movie_int, top_k=5)
    config_data = []
    for i, (cfg, prob) in enumerate(valid_configs, 1):
        cfg_names = " + ".join([sppr.MOVIE_LABELS.get(c, c) for c in cfg])
        config_data.append({"#": i, "Конфігурація (Жанр, Режисер, Хронометраж,  
Бюджет, Сюжет, Актори)": cfg_names, "Вара": f"{prob:.5f}"})
    st.table(pd.DataFrame(config_data).set_index("#"))

with st.spinner("Синтез стратегій..."):
    raw_scores = sppr.step3_synthesize_strategies(valid_configs, selected_profile,
df_movie_strat, df_strat_user)
    norm_scores = sppr.normalize_by_groups(raw_scores)
    time.sleep(0.5)

st.divider()
st.header("Фінальний рейтинг маркетингових рішень")

def plot_strategy_group(group_prefix, title, col):
    group_items = {k: v for k, v in norm_scores.items() if
k.startswith(group_prefix)}
    df_g = pd.DataFrame(list(group_items.items()), columns=['Код', 'Ймовірність'])
    df_g['Стратегія'] = df_g['Код'].map(sppr.STRAT_LABELS)
    df_g = df_g.sort_values(by="Ймовірність", ascending=True)

```

```

col.subheader(title)
col.bar_chart(df_g.set_index('Стратегія')['Ймовірність'], horizontal=True,
color="#ff4b4b")

col1, col2 = st.columns(2)
plot_strategy_group("C7", "📌 Тип реклами", col1)
plot_strategy_group("C11", "📄 Формат релізу", col2)
st.markdown("### Деталізовані маркетингові інструменти:")
col3, col4, col5 = st.columns(3)
plot_strategy_group("C8", "Офлайн-інструменти", col3)
plot_strategy_group("C9", "Онлайн-інструменти", col4)
plot_strategy_group("C10", "Нестандартний маркетинг", col5)
st.success("Розрахунок успішно завершено! Спробуйте змінити параметри глядача у
бічному меню, щоб побачити, як зміниться стратегія.")
else:
    st.info("Будь ласка, оберіть характеристики глядача у меню зліва та натисніть
'Розрахувати стратегію'.")

```

ДОДАТОК Г ПРЕЗЕНТАЦІЙНІ МАТЕРІАЛИ

Система підтримки прийняття рішень персоналізації маркетингових стратегій в медіаіндустрії на основі онтологічного моделювання та мережі морфологічних таблиць

Ковальчук О.М. КА-22

Науковий керівник: Савченко І.О.

Мета, об'єкт та предмет дослідження

2

- **Об'єктом дослідження** є процес прийняття рішень щодо персоналізації маркетингових стратегій та формату релізу кінофільмів.
- **Предметом дослідження** є двоетапний модифікований метод морфологічного аналізу з нерівномірним розподілом оцінок та онтологічні моделі.
- **Метою роботи** є розробка системи підтримки прийняття рішень для персоналізації маркетингових стратегій у медіаіндустрії на основі онтологічного моделювання та мережі морфологічних таблиць.

Актуальність

- Кіноіндустрія – сфера з надзвичайно високими фінансовими ризиками та коротким життєвим циклом продукту.
- Традиційні масові рекламні кампанії втрачають ефективність – ринок вимагає персоналізованих стратегій.
- Існуючі СППР (Cinelytic, Gower Street) спираються лише на статистику минулих проєктів.
- Моя робота пропонує інструмент, що поєднує *об'єктивні дані онтології та якісні експертні оцінки і адаптується до профілю конкретного глядача.*



Постановка задачі

Мета – автоматизувати вибір маркетингових стратегій та формату релізу кінофільмів на основі профілю цільового глядача.

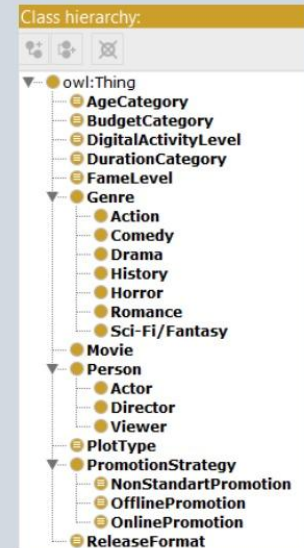
Вимоги до системи:

- побудова онтологічної БЗ з автоматизованим наповненням із відкритих джерел;
- математичне ядро на основі двоетапного МММА з нерівномірним розподілом оцінок;
- обробка матриць взаємної узгодженості між параметрами;
- інтерактивний інтерфейс для введення профілю та генерації рейтингу стратегій.



