

Міністерство освіти і науки України
Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА
за спеціальністю 123 «Комп'ютерна інженерія»

Тема наукової роботи: «Методи та засоби рекомендації товару для покупців у
інтернет-магазині»

Виконав	_____	І.О. Коржов
Керівник роботи	_____	Ю.О. Кумченко
Консультант	_____	В.Г. Саяпін
Нормоконтроль	_____	Д. І. Кузнєцов
Завідувач кафедри	_____	А. І. Купін

Кривий Ріг
2022

Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

Ступінь вищої освіти
Спеціальність

бакалавр
123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри, голова циклової комісії

_____ А. І. Купін

“ ____ ” _____ 20__ року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

_____ (прізвище, ім'я, по батькові)

1. Тема роботи _____

керівник роботи _____,

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “ ____ ” _____ 20__ року №__

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи _____

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) _____

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк виконання етапів роботи	Примітка

Студент _____
(підпис) (прізвище та ініціали)

Керівник роботи _____
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка: 102 сторінок, 47 рисунків, таблиці, 1 додаток, 38 використаних джерела.

Об'єкт дослідження – процес пропозиції подібних товарів у інтернет-магазинах.

Робота складається з трьох розділів.

Перший розділ присвячено аналізу предметної області електронної комерції та проблемам пропозиції товарів. За результатом проведеного дослідження виявлено, що вагому роль при просуванні товарів в інтернет-магазинах відіграє інструмент пропозиції товарів, який збільшує імовірність відповідної зміни фокусу уваги клієнта.

Другий розділ присвячено розробці інформаційної модель інтернет-магазину з адаптивною пропозицією товарів, яка містить формальне подання всіх необхідних сутностей для адаптивної пропозиції товарів за семантичними ознаками користувачам інтернет-магазину.

Третій розділ включає в себе проектування системи мікросервісів, які отримують та зберігають дані, виконують запити на тренування моделі та надають готові рекомендації.

Ключові слова: інтернет-магазин, ключові слова, пропозиція товарів, дисперсійне оцінювання, семантичний аналіз, семантика, профіль товару, інформаційна система.

					КНУ.РМ.123.22.09.Р			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Коржов				РЕФЕРАТ		Літера	Аркуш
Перевірив	Кумченко							
Н.контроль	Кузнєцов						КІ-21м	
Затвердив	Купін							

ABSTRACT

Explanatory note: 102 pages, 47 figures, tables, 1 appendix, 38 used sources.

The object of the study is the process of offering similar products in online stores

The work consists of three sections.

The first chapter is devoted to the analysis of the subject area of electronic commerce and the problems of the supply of goods. According to the results of the conducted research, it was found that the product offer tool plays an important role in the promotion of goods in online stores, which increases the probability of a corresponding change in the focus of the client's attention.

The second section is devoted to the development of an information model of an online store with an adaptive offer of goods, which contains a formal presentation of all the necessary entities for an adaptive offer of goods according to semantic features to users of an online store.

The third section includes the design of a system of microservices that receive and store data, execute model training requests, and provide ready-made recommendations.

Keywords: online store, keywords, product offer, variance estimation, semantic analysis, semantics, product profile, information system.

					KHU.PM.123.22.09.P	Арк.
	Арк.	№ документа	Підпис	Дата		

6
ЗМІСТ

РЕФЕРАТ	4
ПЕРЕЛІК СКОРОЧЕНЬ.....	7
ВСТУП	8
1. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ ТА ПОРІВНЯННЯ ПІДХОДІВ ДО РЕАЛІЗАЦІЇ РЕКОМЕНДАЦІЙНИХ СИСТЕМ.....	10
1.1Визначення рекомендаційної системи	10
1.2Використання систем рекомендацій в електронній комерції	11
1.3Вплив РС на аудиторію користувачів	12
1.4Функції рекомендаційних систем	13
1.5Основні поняття в рекомендаційних системах	14
1.6Дані та джерела інформації для рекомендаційних систем.....	15
1.7Підходи до реалізації рекомендаційних систем.....	24
<i>Висновки за розділом</i>	<i>36</i>
2. АЛЬТЕРНАТИВНІ ПІДХОДИ ДО РЕАЛІЗАЦІЇ РЕКОМЕНДАЦІЙНИХ СИСТЕМ....	37
2.1Гібридні рекомендаційні системи	37
2.2Рекомендації на типовому веб-сайті електронної комерції.....	39
2.3Метрики рекомендаційних систем	59
2.4Рекомендаційна система на основі алгоритму ССО (Correlated CrossOccurrence) ..	63
2.5Інформаційна модель інтернет-магазину з адаптивною пропозицією товарів	64
2.6Схема адаптивної пропозиції товарів у інтернетмагазині за семантичними ознаками	65
2.7Функції інформаційної системи автоматизованого пошуку ключових семантичних одиниць у цифрових текстах.....	67
2.8Математичні моделі обрахунку сумарного лінгвістичного збігу та сумарного семантичного збігу описів профілів товарів	69
<i>Висновки за розділом</i>	<i>70</i>
3. РЕКОМЕНДАЦІЙНИЙ СЕРВІС ДЛЯ ЕЛЕКТРОННОЇ КОМЕРЦІЇ	71
3.1Опис технологій, що використовуються	71
3.2Архітектура програмного продукту	75
3.3 Тестування з набором даних поведінки користувача на веб-сайті електронної комерції	81
3.4 Дослідження методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками	85
<i>Висновки за розділом</i>	<i>87</i>
ВИСНОВКИ.....	88
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	89
Додаток А.....	92

					КНУ.РМ123.22.09.3			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Коржов.				ЗМІСТ		Літера	Аркуш
Перевірив	Кумченко							
Н.контроль	Кузнєцов						КІ-21м	
Затвердив	Купін							

ПЕРЕЛІК СКОРОЧЕНЬ

РС – рекомендаційна система;
 ГРС – гібридна рекомендаційна система;
 CF – алгоритм колаборативного фільтрування (Collaborative filtering);
 CBF – алгоритм фільтрування за контентом (Content based filtering);
 ССО – алгоритм колаборативного фільтрування для багатьох індикаторів, що враховує кореляцію спільності входження (Correlated Cross-Occurrence);
 LLR – алгоритм статичної перевірки відношенням правдоподібностей (Log Likelihood Ratio).

					КНУ.РМ.123.22.09.ПС	Арк.
	Арк.	№ документа	Підпис	Дата		

Кожного дня людство створює щось нове. Близько 90% усієї наявної інформації було створено за 2 останні роки. Наші винаходи розширили фізичні можливості та збільшили свободу в просторі та часі. Кожну хвилину у світі відбувається 154.200 дзвінків у Skype, надсилається 16 мільйонів текстових повідомлень та 103.447.520 спаму у вигляді email[7].

За останнє десятиліття став більш зв'язаним та оптимізованим. Сьогодні сфера будь-якого комерційного бізнесу використовує Інтернет як джерело зручної інформації для клієнтів, для комунікаційних та рекламних цілей. Але компаніям складно самостійно оцінити нові можливості мережі. Тенденція збільшення кількості користувачів та продуктів у сфері електронної комерції, полегшення масштабування ринків, все ще має низку проблем. Серед них такі:

- пошук найкращого кандидата продукту для користувача, який найбільш точно відповідає його потребі;
- збільшення конверсії продажів продукції;
- поліпшення досвіду користування послугою електронної комерції, заохочуючи клієнта використовувати сервіс у майбутньому.

Сучасні умови економічної конкуренції вимагають від підприємців створення все нових і нових технологій привабливості покупців. Однією з таких технологій є рекомендаційні системи. Дані системи принесли нові способи взаємодії звичайних веб-сайтів зі своїми користувачами. На заміну надання статичної інформації, коли потенційні покупці шукають і, ймовірно, купують товари, рекомендаційні системи збільшують ступінь інтерактивності, а також розширюють надані користувачу можливості. Рекомендаційні системи формують рекомендації незалежно для кожного конкретного користувача на основі його минулих покупок і пошуків, а також на основі запитів інших користувачів. Також такі рекомендації збільшують чек клієнтів, завдяки пропозиції суміжних товарів або товарів, які якимось пов'язані з його минулими придбаннями. Існує безліч принципів проектування рекомендаційних систем, деякі з них будуть розібрані в даній роботі.

Завдання дослідження:

- Порівняти існуючі підходи до формування рекомендацій
- Визначити дані які є найбільш репрезентативними показниками вподобань користувача
- Проаналізувати способи відображення рекомендацій
- Спроекувати систему мікросервісів, які будуть зберігати дані та надавати рекомендації
- Розробити РС, що буде надавати рекомендації для користувача базуючись на його «неявних» відгуках – переглядах, додавання в кошик, та ін.

Предметом дослідження є програмні рішення для систем рекомендацій для електронної комерції та набір їх функціональних можливостей, які

забезпечують ефективне вирішення проблеми відповідності продукту та користувача.

Об'єкт дослідження – процес пропозиції подібних товарів у інтернетмагазинах.

У галузі електронної комерції рекомендаційні системи ґрунтуються на використанні даних про поведінку користувачів у системі. У більшості випадків явних відгуків користувачів немає. Системи збирають інформацію про те, які продукти переглядав користувач, які товари були поміщені в кошик та які товари користувач придбав. Це все є неявні відгуки, оскільки користувач не виконує ніяких додаткових дій у системі, за винятком того, що він зацікавлений. Цей вид даних називається неявним відгуком, а ці огляди поділяються на багато типів: покупка товару, перегляд інформації про товар тощо. Більшість реалізацій РС не використовують усі доступні типи неявних зворотних зв'язків або не використовують найбільшу точність, втрачаючи якість рекомендацій.

Метою даної роботи є впровадження багатofакторної масштабованої системи рекомендацій для електронної комерції, яка працює на неявних даних відповідей і здатна надавати релевантні рекомендації користувачам.

Методи дослідження, застосовані для вирішення поставлених завдань: для розв'язання поставлених задач використовуються основні положення методів аналізу даних й теорії множин, для реалізації інформаційної системи – методології проектування інформаційних систем.

Наукова новизна одержаних результатів. Було вдосконалено метод адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками, що на відміну від існуючих дозволяє за значеннями оцінки сумарного лінгвістичного збігу профілів товарів, обрахованої за порівняннями кількостей появ ідентичних слів, та оцінки сумарного семантичного збігу профілів товарів, обрахованої з використанням дисперсійного оцінювання позицій появ слів, виконувати формування множин подібних товарів відповідно до обраного користувачем товару.

Практичне значення одержаних результатів. За результатом прикладного дослідження виявлено, що значний ефект від застосування методу спостерігається у випадках, коли в інтернет-магазині додано нові товари чи змінено властивості існуючих товарів, відповідно найбільша ефективність визначається тоді, коли взагалі відсутня статистика активності користувачів щодо товарів. Це суттєво покращує якість пропозицій, на відміну від традиційних підходів у таких випадках, які полягають в пропозиції товарів з обраної категорії, пропозиції товарів із складеного вручну переліку або пропозиції випадкових товарів з асортименту. У випадку ж, коли в інтернет-магазині наявна статистика активності користувачів щодо товарів, використання методу дозволяє покращити можливості інтернет-магазину з пропозицій користувачам подібних товарів шляхом підбору товарів із семантично подібними описами властивостей.

1. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ ТА ПОРІВНЯННЯ ПІДХОДІВ ДО РЕАЛІЗАЦІЇ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

1.1 Визначення рекомендаційної системи

Фільтрування інформації - це процес моніторингу великих обсягів динамічно генерованих даних та виявлення лише того набору цих даних для користувача, який найбільш точно відповідає його потребам.

Інтерфейси систем фільтрування інформації розділяють на: фільтрувальні (filtering interfaces) та рекомендаційні (recommendation interfaces). Прикладом інтерфейсу фільтрування є фільтри електронної пошти, які розподіляють вхідні електронні листи в різні групи за конкретними атрибутами, що дозволяє користувачеві легко знаходити важливі електронні листи. Прикладом систем з рекомендаційним інтерфейсом є системи з вихідним інтерфейсом списків топ-N та списків продуктів. Елементи в цьому списку відсортовані за зменшенням коефіцієнта рекомендації товару[1].

Рекомендаційна система (РС) – це програмне забезпечення, яке, аналізуючи дані користувача, пропонує йому персональний продукт, який, ймовірно, покриває його потреби. Основною функцією РС є фільтрація певної підмножини інформації [2].

З іншого боку, завдання системи рекомендацій - це завдання сфери машинного навчання. У машинному навчанні більшість завдань передбачає навчання функції, яка буде обробляти дані. Для створення такої функції необхідний набір аргументів і значення функції від цих аргументів. В ідеалі, чим більший набір даних, тим точнішою повинна бути функція, але на практиці ми стикаємося з такими проблемами, як шум даних та неправильно підібраний набір аргументів, що спричиняє проблеми перетренування (overfitting) та недотренування (underfitting). Ці проблеми до кінця не вирішені в рекомендаційних системах. На рисунку. 1.1 можна побачити принцип функціонування рекомендаційної системи.

					КНУ.РМ.123.22.09.01.ОІРППРРС			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Коржов.				ОГЛЯД ІСНУЮЧИХ РІШЕНЬ ТА ПОРІВНЯННЯ ПІДХОДІВ ДО РЕАЛІЗАЦІЇ РЕКОМЕНДАЦІЙНИХ СИСТЕМ	Літера	Аркуш	Аркушів
Перевірив	Кумченко							
Н.контроль	Кузнєцов					КІ-21м		
Затвердив	Купін							



Рисунок. 1.1 Принцип роботи РС

Користувач залишає інформацію явно або неявно, система фільтрує і зберігає її. Збережені дані використовуються алгоритмами машинного навчання для тренування моделей, використовуючи які можна передбачити успішні рекомендації для користувачів.

1.2 Використання систем рекомендацій в електронній комерції

Системи рекомендацій є одним із основних способів, за допомогою яких вебсайти електронної комерції отримують постійних клієнтів. Повторна покупка – один з ключових показників для онлайн-магазинів, оскільки повторний клієнт означає менше грошей, витрачених на те, щоб спонукати його/її до покупки.

Системи рекомендацій – це не нова технологія. Вони з'явилися разом з інтернет-магазинами та стрімінговими сервісами. Сфери застосування рекомендаційних систем різноманітні: пошук фільмів, музики, наукових статей, роздрібна торгівля, соціальні мережі, електронна комерція, онлайн банкінг і т.д.