Онтологічне моделювання

- **Онтологія** – явна специфікація концептуалізації, формалізований машинозчитуваний опис предметної області.
- Архітектура: TBox (ієрархія класів, властивості, обмеження) + ABox (конкретні екземпляри – фільми, режисери).
- Мова формалізації – OWL, базується на дескрипційних логіках.
- Ключова відмінність від реляційних БД: принцип відкритого світу (OWA) – відсутність факту не означає його хибності.



Ієрархія класів OWL-онтології кіноіндустрії (Protégé 5.5)

Двоетапний МММА

Основний об'єкт – морфологічна таблиця: параметри та їхні альтернативи.

Етап 1 (Аналіз сценаріїв): початкові оцінки → матриця узгодженості [-1; 1] → система рівнянь Байєса → уточнені ймовірності.

Етап 2 (Синтез стратегій): очікувана результативність кожної альтернативи з урахуванням ймовірностей сценаріїв.

Шкала Міллера

Якісна характеристика	Кількісна характеристика
Практично неможливо	[0 ÷ 0,1]
Дуже мала ймовірність	[0,1 ÷ 0,25]
Мала ймовірність	[0,25 ÷ 0,4]
Середня ймовірність	[0,4 ÷ 0,6]
Велика ймовірність	[0,6 ÷ 0,75]
Дуже велика ймовірність	[0,75 ÷ 0,9]
Практично гарантовано	[0,9 ÷ 1]

Оцінки для матриць взаємної узгодженості

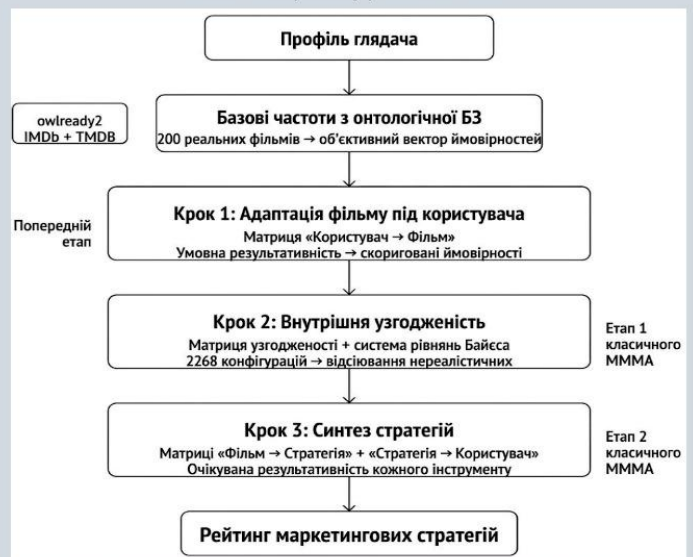
Оцінка	Тлумачення
-1	Альтернативи повністю неузгоджені. Конфігурація неможлива.
(-1; 0)	Неузгоджені. Вибір однієї зменшує ймовірність вибору іншої.
0	Незалежні. Вибір однієї не впливає на вибір іншої.
(0; 1)	Узгоджені. Вибір однієї збільшує ймовірність вибору іншої.
1	Повністю узгоджені. Вибір однієї тягне за собою вибір іншої.

Специфіка та адаптація методу

7

- Оцінки диктуються профілем глядача → суттєво нерівномірні → ризик нереалістичних конфігурацій.
- Для зв'язування профілю з фільмом – матриця «Користувач → Фільм».
- Базові частоти з онтологічної БЗ → збалансовані початкові ймовірності для рівнянь Байєса.

Схема алгоритму роботи СППР



Онтологічна база знань

8

- Структура (TBox): класи Movie, Person (Actor, Director, Viewer), Genre, PromotionStrategy
- Об'єктні властивості: HasAttribute, HasContributor, HasStrategy
- Логічні обмеження: функціональність (hasDuration), транзитивність (isOlderThan), owl:one of
- Наповнення (ABox): Python + owlready2 → 200 фільмів (1990-2026), стратифікована вибірка
- Джерела: datasets.imdbws.com (5 датасетів) + api.themoviedb.org (бюджети)

200 фільмів

7 основних
жанрів

175 режисерів

1699 акторів

Мережа морфологічних таблиць

Морфологічна таблиця профілю користувача

Характеристичні параметри			
Жанрові вподобання	Вік	Цифрова активність	Платформа перегляду
1	2	3	4
1.1 Екшн	2.1 Дорослий (18+)	3.1 Висока	4.1 Кінотеатр
1.2 Комедія	2.2 Підліток (10-17)	3.2 Середня	4.2 Стримінг
1.3 Драма	2.3 Дитина (3-9)	3.3 Низька	4.3 Гібрид
1.4 Фантастика/Фентезі			
1.5 Хоррор			
1.6 Історичний			
1.7 Романтика			

Мережа морфологічних таблиць

Морфологічна таблиця характеристик фільму

Характеристичні параметри					
Жанр	Режисер	Хрон.	Бюджет	Сюжет	Актори
1	2	3	4	5	6
1.1 Екшн	2.1 Відомий	3.1 Короткий	4.1 Великий	5.1 Ориг.	6.1 Відомі
1.2 Комедія	2.2 Маловід.	3.2 Середній	4.2 Середній	5.2 Адаптація	6.2 Наб. поп.
1.3 Драма		3.3 Довгий	4.3 Малий	5.3 Ремейк	6.3 Дебют
1.4 Фантаст./Фентезі					
1.5 Хоррор					
1.6 Історія					
1.7 Романтика					

Мережа морфологічних таблиць

Морфологічна таблиця маркетингових стратегій та релізу

Рішення				
Реклама	Офлайн інструменти	Онлайн інструменти	Нестандартні інструменти	Реліз
7	8	9	10	11
7.1 Офлайн	8.1 Традиційна реклама	9.1 SMM	10.1 Influencer-маркетинг	11.1 Кіно-реліз
7.2 Онлайн	8.2 PR-заходи	9.2 Контент-маркетинг	10.2 Партизанський маркетинг	11.2 Онлайн-ексклюзив
7.3 Нестандартна		9.3 Цифрова реклама	10.3 Гейміфікація	11.3 Гібридний

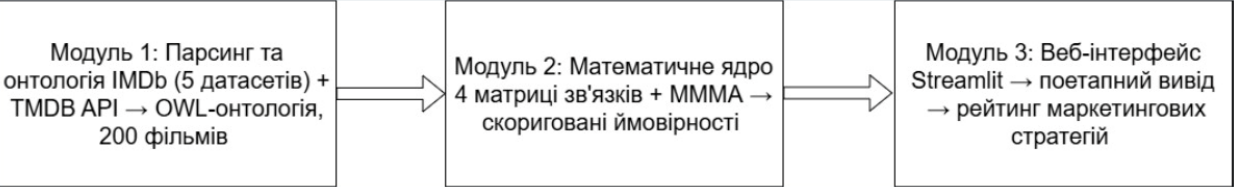
Програмна реалізація

Технологічний стек:

- Python: pandas, numpy (для обчислень);
- Онтологія: owlready2 + OWL-файл;
- Інтерфейс: Streamlit (веб-додаток).

Архітектура (3 модулів):

- Модуль парсингу та наповнення онтології (IMDb + TMDB API);
- Математичне ядро МММА;
- Веб-інтерфейс із поетапним виведенням результатів.



Демонстрація роботи

З 2268 теоретично можливих конфігурацій система відкидає індустріально нереалістичні комбінації через матрицю внутрішньої узгодженості. Нижче – топ-5 найімовірніших конфігурацій фільму для заданого профілю глядача.

✓ Крок 2: Внутрішня узгодженість конфігурацій

Відкидання індустріально неможливих комбінацій фільму та пошук найімовірніших.