Мабуть, одним з найбільш широко відомих прикладів впровадження і використання рекомендаційних систем є компанія Netflix – постачальник відеоконтенту на умовах оренди і у вигляді потокового сервісу[5].

Компанія починала з того, що розсилала клієнтам по підписці VHS-касети і DVD. Користувач дивився і відправляв диски назад, отримував нові. Для Netflix було важливо підвищити якість рекомендацій. Чим краще Netflix рекомендує користувачам фільми, тим більше фільмів беруть в прокат. Відповідно, зростає і прибуток компанії.

У 2006 році компанія Netflix оголосила конкурс на вдосконалення своєї рекомендаційної системи під назвою Cinematch. Рекомендації формувалися з урахуванням як оцінок користувача, так і оцінок інших глядачів - для цього система підбирала користувачів зі схожими перевагами, чії оцінки близькі до їх власним. На підставі цього глядачеві автоматично давалася рекомендація: подивитися той чи інший фільм. Власний алгоритм Netflix передбачав оцінок з

якістю 0.9514 за метрикою RMSE. Завдання було поліпшити прогноз хоча б на 10% - до 0.8563. Переможцю обіцяли приз в \$ 1 000 000.

Netflix виклав у відкритий доступ зібрані дані: близько 100 мільйонів оцінок за п'ятибальною шкалою з зазначенням ID користувачів, які поставили оцінку. Учасники змагання повинні були якомога точніше передбачати, яку оцінку поставити конкретному фільму той чи інший користувач.

Змагання тривало три роки. За перший рік якість поліпшили на 7%, далі все трохи сповільнилося. Проміжні номінації вручалися щороку до тих пір, поки дві команди з невеликою різницею в часі надіслали рішення, кожне з яких проходило поріг в 10%. Перший приз дістався компанії BellKor's Pragmatic Chaos, групі вчених з AT & T, яким вдалося домогтися поліпшення точності рекомендацій на 10,06%.

Компанії продають книги, підписки на музику, фільми, електроніку, побутові товари, автомобілі, нерухомість та багато інших товарів. Також, з'явилися такі платформи для онлайн навчання як Coursera, Udemy, Prometheus. Ці платформи пропонують велике різноманіття курсів на різні тематики, як платних так і безкоштовних. Очевидно, що кількість альтернатив продуктів, які може придбати покупець на веб-сайтах E-Commerce зростає. Сервіси Amazon, Alibaba пропонують сотні різних товарів, сучасні технології дозволяють обробляти набагато більшу їх кількість. При такому різноманітті користувачу складно підбирати підходящі товари, адже проаналізувати кожен з них фізично неможливо. Такі сервіси мають фільтри пошуку, що дозволяють шукати об'єкти за їхніми атрибутами, але не всі товари мають чітко-виділені атрибути. Наприклад, книги та музика. Добре спроектована рекомендаційна система вирішує цю проблему.

Рекомендацій в сфері електронної комерції представляють собою списки об'єктів виду top-N. Чим ближче до голови списку розміщений елемент, що рекомендується, тим більш релевантним він вважається для користувача[5].

На сьогоднішній день майже кожен інтернет-магазин використовує який-небудь тип рекомендацій, що і не дивно, оскільки ці системи можуть суттєво підвищити доходи, CTR, конверсії та інші важливі показники, якщо РС належним чином налаштована. Крім того, вони можуть мати значний позитивний ефект і на взаємодію з користувачем, що перекладається на показники, які важче виміряти, але, тим не менш, ці показники мають надзвичайно важливе значення для онлайн-бізнесу, наприклад, задоволення клієнтів та відсоток користувачів, які повертаються.

1.3 Вплив РС на аудиторію користувачів

Не потрібно проводити детальне дослідження ринку щоб побачити що клієнт буде більш готовий зробити покупку в магазині у якому він відчуває максимальну допомогу у пошуку потрібного продукту. Покупці також набагато частіше повертаються до такого магазину в майбутньому.

					КНУ.РМ.123.22.09.01.ОІРППРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

Кілька місяців тому Netflix підрахував, що його рекомендаційний двигун коштує 1 мільярд доларів США для компанії щорічно [5]. В даний час використання РС є абсолютно необхідною умовою успішного ведення бізнесу в Інтернеті. Найвідоміший приклад - статистичні дані Amazon, за якими 35% їх загальних доходів надходить від покупки продуктів, які клієнти знайшли за допомогою рекомендацій. Це дуже важливий показник при оцінці системи рекомендацій. Компанії не завжди враховують його. Нижче наведено кілька показників, які можуть описувати ефективність рекомендацій:

Коефіцієнт конверсії за рекомендаціями - показник переходів для користувачів, які натискали рекомендації.

CTR - відношення числа кліків на рекомендований продукт до числа його показів. Вимірюється у відсотках. Формула розрахунку: $CTR = (\text{число кліків} / \text{число показів}) \times 100 \%$. Наприклад, якщо рекламне оголошення було показано 100 раз, а клікнули по ньому лише один раз, то показник CTR дорівнюватиме 1 %. Хоча це і очевидний, але дуже важливий показник, який слід враховувати.

% доходу отриманий від рекомендацій - це один з найбільш часто використовуваних показників (також висвітлених в прикладі Amazon вище). Це просто означає доходи через рекомендації / сукупні доходи.

Кількість переглянутих продуктів - кількість продуктів, які переглядають люди, які активно використовують рекомендації під час своїх сеансів. Хоча інтуїтивно більша кількість переглядів може також означати, що користувачу важко знайти потрібне, у дослідженні Wolfgang Digital прийшли до висновку, що час, витрачений на сайті користувачем, і кількість сторінок, які вони відкривають, корелюють з конверсією позитивно.[4]

1.4 Функції рекомендаційних систем

РС займають важливе місце в багатьох бізнес процесах. Саме тому багато великих компаній фінансово підтримують розвиток цієї сфери.

Серед основних функцій РС можна виділити наступні:

- *Збільшення кількості проданих товарів.* Це, ймовірно, найважливіша функція для комерційних РС, тобто, збільшити коефіцієнт продажу товарів за рахунок обігу рекомендованого товару у порівнянні з коефіцієнтом продажу без будь-якої рекомендації. Ці додаткові продажі досягаються тому, що рекомендовані елементи задовольняють потреби користувачів. Імовірно, користувач визнає це після того, як спробував кілька рекомендацій. Загалом, можна сказати, що з точки зору провайдера, основною метою введення РС є збільшення кількості конверсій (conversion rate), тобто, кількості користувачів, які приймають до уваги цю рекомендацію і споживають рекомендований продукт.

- *Продаж більшої кількості специфічних предметів.* Ще однією важливою функцією РС є продаж предметів, які складно знайти без точної рекомендації через їхні унікальні характеристики, які до вподоби невеликій підмножині користувачів. Наприклад, у РС фільмів компанії Netflix,

постачальник послуг зацікавлений в оренді усіх DVD-дисків в каталозі, а не тільки найбільш популярних з них. Цього важко досягнути без РС, оскільки постачальник послуг не може рекламувати фільми, які, ймовірно, влаштовують смак невеликої групи користувачів. Таким чином, РС пропонує рекомендації непопулярних фільмів потрібним користувачам.

- *Підвищення задоволеності користувачів.* Добре розроблена РС може також поліпшити досвід користувача з сайтом або додатком. Поєднання ефективних, тобто точних рекомендацій збільшить суб'єктивну оцінку системи користувачем. Це, в свою чергу, збільшить використання системи та ймовірність того, що рекомендації будуть прийняті до уваги.

- *Збільшення довіри користувача.* Користувачі лояльні до веб-сайтів які пам'ятають попередню активність цього користувача. Це нормальна особливість РС оскільки багато алгоритмів у РС вираховують рекомендації, посилюючись на інформацію, отриману від користувача в попередніх взаємодіях, наприклад, його попередні рейтинги. Отже, чим довше користувач взаємодіє з сайтом, тим більш витонченою та персоналізованою для користувача стає модель [1].

1.5 Основні поняття в рекомендаційних системах

Користувач взаємодіє з системою через її *інтерфейс*, яких поділяється на *вхідний* та *вихідний*.

Вхідним інтерфейсом рекомендаційної системи називається сукупність точок взаємодії між користувачем і системою, через які система отримує персоналізовані дані користувача. Введемо поняття рейтингу. *Рейтинг* – якісна оцінка продукту користувачем, що, зазвичай, виражається числовим значенням. Наприклад, користувач може залишити оцінку відео, яке він переглянув, або залишити коментар під продуктом який він купив. Система отримує персональну інформацію від користувача двома способами: *явним* (explicit) і *неявним* (implicit). Вище наведені приклади *явного рейтингу*, адже користувач ціленаправлено залишає оцінку, або коментар. Прикладом *неявного рейтингу* може послугувати ситуація, коли користувач почав дивитися відео і закінчив перегляд за хвилину. Звідси система може зрозуміти, що відео користувачу не сподобалося, адже він завершив перегляд за хвилину.[6]

Вихідним інтерфейсом рекомендаційної системи називається сукупність точок взаємодії між користувачем і системою, через які користувач отримує рекомендації від системи.

Існує поняття *передбаченого рейтингу*, що визначає з певною ймовірністю передбачену оцінку деякого продукту користувачем.

Рекомендацією називають кінцевий набір предметів, який з певною ймовірністю буде до вподоби користувачу. Цей набір може бути відсортований у порядку спаду рейтингу.

1.6 Дані та джерела інформації для рекомендаційних систем

1.6.1 Типи даних

РС використовують дані різних типів, але їх можливо узагальнити до наступного вигляду: продукти та користувачі.

Продукти є предметами, які рекомендуються. Продукти можуть бути охарактеризованими їхніми властивостями та користю. Наприклад, одяг можливо поділити чоловічий та жіночий, він має колір та розмір. З іншої сторони користувач може бути задоволений, чи розчарований купленим одягом. Це також є характеристикою продукту. РС використовують ці характеристики по різному залежно від алгоритму. Найчастіше, дані про властивості продукту використовуються алгоритмами фільтрування за контентом (CBF), та алгоритмами кластеризації. Властивості продукту називають його ознаками, або опціями. Обов'язковим для продукту являється його ідентифікатор[6].

Користувач є об'єктом рекомендацій. Для того, щоб ці рекомендації були персоналізованими користувач має деякий профіль. Профіль користувача може містити різноманітну інформацію, як, наприклад, вік та стать. Профілем користувача вважається також список рейтингів, з якими він оцінив деякі продукти. Остання характеристика використовується алгоритмом колаборативного фільтрування (CF). Обов'язковим для користувача є його ідентифікатор. На рисунку. 1.2 зображено відношення між користувачами та продуктами.

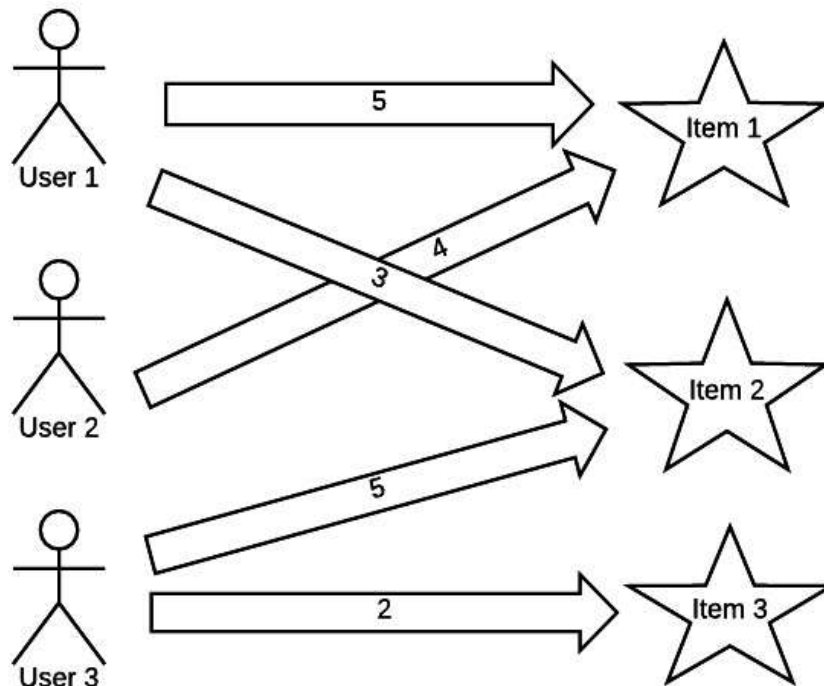


Рисунок. 1.2. Відношення між користувачами та продуктами

Користувачі купують продукти і залишають явний відгук про покупки за п'ятибальною шкалою. Користувач 1 оцінив продукт 1 та 2 з рейтингом 5 і 3 відповідно. Користувач 1 оцінив продукт 1 на 4. Користувач 3 оцінив продукт 2 та 3 на 5 та 2 відповідно. Ці дані рекомендаційна система використовує для підрахунку рекомендацій продуктів, які користувач ще не купував.

1.6.2 Джерела інформації в рекомендаційних системах

Дані, отримані для використання у рекомендаційних системах розділяють на *явні* та *неявні*. Явні дані збираються шляхом формування запитань для користувача, на які від дає відповіді. Це може бути його ініціатива, наприклад користувач залишає коментар, або оцінку продукту. Неявні дані збираються системою без потреби посередніх дій користувачем. Такі дані формуються поведінкою користувача на сервісі.

РС черпає явні дані для тренування моделей з набору рейтингів, характеристик продуктів та профілю користувачів. Рейтинги задаються у форматі, зображеному таблицею 1.1.

Таблиця 1.1 Список рейтингів до рисунку 1.2

Ідентифікатор користувача	Ідентифікатор продукту	Рейтинг
1	1	5.0
1	2	3.0
2	1	4.0
3	2	5.0
3	3	2.0

Зазвичай рейтинги використовуються алгоритмами колаборативного фільтрування та алгоритмами підрахунку неперсоналізованих рекомендацій. Продукт та його характеристики також несе важливу інформацію. Характеристики продукту, зазвичай, використовуються алгоритмами фільтрування за контентом (CBF) та алгоритмами кластеризації. Останні дають змогу підрахувати рекомендації групи продуктів, які має сенс купувати разом. Саме за допомогою цього алгоритму прославилася РС Amazon. Профіль користувача також може покращити точність рекомендацій. Наприклад, у випадку з одягом, інформація про стать користувача є надзвичайно важливою, адже це дасть змогу уникнути помилки рекомендації одягу для жінок чоловікам і навпаки.

Через те, що явних даних занадто мало по відношенню до кількості користувачів та продуктів сучасні підходи у більшості випадків

					КНУ.РМ.123.22.09.01.ОІРППРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

використовують неявні дані. Вони задаються у вигляді журналу подій користувача на сервісі. Наприклад, користувач переглянув інформацію про продукт, користувач поклав продукт до кошика, користувач придбав продукт.

1.6.3 Аналіз ресурсів аналогічної тематики

Загальний відсоток продажів в інтернет-магазинах стрімко зростає. Сьогодні набагато простіше витратити 10 хвилин часу, вибрати те, що необхідно, і замовити доставку на будинок, ніж вийти в магазин і простояти в черзі на касі. При цьому купити в інтернет-магазинах можна абсолютно все: від коробки сірників до запчастин для кораблів-криголамів. Існують магазини різних масштабів – найбільші онлайн-портали *Amazon*, *Aliexpress*, *Ebay*, популярні інтернет-магазини в рамках країни (та ж *Rozetka*) або невеликі локальні магазинчики, які прагнуть до слави.

Створити хороший інтернет-магазин досить складно. Можна взяти ношу на себе і спробувати свої сили в онлайн-конструкторі, ризикую допустити ряд критичних помилок. Можна інвестувати в проект і замовити розробку онлайн-магазину в вебстудії, отримавши якісний сайт. Так чи інакше, успіх сайту і, отже, бізнесу, залежить від ряду факторів. І тут починається найцікавіше, адже більшість інтернет-магазинів закриваються вже через рік, не в силах витримати конкуренцію. Чому так відбувається? Чому деякі магазини виходять в дамки, обдаровуючи своїх власників приголомшливою конверсією і прибутком, а інші забуваються, практично, відразу після їх відкриття.

Як вже було зазначено вище, успіх інтернет-магазину залежить від ряду факторів, які, умовно, можна розділити на дві групи [4]:

- якість інтернет-магазину з точки зору його відповідності технічним і призначеним для користувача вимогам;
- якість ведення бізнесу, можливості самої компанії і т.д.

Розглянемо приклади інтернет-магазинів, що займається реалізацією мобільної техніки.

Citrus.ua

Інтернет-магазин електронної техніки “Цитрус” продає величезну кількість товарів, але при цьому його структура і дизайн повністю зрозумілі для користувача. Тут складно загубитися - правильна розбивка на категорії підкатегорії на інтуїтивному рівні дозволяє швидко знайти те, що необхідно (рис. 1.1).



Рис. 1.3. Магазин “Цитрус”

Але окремої уваги заслуговує картка товару, яка являє собою повноцінний *landing page* (рис. 1.4).



Рис. 1.4. Картка товару магазину “Цитрус”

Тут продуманий кожен елемент, в кожному блоці є заклик до дії або тригер, який мотивує до покупки. Функціонал картки приголомшує – тут все в одному місці, не потрібно гортати сторінки для вибору відповідного варіанту моделі.

Сторінка ідеально зверстана, є вся інформація про продукт. Дуже зручно, просто і зрозуміло. А в поєднанні із закликами до дії досягається результат, заради якого і створюється продає сторінка.



Рис. 1.5. Сторінка інформації про певні додаткові функції

На окрему увагу заслуговує опис продукції. Воно взято з офіційного сайту продукту і дуже вигідно оформлено прямо тут на сторінці магазину. Яскраві і красиві фото лише додають ефектності (рис. 1.6).



Рис. 1.6 Інформація про телефон, магазин “Цитрус”

Переглянувши таку картку товару, у цільового відвідувача не залишається ні найменшого шансу піти з сайту почитати щось ще. І все це завдяки приголомшливому дизайну і функціоналу.

Магазин F.ua

Ще один приклад відмінного українського інтернет-магазину електронної техніки. Дизайн сайту досить простий. Тут немає нічого незвичайного і відволікаючого уваги. Лише яскраві банери додають нових фарб (рис. 1.7).

					КНУ.РМ.123.22.09.01.ОІРППРРС	Арк.
Арк.	№ документа	Підпис	Дата			



Рис. 1.7. Головна сторінка магазину “F.ua”

Але простий і зрозумілий дизайн – лише плюс для великих сайтів з тисячами сторінок. Та й не тільки він став перевагою інтернет-магазину. Функціональні можливості сайту просто приголомшливі.

При виборі будь-якої категорії з правого боку відкривається фільтр з величезним переліком варіантів параметрів, частина з яких заточені під SEO (рис. 1.8).



Рис. 1.8. Сторінка фільтрації магазину “F.ua”

Можна знайти ідеальну модель телефону, відсортувавши його за ціною, розміром діагоналі екрана, виробнику, кольором, при цьому вказавши, що модель потрібна для роботи або в якості подарунка для дівчини. Такий фільтр максимально персоналізує покупку, дозволяючи споживачеві знайти саме те, що потрібно.

Але на цьому переваги інтернет-магазину не закінчуються. Переходимо в картку товару (рис. 1.9).

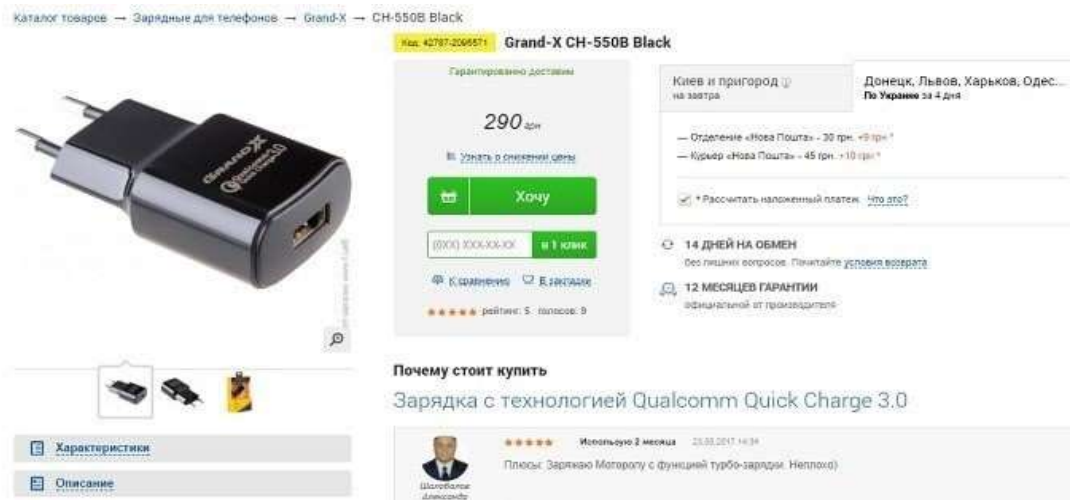


Рис. 1.9. Сторінка покупки товару в магазині “F.ua”

Картка досить проста і зручна. Є все необхідне з точки зору споживчого зручності. Фішка полягає в тому, що з правого боку нам відкривається зображення товару з посиланнями на його опис та характеристики. У міру пересування між об'єктами фото з посиланнями переслідує користувача і це відмінне рішення (рисю 1.10).

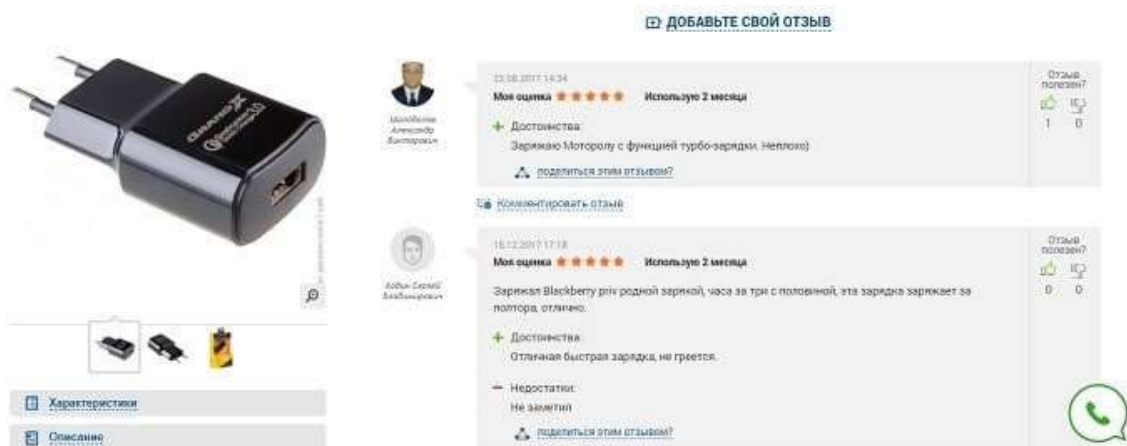


Рис. 1.10. Сторінка картки в магазині “F.ua”

Товар завжди на виду і можна швидко перейти до інформації про нього, що дуже зручно, особливо, якщо сторінка довга. До того ж, на картці товару відразу кілька лідогенерующих кнопок. Одна, за традицією, вгорі сторінці, а друга завжди присутня близько фотографії справа, що дозволяє відвідувачеві зробити замовлення в той момент, коли йому цього захочеться без необхідності скролити сторінку вгору.

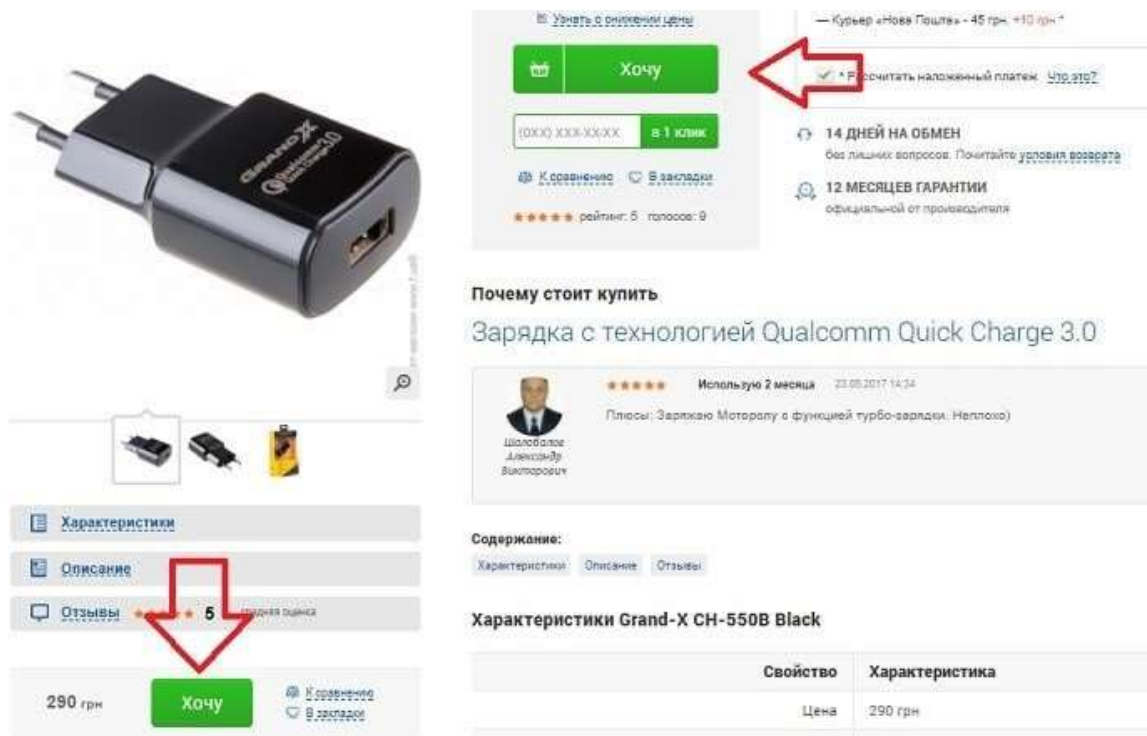


Рис. 1.11. Сторінка товару в магазині “F.ua”

Магазин Allo.ua

Ще один український інтернет-магазин з продажу електронної та побутової техніки – АЛЛО. Яскравий і красивий дизайн виконаний у фірмовому стилі, зручна навігація і функції пошуку по фільтрам (рис. 1.12).



Рис. 1.12. Головна сторінка магазину “Алло”

Як і в попередніх прикладах, функціонал сайту детально продуманий. Він повністю орієнтований на покупця і створений для його комфорту. І ось яскравий тому приклад (рис. 1.13).



Рис. 1.13. Картка товару в магазині “Алло”

У картці товару, крім маркетингових продають тригерів, які мотивують людину до покупки, і зручних функцій вибору найбільш відповідної моделі та її дизайну є функція оцінки товару. При цьому оцінка виконується не в загальному за товар, а по його окремим ключовим характеристикам. Оцінка на початку сторінки є усереднена, формується виходячи оцінки клієнтів, які залишають її в своїх коментарях (рис. 14).



Рис. 14. Сторінка відгуків, магазин “Алло”

Це дуже зручна і корисна функція, адже на її основі можна робити суб'єктивні висновки щодо продукту. Ще один плюс в картці товару – взаємодія магазину зі своїми клієнтами. Крім відгуків тут є можливість залишити питання і отримати на нього відповідь від консультантів компанії. Відмінна ідея, як, втім, і сам магазин.

Всі наведені інтернет-магазини об'єднують одна найважливіша перевага. Крім того, що вони максимально орієнтовані на свою цільову аудиторію (про що свідчить і навігація, і дизайн, і функціонал) і мають структуру зручну для SEO-просування, що дуже важливо для ефективно розкрутки. Тому при створенні свого інтернет-магазину, обов'язково передбачте цей фактор. А якщо ви стикаєтеся з цим питанням вперше, рекомендуємо прочитати статтю як створити правильну структуру під SEO.

1.7 Підходи до реалізації рекомендаційних систем

1.7.1 Типи

Існує 4 типи рекомендаційних систем:

Коллаборативна фільтрація (Collaborative Filtering) - рекомендації засновані на історії оцінок як самого користувача, так і інших користувачів. Цей підхід має теоретично високу точність, але при цьому має одну важливу проблему - високий поріг входу.