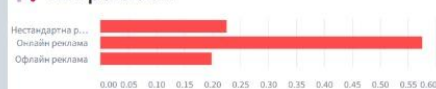
Конфігурація (Жанр, Режисер, Хронометраж, Бюджет, Сюжет, Актори)	Bara
Екшн + Відомий реж. + Середній (90-150 хв) + Великий (30+ млн) + Оригінальний + Відомі актори	0.29652
Фантастика/Фентезі + Відомий реж. + Середній (90-150 хв) + Великий (30+ млн) + Оригінальний + Відомі актори	0.13034
Екшн + Відомий реж. + Середній (90-150 хв) + Середній (2-20 млн) + Оригінальний + Відомі актори	0.09386
Історичний + Відомий реж. + Середній (90-150 хв) + Великий (30+ млн) + Оригінальний + Відомі актори	0.04669
Екшн + Відомий реж. + Довгий (150+ хв) + Великий (30+ млн) + Оригінальний + Відомі актори	0.04412

Демонстрація роботи

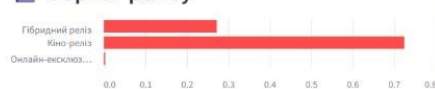
Для кожної реалістичної конфігурації фільму система розраховує очікувану результативність кожного маркетингового інструменту через матрицю «Фільм → Стратегія». Результат додатково коригується матрицею «Стратегія → Користувач» під цифрову активність та платформу перегляду глядача. На виході система формує рейтинг усіх маркетингових рішень у розрізі 5 груп.

Фінальний рейтинг маркетингових рішень

Тип реклами

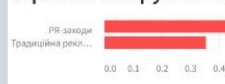


Формат релізу

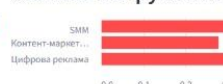


Деталізовані маркетингові інструменти:

Офлайн-інструменти



Онлайн-інструменти



Нестандартний маркетинг



Аналіз результатів

17

Формат релізу: Кіно (0.72) – беззаперечний лідер завдяки вибору кінотеатру як платформи.

Тип реклами: Онлайн (0.57) – висока цифрова активність глядача визначила пріоритет.

Онлайн-інструменти: Цифрова реклама (0.42) та SMM (0.38) – найефективніші канали для даного профілю.

Система підтвердила очікувану логіку: кінотеатральний глядач з високою цифровою активністю → ексклюзивний кіно-реліз + цифрові канали комунікації.

Група рішень	Маркетинговий інструмент	Очікувана результативність
Тип реклами	Онлайн реклама	0.57
	Нестандартна реклама	0.23
	Офлайн реклама	0.20
Формат релізу	Кіно	0.72
	Гібрид	0.28
	Онлайн	0.00
Офлайн-інструменти	PR-заходи	0.65
	Традиційна реклама	0.35
Онлайн-інструменти	Цифрова реклама	0.42
	SMM	0.38
	Контент-маркетинг	0.20
Нестандартний маркетинг	Influencer	0.40
	Партизанський	0.37
	Гейміфікація	0.23

Висновки

18

Досліджено предметну область медіаіндустрії – виявлено недостатність існуючих СППР для роботи з якісними експертними оцінками.

Спроектовано OWL-онтологію та автоматизовано її наповнення стратифікованою вибіркою з 200 реальних фільмів (IMDb + TMDB API).

Розроблено мережу з 3 морфологічних таблиць та 4 матриць зв'язків, що охоплює профіль глядача, характеристики фільму та маркетингові рішення.

Реалізовано двоетапний МММА з нерівномірним розподілом оцінок — відсіювання нереалістичних конфігурацій з 2268 можливих.

Розроблено веб-застосунок Streamlit – тестування підтвердило логічну відповідність рекомендацій профілю глядача.

Подальший розвиток

Розширення онтологічної бази знань новими параметрами – регіональна специфіка ринку, сезонність, конкурентне середовище – для підвищення точності початкових частот.

Інтеграція алгоритмів автоматичного збору ринкових даних у реальному часі для динамічного оновлення матриць узгодженості без участі експерта.

Автоматизація формування експертних оцінок на основі текстового аналізу новин та соціальних медіа.

Розширення системи на інші сегменти медіаіндустрії – музичні релізи, серіальне виробництво, стримінгові оригінали.

Дякую за увагу!