Засновані на контенті (content-based) - рекомендації засновані на даних, зібраних про кожен конкретний товар. Користувачеві рекомендуються об'єкти, схожі на ті, якими він раніше цікавився або вже купував. Схожість оцінюється виходячи з вмісту об'єктів. Великий плюс - можливість зацікавити нового користувача пропозиціями з перших кроків. Для цього не потрібно довго збирати дані про переваги, а можна відразу включити споживача в роботу з ресурсом. Можливо рекомендувати навіть ті об'єкти, які не отримали оцінку інших користувачів. Основні недоліки - сильна залежність від предметної області, зниження точності і обмеженість корисності рекомендацій.

Засновані на знаннях (knowledge-based) - рекомендації засновані на знаннях про предметну область (а не про кожен товар). Такий тип рекомендацій має високу точність, пропонуючи користувачеві те, що йому потрібно. Крім цього, система вивчає і аналізує взаємозв'язки між об'єктами, враховує ряд додаткових опцій, що відносяться до індивідуальних властивостей конкретного користувача. До таких властивостей відносяться призначені для користувача побажання (наприклад, їх використовує Rozetka.ua) і демографічні особливості (вихідні дані, які використовують найбільші соціальні мережі, такі як Facebook, LinkedIn, Вконтакте та інші). Основний мінус - складність розробки та збору даних.

Гібридні (hybrid) - рекомендації засновані на комбінуванні коллаборативного і тематичних підходів, що дозволяє уникнути більшості недоліків, властивих кожній системі.

Усі вище розглянуті підходи мають як переваги так і недоліки. Ці підходи детально порівнюються в наступному пункті [8].

1.7.2 Порівняння підходів до реалізації рекомендаційних систем

У цьому пункті порівнюються попередньо-розглянуті підходи до обчислення рекомендацій. Як було сказано, кожен підхід має свої переваги та недоліки. Нижче описуються деякі загальні проблеми, які будь-яка РС має вирішувати.

Розрідження матриці рейтингів. Матриця рейтингів – це матриця, де кожна колонка представляє продукт, а кожен рядок – користувача. Коли користувач i залишає рейтинг r для продукту j , то в матриці рейтингів елементу

					КНУ.РМ.123.22.09.01.ОІРППРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

ij присвоюється значення **r**. Як правило, користувач залишає мало рейтингів для продуктів по відношенню до кількості продуктів. Саме тому матриця рейтингів є дуже розрідженою, що часто спричиняє проблему бракування інформації для знаходження користувачів з подібними уподобаннями. Ця проблема називається *gray sheep problem*.

Проблема холодного старту. Ця проблема виникає у CF алгоритмах, коли додається нова сутність в систему. Новий продукт не може бути рекомендованим з самого початку, коли він додається в систему і ще не має жодного рейтингу. MovieLens (movielens.org) не може рекомендувати нові відео, поки вони не отримають деякі початкові рейтинги. Проблема нового користувача є дещо складнішою для вирішення, адже неможливо знайти схожих користувачів, або створити профіль для CBF алгоритма, допоки цей користувач не залишив хоча б одного рейтингу.

Проблема масштабованості. Ця проблема виникає завжди, коли об'єм даних зростає. У реальних PC Netflix та Amazon алгоритми є масштабованими. У найкращому випадку обчислювальна складність алгоритму має лінійно зростати зі збільшенням кількості даних. Також деякі алгоритми дають високу точність на невеликих наборах даних, але на реальних даних якість рекомендацій значно падає.

Проблема конфіденційності. Щоб обчислювати якісні рекомендації PC має потребу збирати якомога більше даних про користувачів. З іншої сторони користувач може мати підстави підозрювати, що система знає занадто багато про нього. Цей аспект негативно впливає на подальше користування сервісом. Саме тому PC повинна ненав'язливо і прозоро збирати інформацію про користувача, даючи можливість надати більше даних, якщо цього бажає користувач.

Міцність системи. Під цим терміном розуміють можливість системи розпізнати фальшиві транзакції користувачів, спрямовані на те, щоб зробити деякі продукти більш популярними ніж інші.

Це спричиняє велике обчислювальне навантаження.

На практиці комбінують підходи до обчислення рекомендацій, адже це дає змогу покрити недоліки одного алгоритму перевагами іншого.

Рекомендаційна система є необхідним інструментом для фільтрування непотрібних або нецікавих сутностей тому чи іншому користувачу. Наразі PC використовуються багатьма комерційними компаніями. Вони спрощують життя користувачам інтернет-магазинів, та є основним джерелом збільшення кількості продаж продуктів.

Інтернет-магазин надає можливість використовувати різне програмне забезпечення та засоби комунікації для спілкування з клієнтами. До прикладу, телефонний зв'язок, інтернет реклама, розсилка повідомлень у месенджери та ін.

Прикладом такого інтернет ресурсу є магазин «Ібіс» [5], на рисунку 1.15 зображено його головну сторінку. Магазин має категорії товарів, завдяки яким клієнти можуть з легкістю орієнтуватись в товарах та обирати категорію за власним запитом (рисунок 1.16) та каталог товарів, для

можливості перейти на потрібний товар з будь-якої точки сайту (рисунки 1.17).

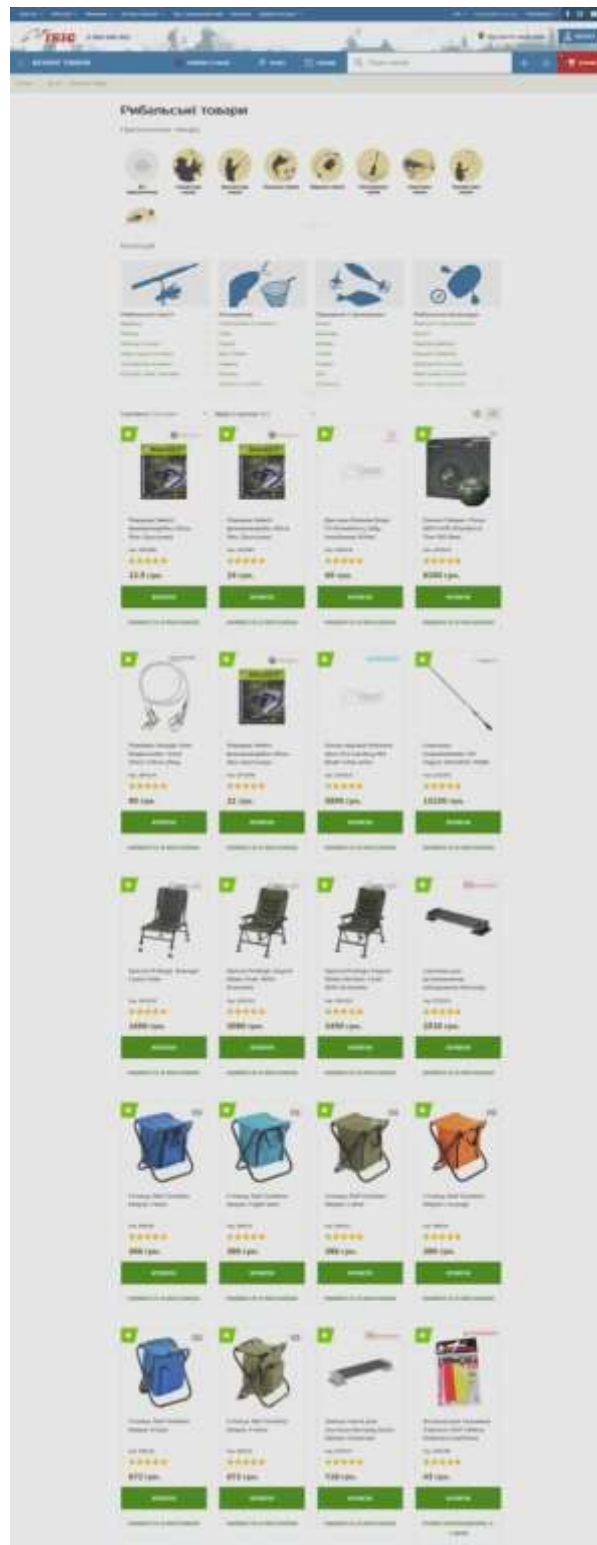


Рисунок 1.15 – Головна сторінка сайту магазину «Ібіс»

Також, однією з функцій сайту є те, що користувач може переглянути та взяти участь у заходах які проводить компанія (рисунки 1.18).

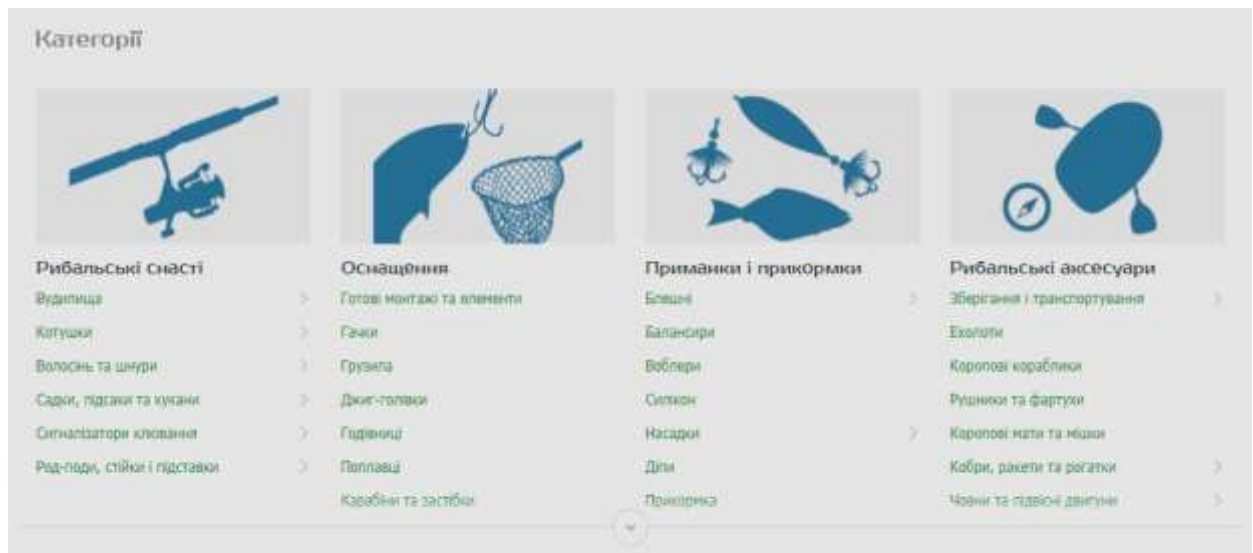


Рисунок 1.16 – Категорії товарів інтернет-магазину

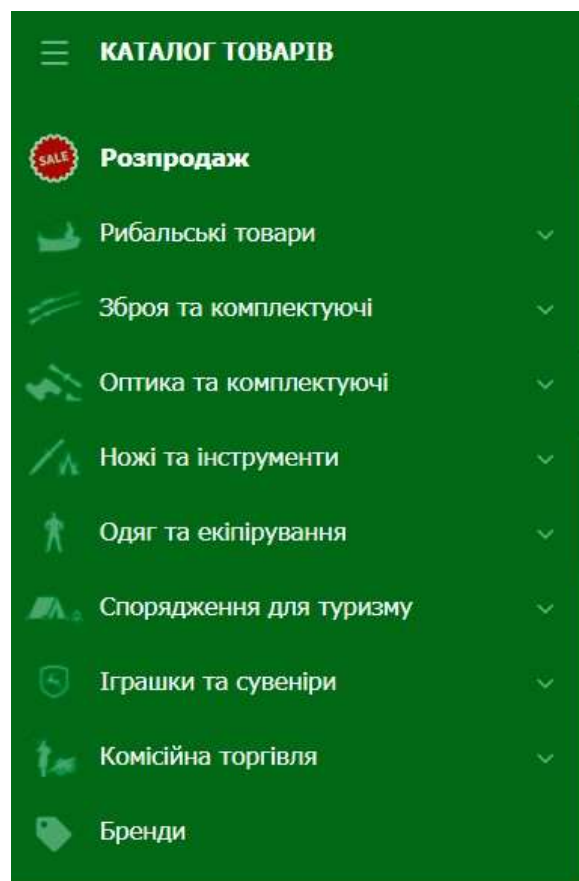


Рисунок 1.17 – Каталог товарів інтернет-магазину

Даний сайт досить зручний та зрозумілий у користуванні, він дає можливість користувачу переглянути товар, оформити замовлення, переглянути актуальну інформацію про компанію та ін.

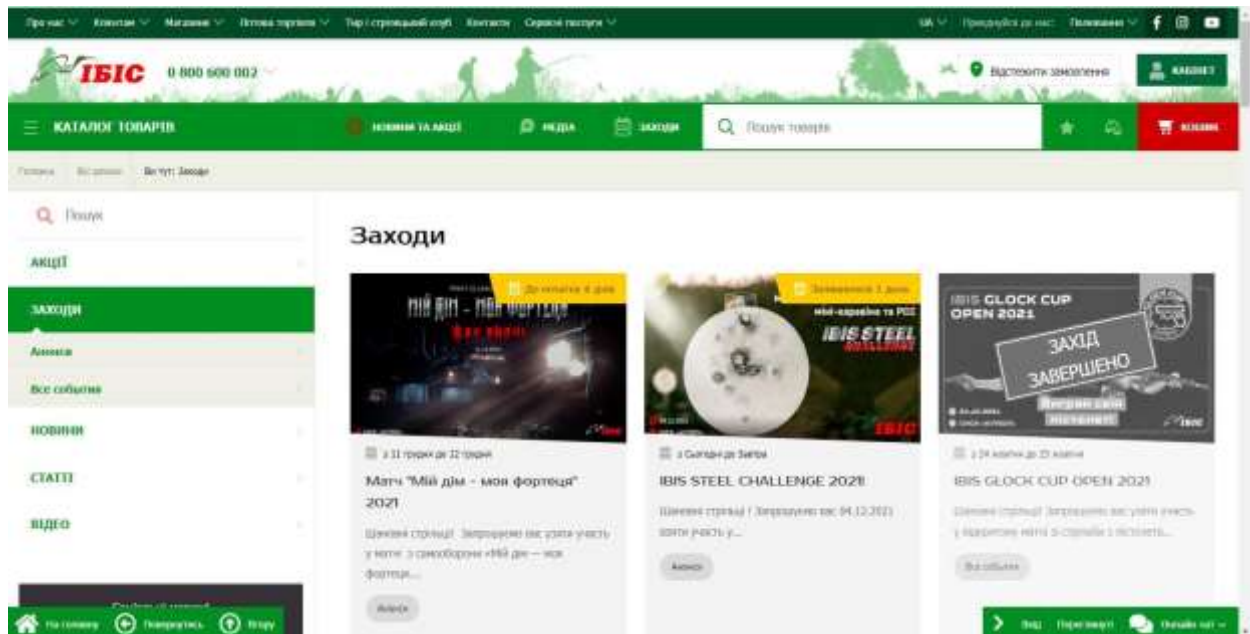


Рисунок 1.18 – Сторінка «Заходи» інтернет-магазину

Ще одним прикладом інтернет-магазину є офіційний сайт магазину «Світ рибалки», головна сторінка якого зображена на рисунку 1.19.

Сайт має вкладку «Товари та послуги», що як і в попередньому інтернет-магазині дозволяє користувачу з легкістю знайти необхідний товар (рисунок 1.20).

Після переходу на одну з категорій відкривається нова сторінка з переліком товарів обраної категорії та бічне меню з переліком категорій що були у вище згаданій вкладці (рисунок 1.21).

Вищезгаданий веб-сайт не має англійської мови і в порівнянні з попереднім ІМ на даному сайті досить простий інтерфейс, що робить його менш привабливим.

Щоб зрозуміти специфіку даної тематики, розглянемо ще один приклад інтернет-магазину. Ним буде офіційний сайт інтернет-магазину «Карпан» [6], головна сторінка якого зображена на рисунку 1.22.

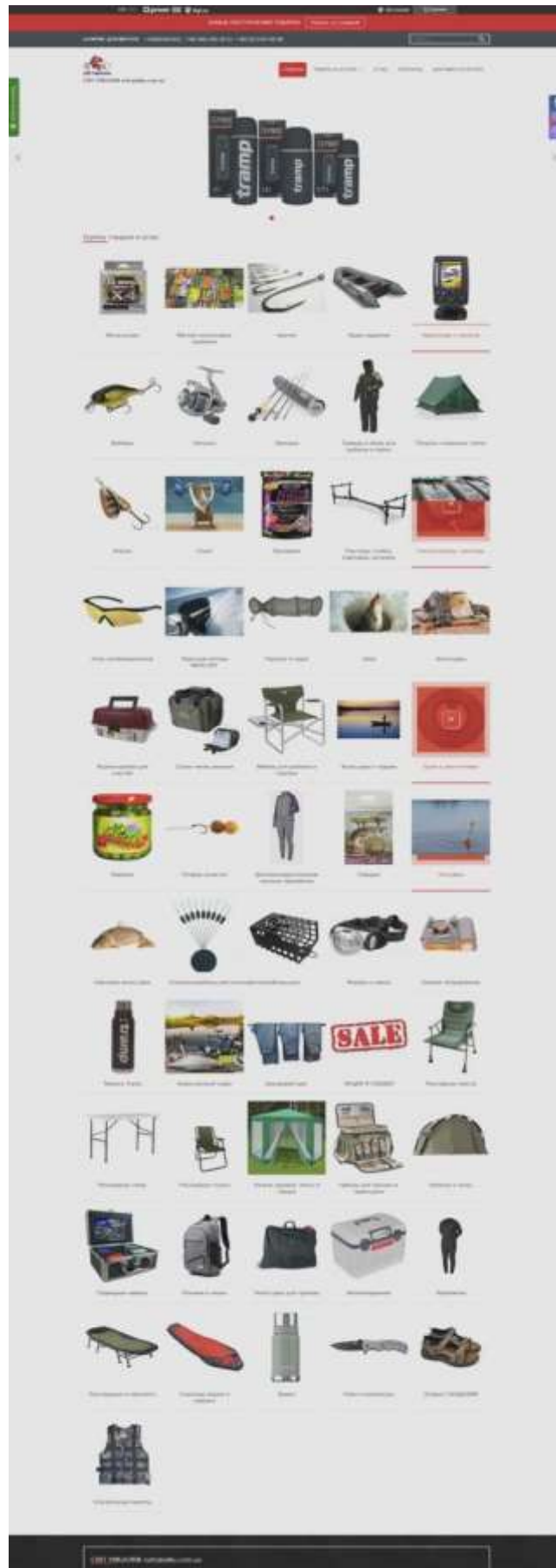


Рисунок 1.19 – Головна сторінка сайту магазину «Світ рибалки»

ГЛАВНАЯ	ТОВАРЫ И УСЛУГИ ▾	О НАС	КОНТАКТЫ	ДОСТАВКА ТА ОПЛАТА
Лески,шнуры	Мягкие силиконовые приманки	Крючки	Лодки надувные	
Навигаторы и эхолоты	Воблеры	Катушки	Удилища	
Одежда и обувь для рыбалки и охоты	Палатки, спальники, тенты	Блесна	Спорт	
Прикормки	Род-поды, стойки, подставки, рогачики	Сигнализаторы, свингеры	Очки поляризационные	
Лодочные моторы MERCURY	Подсаки и садки	Зима	Аксессуары	
Ящики,коробки для снастей	Сумки чехлы рюкзаки	Мебель для рыбалки и туризма	Аксессуары к лодкам	
Груза и джиг-головки	Наживка	Готовые оснастки	Детское(подростковое)и женское термобелье	
Поводки	Поплавки	Карповые аксессуары	Стопора,карабины,светлячки	
Кормушки	Фонари и лампы	Газовое оборудование	Термоса Tramp	
Комиссионный отдел	Брендовый одяг	АКЦИИ И СКИДКИ	Раскладные кресла	
Раскладные столы	Раскладные стулья	Качели садовые, тенты и гамаки	Наборы для пикника и термосумки	

Рисунок 1.20 – Вкладка «Товари та послуги»

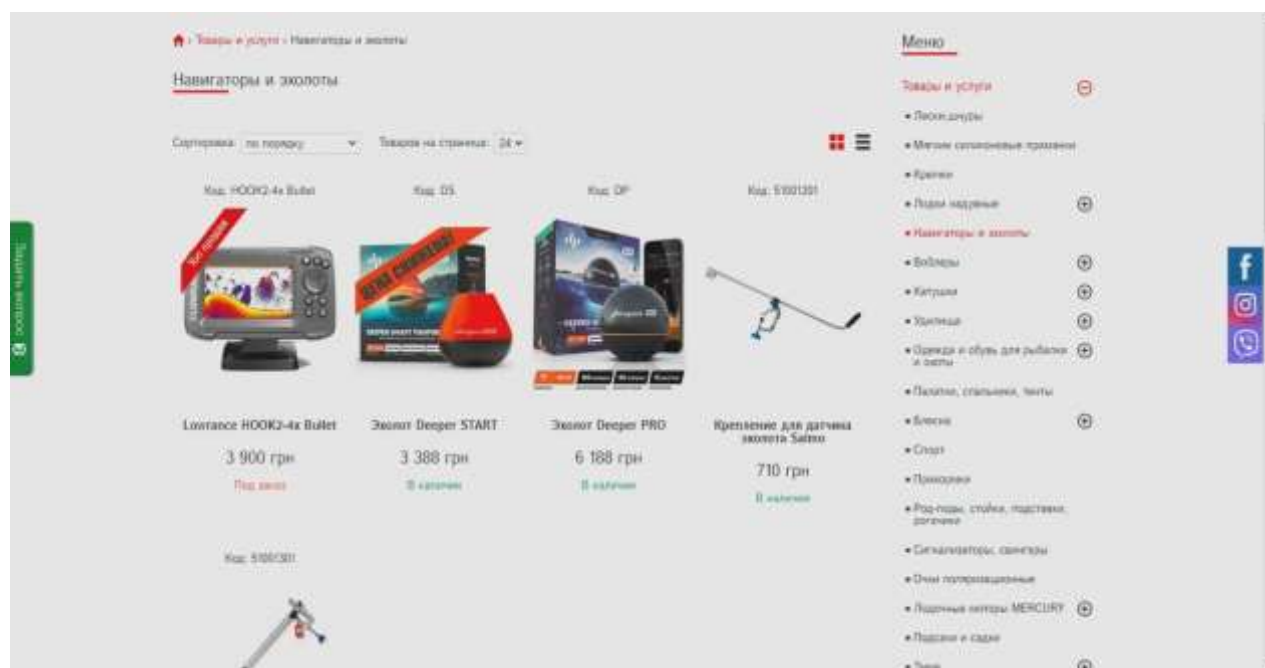


Рисунок 1.21 – Товари обраної категорії

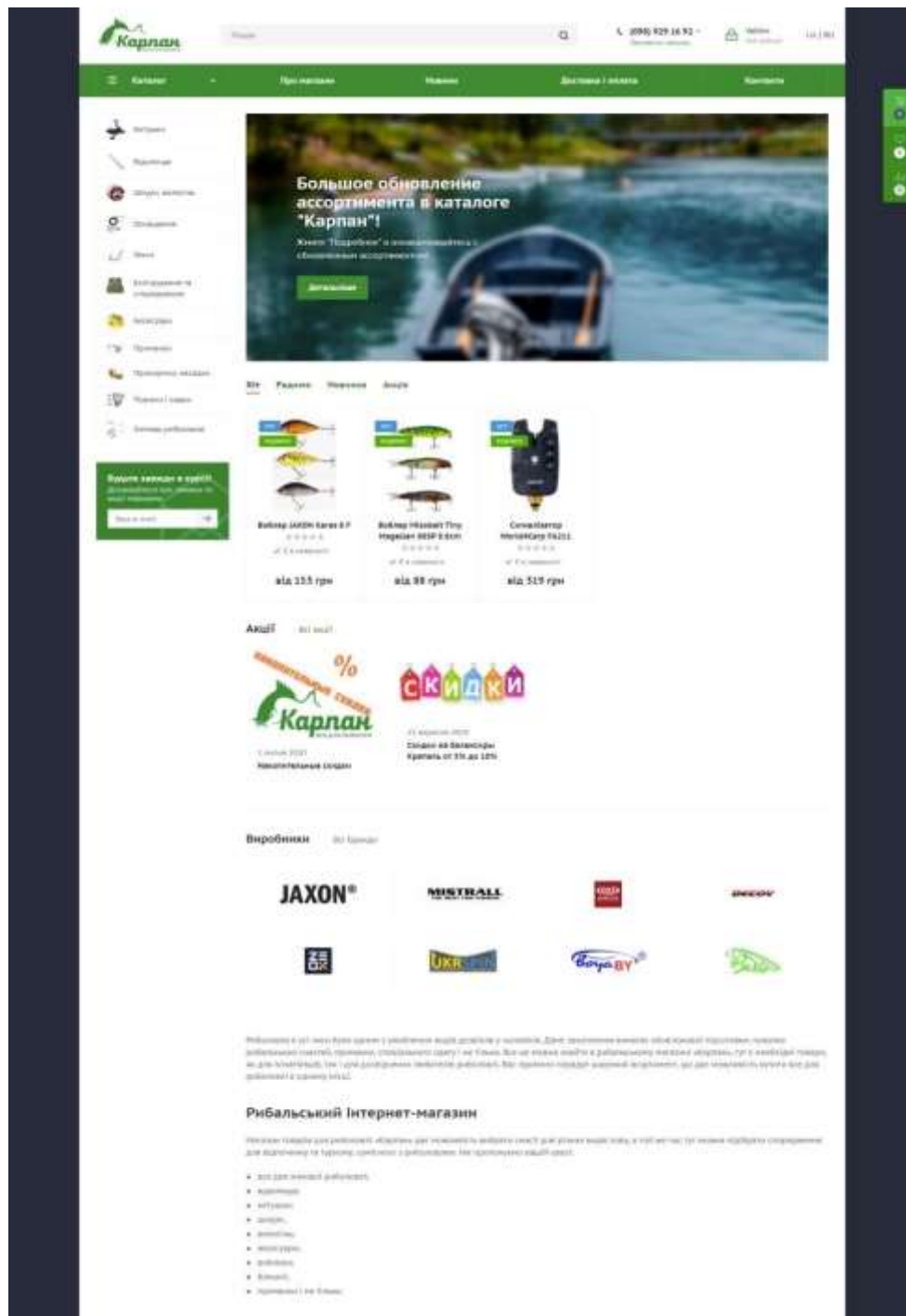


Рисунок 1.22 – Головна сторінка сайту магазину «Карпан»

Сайт має каталог товарів (рисунок 1.23), який допомагає відсортувати товари, також, завдяки фото біля кожної категорії користувачу буде більш зрозуміло які товари туди відносяться.

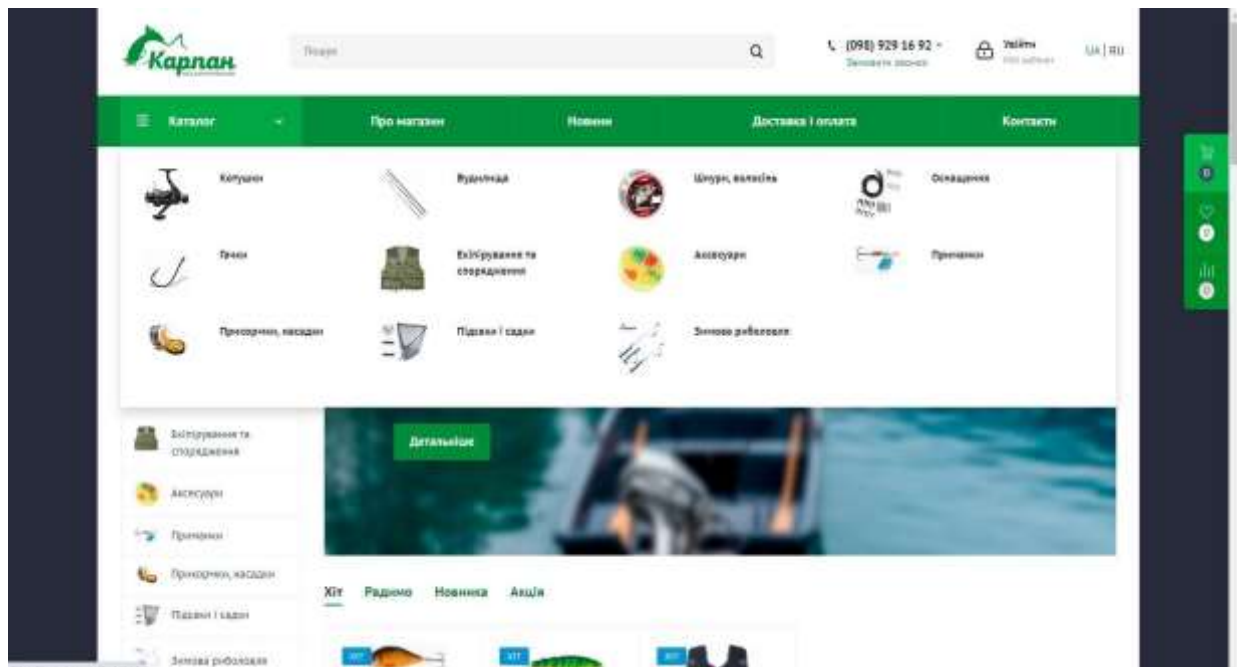


Рисунок 1.23 – Категорії товарів ІМ

Під обраним товаром знаходиться опис товару та рекомендовані, супутні товари (рисунок 1.24).

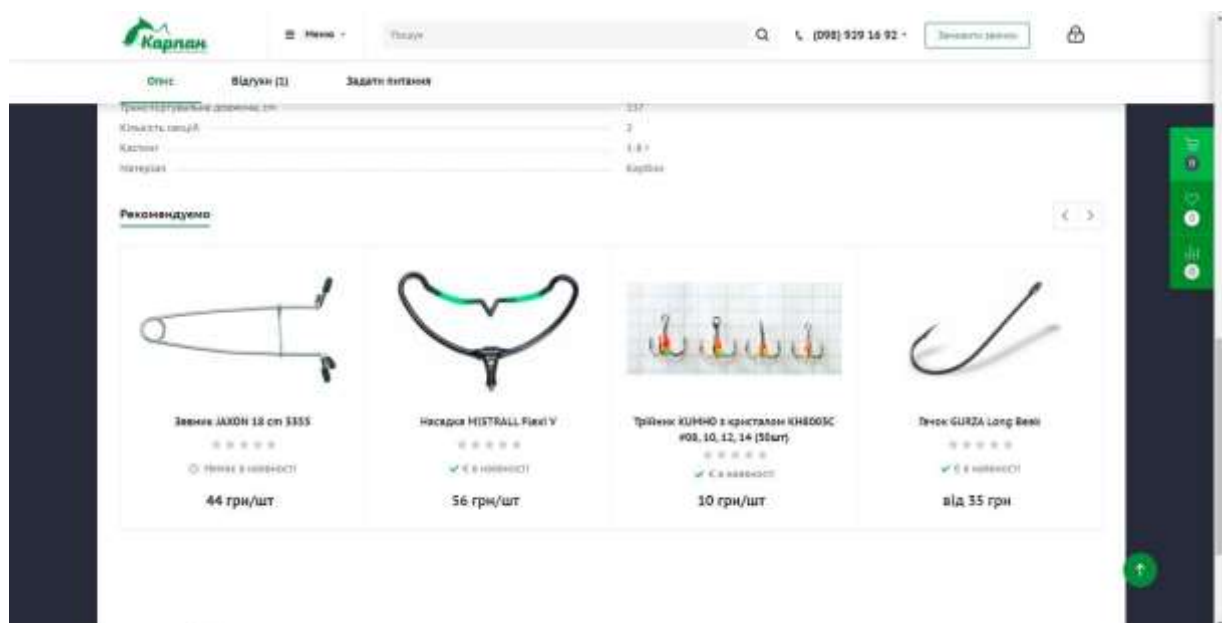


Рисунок 1.24 – Рекомендовані товари

Сайт не має англійської мови, так як і попередній, проте завдяки привабливому інтерфейсу дозволяє отримати більше клієнтів.

Отже, переглянуті та наведені у прикладах системи електронної комерції доводять, що створення та застосування інтернет-магазину для автоматизації продажу рибачького знаряддя через мережу інтернет є ефективним засобом у торгівлі та допомагає розширити базу клієнтів.

У випадках, коли в інтернет-магазині наявна статистика активності користувачів щодо товарів, для пропозиції товарів зазвичай

використовуються традиційні підходи, які полягають в пропозиції товарів згідно статистики переходів попередніх покупців, пропозиції товарів згідно статистики переглядів попередніх покупців та пропозиції товарів згідно статистики покупок попередніх покупців. Проте у випадках, коли в інтернет-магазині додано нові товари чи змінено властивості існуючих товарів, чи й взагалі відсутня статистика активності користувачів щодо товарів, то застосовуються підходи, які полягають в пропозиції товарів з обраної категорії, пропозиції товарів із складеного вручну переліку або пропозиції випадкових товарів з асортименту. Це суттєво погіршує якість пропозицій.

Крім знань про структуру мови, семантика тісно пов'язана із психологією, філософією та іншими науками, тому що неминуче торкається питань про походження значень слів, їх ставлення до буття та осмислення. При семантичному аналізі необхідно враховувати соціальні і культурні особливості носіїв мови. Процес людського мислення, так само як і людська мова, яка являється інструментом вираження думок, дуже гнучкий і важко формалізується. Тому семантичний аналіз вважають найскладнішим етапом автоматичної обробки текстів.

На поточний момент існує багато методів представлення сенсу висловлювань, але жоден із них не являється універсальним. Над співвіднесенням сенсу тексту працювало досить багато дослідників. Так, І.А. Мельчук [7] ввів поняття лексичної функції, розвинув поняття синтаксичних і семантичних валентностей та розглянув їх у контексті тлумачно-комбінаторного словника, який є мовною моделлю. Автор показав, що значення слів співвідносяться не безпосередньо із навколишньою дійсністю, а з уявленнями носія мови про цю дійсність. В.Ш. Рубашкін та Д.Г. Лахуті [8] ввели ієрархію синтаксичних зв'язків для ефективної роботи семантичного аналізатора. Найважливішими є обов'язкові рольові зв'язки, потім ідуть зв'язки кореференції, потім факультативні рольові зв'язки і лише потім предметно-асоціативні. Відомий лінгвіст О.В. Падучева [8] запропонував розглянути тематичні класи слів, зокрема дієслів, оскільки вони несуть основне смислове навантаження.

У цьому підході істотною являється ідея поділу понять мови на деякі семантичні групи з урахуванням того, що ці поняття мають деякий нетривіальний загальний значеннєвий компонент. Елементи таких груп схильні мати той самий набір залежних понять. Універсальна мова представлення знань має повинна бути зручною для здійснення виводу нових знань із уже наявних, а значить, необхідно створити апарат для перевірки правильності висловлювань. Тут якраз корисні логічні моделі уявлення знань. Наприклад, семантична мова, запропонована В.А. Тузовим [8], має в собі формалізми логіки предикатів, присутні атомарні поняття, функції над цими поняттями та правила виведення, за допомогою яких можна описувати нові поняття. Не виключено, що у напрямку створення подібних семантичних мов розвиватиметься наукова думка у майбутньому.

У наш час, в умовах постійного збільшення потоку наукової інформації, поява нових текстових масивів виводить на перший план проблему

КНУ.РМ.123.22.09.01.ОІРППРРС					Арк.
Арк.	№ документа	Підпис	Дата		

систематизації наукових текстів у інформаційному просторі та оптимізації їх пошуку. Актуальними стають питання вивчення лексичних одиниць, які є основними з точки зору передачі змісту наукового тексту, так званих ключових слів дослідження їхньої сутності, а також розроблення методики виділення їх у науковому тексті та визначення їх функцій [9].

Семантичну будову тексту утворюють елементи значень, що містяться в ньому, вони ж започатковують і тему – «сміслові ядро» тексту [10]. Логічною будовою вважається послідовна структура розміщення смислових елементів у процесі розгортання тексту. Сенс тексту формується на основі згорнутої, не вираженої явно смислової структури, що виводиться відповідно до обсягу лінгвістичних, енциклопедичних, фонових знань реципієнта, його досвіду та приналежності до тієї чи іншої культурної спільності, і тому передбачає множинність трактів, що виникають у процесі сприйняття та інтерпретації. Ця структура представляє константне стиснуте смислове ядро тексту, витягнуте з ключових слів, їх асоціативних зв'язків та імплікацій, яке має бути збережене за всіх змін змісту інтерпретаторами.

Основною складністю декодування інформації тексту являється не знання варіантів мовного коду. Крім знання загальномовного коду, правил поєднання слів, морфем, речень, існують ще варіанти такого коду, що визначають правила користування мовними засобами у тих або інших текстах. Знаки вторинних кодів (мов науки та культури) мають більш складну структуру і виконують специфічні функції. У процесі комунікації промовець здійснює відбір засобів мовного вираження сенсу в залежності від соціальних та культурних умов, у яких відбувається спілкування, від мети комунікативного акту. Наприклад, текст законопроекту (будучи науковим текстом) містить елементи офіційно-ділового тексту, мова адвоката містить ораторський стиль, журнальна стаття є публіцистичною. Отже, змінюється план використання емоційних можливостей слова, ускладнюється синтаксис, змінюється логічне побудова тексту. Це особливо помітно під час аналізу текстів законодавчих актів, основною рисою яких є надмірність, перенасиченість пропозицій підрядними [9].

Структура мовного знака ускладнюється, відбувається так зване «прирошення сенсу». Лексична одиниця вживається у своєму вузькому термінологічному значенні, яке, у свою чергу, є більш комплексним, складним та глибшим, ніж значення слова, яке звичніше не фахівцеві поточної галузі знань.

Важливою проблемою є й те, які саме одиниці тексту є вирішальними для розуміння всього тексту. Значення окремих елементів тексту для вираження загального сенсу неоднакове та поряд з центральними елементами є також і другорядні елементи тексту. При зміні центральних елементів зміст тексту змінюється. Якщо змінити другорядні одиниці тексту, то загальний сенс може зберегтися. Актуальним на даний момент являється питання про складання об'єктивної методики виділення ключових слів у науковому тексті. У сучасних зарубіжних та вітчизняних дослідженнях проблема ключових слів розглядається у зв'язку з розробкою інформаційних пошукових систем, у рамках теорії комунікації, психолінгвістики, когнітивної лінгвістики [9].

					КНУ.РМ.123.22.09.01.ОІРППРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

Отож і підходи до визначення даного поняття суттєво відрізняються.

Дослідники звертають увагу на ієрархію мовних засобів у тексті, на існування в ньому домінантних та субдомінантних мовних коштів, виділяючи у своїй різні аспекти даного мовного явища.

У загальнонауковому плані ключові слова розглядаються як найважливіший, існуючий для розуміння смисловий елемент. Незважаючи на різноманітність багатосторонніх підходів до тлумачення ключових слів, абсолютно об'єктивним є те, що ключове слово визначає зміст тексту та є носієм його основного сенсу. Під ключовими словами розуміють слова, що мають особливий статус, виражають важливе соціальне значення що містить оцінку, відіграють у тексті важливу роль. З лінгвістичної точки зору вони сприяють довготривалому пошуку одиниць значення [9].

Ключові слова мають низку диференціальних ознак, що дозволяють впізнавати їх у тексті. Належність слова до ключових визначається перш за все за [9]:

- Частотністю, повторюваністю у цьому тексті. Цей критерій реалізується шляхом підрахунку повторень слівформ протягом усього тексту та виділення в якості ключових тих слів, частотність вживання яких у цьому тексті перевищує частотність їх вживання у мові.
- Семантичною близькістю слова до теми тексту (тут розглядаються синонімічні, родовидові відносини та ін).
- Інформаційною насиченістю (ця вимога виходить з особливостей функціонального стилю – максимум інформації на мінімальному просторі). Економія має своїм результатом той факт, що найбільш «важкими» у смисловому аспекті стають наукові терміни. Термін характеризується інформаційною насиченістю, яка визначається числом безпосередніх та опосередкованих семантичних компонентів терміна [9].

Слід зазначити, що у тексті використовуються як терміни, але й інші ємні слова (наприклад, власні імена, географічні назви, обставини часу та місця). У рефераті (як вторинному науковому тексті) також може бути експліцитно виражена вказівка на коло читачів, і вона також може не містити термінів даної галузі знання. Для спрощення структури наукового тексту (для якого характерні складні розгорнуті синтаксичні конструкції), для надання йому стерео типізації, в ньому часто використовуються паралельні конструкції. Завдяки синтаксичному паралелізму вводяться перерахування, уточнення, з'являються синонімічні повтори та заміни, які концентрують увагу на лексичній одиниці – носійці основного змісту тексту.

Для аналізу тексту, визначення його характеристик використовують семантичний аналіз. Завдяки такому аналізу можна отримати дані, які у подальшому будуть використовуватись для пошуку тексту та просуванні аналізованого тексту у пошукових системах [11].

Результатом семантичного аналізу тексту є те, що стають відомими такі характеристики:

- обсяг тексту (кількість символів);
- кількість слів та стоп-слів;

- частота вживання слів, тобто ключові слова;
- відсоток водності;
- нудота тексту;
- граматичні та орфографічні помилки.

Прикладною лінгвістикою називають розділ мовознавства, який розв'язує питання з практичної точки зору [12]. Існує багато сервісів для пошуку ключових слів текстів, які допомагають швидше знайти та визначити слова для статті або тексту.

Таким чином, застосування семантичного аналізу тексту є перспективним напрямком для пропозиції подібних товарів. У випадках, коли в інтернет-магазині додано нові товари чи змінено властивості існуючих товарів, чи взагалі відсутня статистика активності користувачів щодо товарів, то існуючі методи пропозиції товарів виявляють дуже низьку ефективність. В таких випадках підходи в напрямку автоматизованої адаптивної пропозиції товарів у інтернет-магазині за семантичними та лінгвістичними ознаками описів профілів товарів, які можуть бути застосовані навіть у випадках відсутності повної статистики по діях із товарами користувачів інтернет-магазину є ефективними, що визначає актуальність роботи в даному напрямку.

Такий підхід при прикладному застосуванні забезпечить високу стійкість за коригування прайсу товарів, оскільки додавання чи видалення товарів не впливають вже виконані обрахунки семантичної подібності товарів.

Висновки за розділом

В розділі проведено аналіз предметної області електронної комерції та проблеми пропозиції товарів. За результатом проведеного дослідження виявлено, що вагому роль при просуванні товарів в інтернет-магазинах відіграє інструмент пропозиції товарів, який збільшує імовірність відповідної зміни фокусу уваги клієнта.

Встановлено, що у випадках, коли в інтернет-магазині наявна статистика активності користувачів щодо товарів, для пропозиції товарів зазвичай використовуються традиційні підходи, які полягають в пропозиції товарів згідно статистики переходів попередніх покупців, пропозиції товарів згідно статистики переглядів попередніх покупців та пропозиції товарів згідно статистики покупок попередніх покупців. Проте у випадках, коли в інтернет-магазині додано нові товари чи змінено властивості існуючих товарів, чи й взагалі відсутня статистика активності користувачів щодо товарів, то застосовуються підходи, які полягають в пропозиції товарів з обраної категорії, пропозиції товарів із складеного вручну переліку або пропозиції випадкових товарів з асортименту. Це суттєво погіршує якість пропозицій.

2. АЛЬТЕРНАТИВНІ ПІДХОДИ ДО РЕАЛІЗАЦІЇ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

2.1 Гібридні рекомендаційні системи

Гібридні рекомендаційні системи об'єднують два, або більше рекомендаційних підходів, щоб отримати більшу точність та продуктивність, покриваючи недоліки кожного окремого підходу в системі. Найчастіше підхід колаборативної фільтрації (CF) поєднується з деяким іншим підходом, щоб уникнути проблеми надмірної специфічності рекомендацій[8]. Таблиця 2.1 показує методи комбінування різних підходів до обчислення рекомендацій.

Таблиця 2.1 Порівняння рекомендаційних підходів

Гібридний підхід	Опис
Зважений (Weighted)	Рейтинги, що передбачені різними алгоритмами лінійно, або нелінійно комбінуються в єдине значення, яке й використовується для вибору продуктів для рекомендації
Перемикання (Switching)	Система вибирає підхід, що використовується для підрахування рекомендації залежно від ситуації
Мікс (Mixed)	Рекомендації з декількох різних підходів змішуються в один список
Комбінування властивостей (Feature combination)	Властивості від різних джерел даних комбінуються і передаються як вхідні дані до деякого алгоритму
Каскадування (Cascade)	Один підхід уточнює результати рекомендацій іншого

					КНУ.РМ.123.22.09.02.АРРС			
Змн.	Арк.	№ документа	Підпис	Дата	АЛЬТЕРНАТИВНІ ПІДХОДИ ДО РЕАЛІЗАЦІЇ РЕКОМЕНДАЦІЙНИХ СИСТЕМ	Літера	Аркуш	Аркушів
Розробив	Коржов.							
Перевірив	Кумченко							
Н.контроль	Кузнєцов					КІ-21м		
Затвердив	Купін							

Зважений підхід (Weighted)

У зваженому гібридному підході оцінка рекомендованого продукту обчислюється виходячи з результатів усіх доступних методів рекомендацій, що присутні в системі. Перевага зваженої гібридної системи в тому, що всі її можливості задіяні у просту та зрозумілу лінійну комбінацію результатів. Зміну коефіцієнтів можна виконувати за лічені секунди.

Перемикування (Switching)

Перемикувальна система використовує деякий критерій для визначенні проміжків часу або ж певних моментів, коли той чи інший підхід буде давати кращі результати.

В рамках перемикувальної системи ми знову незалежно будуємо наші рекомендаційні системи, і далі, в той момент, коли нам потрібно обчислити рекомендації для користувача, ми вибираємо, яка рекомендаційна система буде це робити. Тобто в рамках цього підходу ми вже не змішуємо разом рекомендації від різних систем, а вибираємо, яка система буде джерелом рекомендацій для користувача. Для того щоб зробити такий вибір, нам потрібно побудувати деякі критерії, які підкажуть нам, яку систему слід вибрати. Ідея тут досить проста. Наприклад, CBF підхід дозволяє нам вирішувати проблему

холодного старту для нових продуктів. Тому якщо ми хочемо порекомендувати користувачеві об'єкт, якого в нашій системі ще не було або він з'явився недавно, і досить багато статистики про нього ми ще не накопичили, то в цьому випадку нам логічніше використовувати CBF підхід. Якщо ж ми намагаємося оцінити продукт, який вже давно в нашій системі та є багато рейтингів цього продукту, то CF, можливо, в цьому завданні спрацює краще. Відповідно, ми можемо побудувати деякі критерії, на підставі яких ми будемо робити вибір, і використовувати правильну рекомендаційну систему.

Мікс (Mixed)

В рамках цього підходу ми знову генеруємо рекомендації незалежно, кожна система будує свій список рекомендацій, однак підсумковий список будується на основі суміші рекомендацій від різних систем. Це добре працює в тих випадках, коли нам потрібно одночасно отримати довгий, різноманітний список рекомендацій, наприклад, безперервний feed новин або рекомендації медіаконтенту. В даному випадку часто використовують цю схему, тому що ми можемо доповнювати рекомендації від різних рекомендаційних систем один одним.

Комбінування властивостей (Feature combination)

Цей підхід в деякій мірі заснований на підході CBF і є його доповненням. Ідея в тому, що ми хочемо об'єднати в одній навчальній вибірці ознаки від різних підходів.

Найчастіше це виглядає наступним чином: ми використовуємо інформацію про переваги схожих користувачів, отриману в рамках колаборативного підходу (CF), в якості ознак для CBF підходу. Тобто ми з вами можемо розглянути рейтинги, які отримав об'єкт за допомогою користувачів, схожих на нашого користувача, і цю інформацію використовувати як ознаки. Таким чином, ми збагачуємо ознаки опису об'єкта, отриманого в рамках content-based підходу, і додаємо туди інформацію про рейтинги. Це часто досить непогано працює[8].

Каскадування (Cascade)

Це дуже красива ідея, в рамках якої ми послідовно застосовуємо декілька рекомендаційних систем для уточнення рекомендацій. Припустимо, що у нас є рекомендаційна система, яка працює швидко і може досить грубо і просто поділити об'єкти на дві множини: точно нерелевантні, тобто нецікаві користувачеві і потенційно релевантні. Тоді давайте застосуємо цю рекомендаційну систему, відкинемо ті об'єкти, які нашого користувача точно не зацікавляють, і далі будемо працювати тільки з тією групою об'єктів, які потенційно релевантні. Ось до них давайте застосуємо більш складну рекомендаційну систему, яка, можливо, працює довше, але зате вона зможе більш точно відранжувати ці об'єкти для користувача. Така схема часто називається Candidate selection, коли за допомогою одного алгоритму, що виконує обчислення більш ефективного, ми з відбираємо безліч кандидатів і далі за допомогою іншої, більш складної, системи їх ранжуємо.

2.2 Рекомендації на типовому веб-сайті електронної комерції

Є багато місць, де рекомендації можуть відображатися на сайті електронній комерції. Основні методи рекомендацій найкраще працюють на головних сторінках, у кошику, на сторінках категорії.[4]

Маркетингові дослідження. Як і будь-який бізнес, створення інтернетмагазину потрібно начитати з маркетингових досліджень. Мета цього важливого етапу - отримати інформацію, яка буде основою для успішного початку бізнесу. Потрібно обов'язково дослідити попит на продукцію, що продається, конкурентів, цільову аудиторію і інші важливі аспекти.

Правильний варіант – віддати дослідження в маркетингове агентство або, принаймні, проводити їх спільними зусиллями з фахівцем з досліджень. Економ варіант - провести дослідження власними силами, які теж дадуть багато корисної інформації. Якщо вже є чітке рішення робити магазин, можна знайти підрядника, який зможе забезпечити і розробку, і просування магазину, після чого попросити допомоги у нього в проведенні передпроектних досліджень з метою створення

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

якісного магазину. Спільними силами можна отримати інформацію про конкурентів і їх переваги, дізнатися очікуваний попит, останні технічні новинки в цій галузі і багато іншої інформації. Вся ця інформація повинна враховуватися як при розробки сайту магазину, так і при його просуванні.

Комунікація. Наступний важливий елемент даного бізнесу - організація зручною для споживачів комунікації. Мета етапу - зробити багато різних варіантів комунікації, щоб споживач з будь-якими перевагами зміг швидко зв'язатися з менеджерами магазину [17].

Перший канал комунікації - це телефон, на нього можна очікувати найбільше запитів. Зараз з телефонами особливих проблем немає: з будь-якої точки світу можна приймати дзвінки і дзвонити по прямому київським (044) або іншим номером, і коштує це зовсім недорого в залежності від технології (IP телефонія, CDMA і ін.) І телекомунікаційної компанії - 5-30 \$ / міс.

Наприклад, інтернет-магазин “Зв'язковий” - <http://www.svyaznoy.ru/> приймає дзвінки цілодобово. Можна організувати багатоканальну гарячу лінію (наприклад, інтернет-магазин «АЛЛО» – <http://www.allo.ua/> має гарячу лінію).

Важливою частиною в магазині є *on-line* чат, який дозволяє спілкуватися за допомогою миттєвих повідомлень (наприклад, сервіс миттєвих повідомлень для сайту “Siteheart” - <http://siteheart.com/>, використовувати можна безкоштовно). У клієнтів часто виникають питання, і дуже зручно зробивши всього один клік, отримати відповідь на питання, що цікавить (наприклад, інтернет-магазин

“SVEN” – <http://shop.sven.ua/> має on-line чат).

Наступне, на що варто звернути увагу – це меседжер, типу *Telegtam, Viber, Skype* і багато інших. Зареєструвати їх не складає труднощів. Однак для великих магазинів може виникнути труднощі з розподілом навантаження на кожен контакт. У цьому випадку проблему можна вирішити технічно: кожному відвідувачу показується 1 – 2 контакти, але при цьому вони постійно змінюються (відвідувачеві “А” показують контакт “А”, відвідувачеві “Б” показують контакт “Б” і так по колу), таким чином, навантаження буде рівномірно розподілятися.

Потім можна згадати стару технологію *email*, яка до сих пір добре себе почуває. Мінімум потрібно створити один загальний *email* і по одному для кожного менеджера з продажу, на сайті розмістити форму зворотного зв'язку, якої до сих пір користуються деякі люди, а потім регулярно перевіряти пошту. Покупці часто пишуть електронні листи і нерідко не отримують зворотної відповіді від магазинів – це велика помилка, яка призводить до втрати клієнтів.

Менеджери з продажу. Чимало людей чомусь думають, що для інтернетмагазину майже не потрібні люди, зокрема, менеджери з продажу. Це оману. Будьмагазин без професійних продавців добре працювати не зможе.

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

Уявімо ситуацію, коли дзвонить має сумнів потенційний клієнт і задає пару технічних питань: «технар» просто відповість на питання і попрощається, а “продажник” відповість на питання, потім з'ясує потреби і підбере такий продукт, який максимально підійде клієнту – звичайно, різниця величезна, особливо якщо перенести цю ситуацію на великий потік клієнтів і кінцеві прибутку [1].

Менеджери з продажу в інтернет-магазині повинні бути і повинні знати всі особливості продажів через інтернет, тому що вони значно відрізняються від традиційних продажів в реальному світі.

Доставка. За даними *ROMIR* покупці виділяють 2 основні переваги інтернетмагазинів: 68 % респондентів називають економію часу і 54 % економію грошей. Звичайно, доставка в економії часу грає велику роль, тому дуже важливо зробити її швидкої і зручної для користувачів. Покупець повинен зайти в магазин, швидко знайти потрібний товар, легко за нього заплатити і швидко його отримати у зручний йому час.

Зараз є багато служб доставки, яким це можна віддати на аутсорсинг (місцеві кур'єрські служби, міжміські, наприклад, “Автолюкс” – <http://www.autolux.ua/>, “DHL” – <http://www.dhl.com/>, “Нічний експрес” – <http://www.nexpress.com.ua/> і інші). Для великих магазинів можна побудувати власну службу доставки. На сайті це буде виглядати як опція при замовленні і окрема сторінка з описом доставки, а для магазину важливо.

Оплата. Ще один важливий для користувачів магазину елемент – оплата. В процесі оформлення покупки у користувача повинна бути можливість вибору зручного варіанту оплати. Сам процес може відбуватися по кроках: спочатку вибираємо сам спосіб оплати (готівкою, електронними грошима, пластиковою картою і т.д.), потім система автоматично, в залежності від обраного способу, генерує рахунок на оплату, а також пропонує вибрати електронну платіжну систему або ввести дані карти і т.д. Суть в тому, щоб користувач оплатив так, як йому зручно, і витратив мінімум часу (наприклад, інтернет-магазин “Болеро” – <http://www.bolero.ru/> пропонує багато різних способів оплати).

Для організації таких платежів можна використовувати популярні системи електронних платежів (наприклад, “ASSIST” – <http://www.assist.ru/>,

“ChronoPay” – <http://www.chronopay.com/>, “Інтеркаса” – <http://www.interkassa.com/> і інші, і всі вони беруть відсоток за послуги посередництва) або зробити все без посередників, прямі прийоми платежів.

Сезонність. Стандартна особливість торгового бізнесу - в різні пори року буде різний рівень продажів. Як правило, традиційні бізнес сезони весна і осінь, в цей час продажу вище, а “несезон” літо і зима. Також продажу деяких категорій товарів ростуть в свята (наприклад, інтернет-магазин “GOLD.ua” – <http://gold.ua/>

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

пропонує подарунки до традиційних свят на 23 лютого і 8 березня і навіть виділяє це в контекстному меню зліва).

Сезонність потрібно враховувати не тільки в плануванні продажів, але і перед запуском магазину. Розробку нового магазину варто починати не пізніше, ніж за 3 місяці до початку сезону, інакше можна не встигнути до початку сезону закінчити досягнення високого рівня продажів і втратити прибуток.

Сайт магазину. Один з найважливіших елементів успішного магазину. “Сайт повинен ефективно перетворювати відвідувачів в покупців” – стандартна фраза багатьох розробників, за якою ховається важливий сенс: на одних сайтах потенційний покупець швидко і без коливань робить покупку, а на інших не може знайти потрібний товар, хоча він там є. Для вимірювання ефективності сайту зазвичай використовують поняття “коефіцієнт конверсії”, який показує співвідношення відвідувачів до покупців. На деяких сайтах цей коефіцієнт сягає значення 20 % і більше, тобто з 100 відвідувачів – 20 роблять покупку [3].

Існує цікава маркетингова модель *AIDA* (*Attention, Interest, Desire, Action* – увага, інтерес, бажання, дія), яка активно використовується на практиці в США. Її суть полягає в тому, щоб змусити відвідувача пройти 4 етапи: спочатку звернути його увагу на продукт, потім зацікавити його продуктом, після викликати бажання отримати продукт і в кінці змусити зробити дію (в нашому випадку купити продукт). Весь сайт повинен будуватися за цією моделлю, вона дуже допомагає збільшувати число покупців, але, на жаль, далеко не всі розробники знають про існування цієї моделі.

Зовні магазин повинен обов'язково викликати довіру, цьому сприяє багато чинників: популярність, вік бізнесу, хороший дизайн, сертифікати від виробників і т.д. Для Інтернету це дуже важливі елементи, зараз у багатьох користувачів рунеті все ще розвинене велика недовіра до *on-line* покупок.

Конверсія. Цьому показнику варто приділити особливу увагу. Про нього доведеться думати завжди: при розробці сайту, в процесі просування і в майбутніх прогнозах. Конверсія - універсальне мірило ефективності інтернет-магазину.

Середній рівень конверсії в рунеті 1 – 2%, тобто з 100 відвідувачів покупку зроблять всього 1 – 2 людини. На заході цей показник значно вищий. Пов'язано це, перш за все, з неправильним підходом до побудови цього бізнесу у нас, вірніше до банального невміння, але про помилки розповім нижче. У той же час в деяких магазинах конверсія перевищує 20 % і більше (наприклад, за даними видання “*Marketing Charts*” – <http://www.marketingcharts.com/>, деякі магазини досягають конверсію в 20, 30 і навіть 40%; один з таких магазин

“ProFlowers” – <http://www.proflowers.com/> в листопаді 2009 отримав конверсію в 20,1 %, а в грудні – 22 %, і на такому рівні конверсії він знаходиться мінімум 2 роки і, до слова сказати, магазин дуже популярний) [13].

Дуже важливим для конверсії буде ціна: чим вона вища, тим менше конверсія. Це пояснює високу конверсію на сайтах з недорогою продукцією (наприклад, квіти) і низьку з дорогими товарами (наприклад, автомобілі).

Конверсія залежить, перш за все, від 3-х глобальних частин:

- Бізнес. Бренд, ціни на продукти, якість продуктів, умови оплати, доставки, гарантія тощо
- Сайт. Наскільки сайт підходить для продажів, що продає, який у нього функціонал, дизайн і т.д.
- Просування. Хто приходить на сайт, яка лояльність у відвідувачів, що шукають і т.д.

Якщо випустити з уваги хоча б одну з 3-х частин - конверсія буде низькою. Але, як не дивно, у нас примудряються втратити і 1-у, і 2-у, і, в особливо безнадійних випадках, навіть 3-ю частини чарівної формули.

Звичайно, вище описані в повному обсязі особливості, і, звичайно, є ще стандартні моменти торгового бізнесу: реальний офіс, логістика, кредитування покупців, програми лояльності та багато-багато інших важливих моментів.

Головне, щоб бізнес був цілком конкурентоспроможним на ринку, тільки в цьому випадку можна працювати над сайтом і його просуванням, і саме тому сильно виграють великі роздрібні мережі магазинів з реального світу, які багато років працювали над збільшенням конкурентоспроможності та власним брендом; таким гігантам досить тільки грамотно інтегруватися в мережу.

Хостинг і домен. Для звичайного інтернет-магазину з великою кількістю товару і високою передбачуваною відвідуваністю підійде простий хостинг на 5000 мб з необмеженим трафіком (100-150 \$ / рік), для великого гіпермаркету потрібно брати VDS (від 200 \$ / рік) або виділений сервер (від 1500 \$ / рік): він і для навантаження потрібен, і багато місце на диску потрібно.

Доменне ім'я повинне бути просте, добре запам'ятовується і бажано відображати сенс магазину (наприклад, книжковий інтернет-магазин

“Books.Ru” – <http://www.books.ru/>, відразу зрозуміло, що там продають, і складаються правильні асоціації).

Платформа. Зараз є дуже багато різних рішень для інтернет-магазинів. У всіх є переваги, недоліки і особливості (огляд CMS для інтернет-магазинів з цікавим обговоренням можна знайти тут: <http://www.nezname.com/?P=352>).

Перший варіант - створити власну систему управління сайтом (CMS) під свої вимоги. Цей варіант підходить в основному для дуже великих або нестандартних інтернет-магазинів, в яких потрібно враховувати багато

нестандартних функцій, що найкраще робити з нуля. Однак у цього варіанту багато недоліків, варто виділити хоча б 3 основних:

- **Вартість.** Звичайно, створювати новий якісний рішення з нуля завжди набагато дорожче, це, напевно, основний недолік.
- **Терміни.** У цьому випадку вони збільшаться в кілька разів у порівнянні з розробкою на коробковому вирішенні.
- **Безпека.** Якщо багато коробочок CMS вже пройшли купу тестів на безпеку, то у власній системі цей момент потрібно буде враховувати своїми силами.

Через це варіант розробки власної платформи використовується дуже рідко, тим більше, що є багато якісних коробкових рішень з відкритим кодом, які можна самостійно допрацьовувати.

Другий варіант – використовувати безкоштовне коробочки рішення. Зараз є багато безкоштовних CMS з відкритим вихідним кодом, в тому числі і спеціалізованих для інтернет-магазинів. Одним з основних переваг такого рішення є ціна – вартість розробки буде низька, а за саму ліцензію платити нічого не потрібно. Недоліків втім, набагато більше: відсутність технічної підтримки, часто функціональність гірше платних аналогів, можуть бути серйозні проблеми з безпекою, що для магазинів особливо погано. Різних систем багато, відразу можна подивитися в бік: “osCommerce”, “*PHPShop*”, “Joomla”, “*Drupal*” і ін.

Третій варіант – платне коробочки рішення. Вибір таких рішень на ринку дуже великий, з різною функціональністю і різними цінами. Переваг багато: офіційна технічна підтримка, хороша функціональність, висока безпека і продуктивність, безкоштовні оновлення і т.п., основний недолік один - за ліцензію потрібно заплатити, а кінцева вартість розробки буде більше, ніж при розробці на безкоштовному аналогу. У цій категорії варто звернути увагу на: “1С-Бітрікс”, “Amiro”, “HostCMS”, “NetCat”.

Будь-який з 3-х варіантів має своїх споживачів: свою платформу роблять великі проекти, безкоштовну CMS вибирають малобюджетні проекти, а платну - середні за розміром магазини з хорошими вимогами до якості. Я рекомендую дивитися в бік третього варіанту.

Проектування і функціональність. Це, мабуть, найважливіша частина розробки інтернет-магазину, відразу на думку спадає російське народне прислів'я: “Що посієш, те й пожнеш”. На цьому етапі дуже важливо все добре і однозначно спроектувати в майбутній інтернет-магазин. Підсумком проектування повинен стати спеціальний документ – технічне завдання, його зазвичай пишуть розробники сайту магазину, відштовхуючись від побажань клієнта [2].

ТЗ в загальному вигляді містить опис всього функціоналу, вимоги до дизайну (ці вимоги часто виносяться в окреме ТЗ, вони потребують особливо ретельного підходу, тому що саме тут процес розробки може зайти в глухий кут, якщо з'явиться нерозуміння між клієнтом і розробником щодо дизайну), макетування сторінок, вимоги до верстки, безпеки, продуктивності і т.д. Однак навіть дуже докладний і пророблена ТЗ може трохи змінюватися в процесі роботи, це цілком нормально.

В технічні подробиці не будемо вдаватися, зупинимось на спеціальних функціях, які приносять користь з точки зору продажів і тому обов'язково повинні закладатися в ТЗ. Сьогодні особливе місце потрібно відводити інформації про продаваних продуктах (за даними “ROMIR” такої інформації не вистачає 51 % покупців). Можна виділити десять основних функцій впливають на продаж:

- Перша така функція – фільтри за різними параметрами (наприклад, фільтр в інтернет-магазині “Rozetka” дозволяє впорядкувати товари по імені, ціною і іншим параметрам). Дуже зручно для вибору потрібного товару.

- Друга функція – порівняння однорідних товарів за параметрами (наприклад, таблиця порівняння програмного продукту “1С-Бітрікс” – <http://www.1cbitrix.ru/products/cms/editions/#tab-table-link>). Дуже зручно при виборі потрібного продукту, а значить, допомагає потенційному клієнтові стати реальним.

- Третя функція – функціонал для демонстрації товарів. Важливо дати клієнту можливість по максимуму оцінити товар, для цього, в залежності від товару, підійдуть можливості подивитися фотографії, відео (наприклад, інтернет-магазин з продажу відео “Video-Shop” – http://videoshop.com.ua/catalog/Filmi_DVD_Video/Rozhdestvenskaya_istoriya_DVD_.html пропонує переглянути трейлери продаваних фільмів) і багато іншого. Окремо можна виділити інтерактивне зміна параметрів, хоча це швидше перспектива майбутнього; Зараз подібних прикладів дуже мало, але даний функціонал може бути вельми корисним для користувачів, а зробити його не складно (наприклад, змінити колір у віртуального автомобіля або по заданим параметрам подивитися, як буде виглядати новий плазмовий телевізор на стіні власної кімнати, або певні квітам підібрати поєднуються відтінки для нового товару і т.д.). Допомагає покупцеві прийняти рішення.

- Четверта функція - розширений пошук за різними параметрами. Будь-який товар користувач повинен легко і швидко знайти, чи не риючись в каталозі, а просто ввівши його назву в рядок пошуку, або ж увівши потрібні йому параметри, в цьому випадку показуються схожі за функціями товари (наприклад,

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

розширений пошук в інтернет-магазині “Softkey” – <http://www.softkey.ru/catalog/search.php?adv=Y>).

- П'ята функція - соціальні елементи. Сьогодні інтернет-магазини все більше перетворюються в своєрідні соціальні мережі. Звичайно, не варто робити черговий клон “Facebook” з можливістю здійснювати покупки, але безперечно варто зробити не обов'язкову реєстрацію із заповненням інформації про користувачів (в тому числі інтересів), відгуки про товари, оцінок (наприклад, коментарі та оцінки користувачів є в інтернет-магазині “OZON.ru” – <http://www.ozon.ru/>, які значно впливають на продаж), обговорення і т.д. Ці функції допоможуть при виборі товару і підвищать лояльність покупців в цілому до магазину.

- Шоста функція – спеціальні пропозиції для стимулювання збуту. Вони можуть бути як загальними для всіх покупців, так і персоналізованими, в залежності від інтересів користувача, історії покупок, активності і т.д. Загальні спеціальні пропозиції покликані підвищити попит на конкретні продукти, наприклад, ті, які залежалися на складі, і їх потрібно розпродати. А персоналізовані пропозиції покликані підвищити продажі конкретному покупцеві, наприклад, якщо користувач цікавиться маркетингом, в історії покупок у нього більше 50 % покупок книг цієї категорії, і відвідує він в магазині відповідні сторінки – йому буде не складно продати продукцію цієї або подібної тематики, причому в автоматичному режимі сайт аналізує інформацію і пропонує йому продукти, до яких у нього потенційно буде підвищений інтерес. По суті можна створити цілу аналітичну систему для цієї мети і, теоретично вона може в кілька разів збільшити прибуток інтернет-магазину, адже у всіх є приховані потреби, потрібно тільки докопатися до них.

- Сьома функція - мобільна версія сайту. У століття мобільних пристроїв потрібно обов'язково оптимізувати інтерфейс магазину під них, інакше можна втратити значну частку покупців (наприклад, мобільна версія магазину є у компанії “ТЕХНОТРЕЙД” – <http://tehnotrade.com.ua/>).

- Восьма функція – популярні товари в інтернет-магазині або окремої категорії. Такі товари вже зарекомендували себе, як добре продаються, і є сенс їх виділяти із загальної маси для інших покупців, причому не забувати про різних часових відрізках, іноді виходить, що товар залишається самим популярним цілий рік.

- Дев'ята функція – розсилка по підписці. Важливий елемент, особливо для постійних покупців, які хочуть бути в курсі новинок та акцій або очікують появи товару на складі. Безумовно, щоб не потрапити в папку спаму, потрібно продумати зручний функціонал з налаштування цієї розсилки і дати просту можливість відписатися, а за стандартом вона повинна бути ненав'язлива

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

(наприклад, розсилка є у інтернет-магазині “Walmart” – <http://www.walmart.com/>, правда відписатися від неї проблематично).

- Десята функція – інтеграція інтернет-магазину зі сторонніми системами. Ця функція побічно впливає на продажі, але при цьому значно допомагає роботі магазину. Не завжди, але досить часто потрібно інтеграція з *IC*, *ERP*, *CRM* і т.д. Це не складно зробити, у багатьох коробкових *CMS* є стандартний функціонал для інтеграції з деякими продуктами.

Дизайн. Чи не найголовніший, але досить важливий елемент сучасного інтернетмагазину, на який часто не звертають уваги. Особливо важливо правильно подати потрібну інформацію, часто інтернет-магазини настільки перевантажені інформацією, що знайти потрібний товар стає складно.

Дизайн повинен бути сучасним, легким, повітряним, з мінімальною графікою і без анімації, в даному випадку красива графіка і анімація будуть тільки відволікати користувача від товарів, що продаються, хоча в деяких товарних категоріях графіка цілком допустима. Важливо, щоб дизайн був розрахований на цільову аудиторію магазину, а не замовника (наприклад, хороший інформаційний дизайн у інтернетмагазині “ПАРФЮМЕРІЯ” – <http://www.parfumeria.ua/> без надмірностей в графіку).

Основний упор в дизайні потрібно робити на інтерфейс і юзабіліті, сайт магазину повинен бути зручний для пошуку інформації, правильно доносити цю інформацію користувачеві і мотивувати зробити покупку або зробити запит в магазин. При цьому робити це демократично, не нав'язуючи будь-які дії, а лише надавати можливість їх здійснити.

Верстка. Цьому етапу замовник зазвичай приділяє мало уваги, тому що на перший погляд він нічого особливого не несе в собі. Здавалося б, немає нічого складного: з готового дизайну зверстати шаблон майбутнього магазину. Однак, тут варто звернути увагу на дві пов'язані між собою речі: по-перше, верстка повинна відповідати стандартам W3C (перевірити це можна тут: <http://validator.w3.org/> – потрібно тільки ввести *url* сторінки), а по-друге, сайт повинен правильно відображатися усіма популярними браузерами (перевірити можна тут: <http://browsershots.org/> – зараз найпопулярніші браузери *Internet Explorer* (6,7,8), *Firefox* (3x), *Google Chrome* (3x), *Safari* (4x) і *Opera* (9x, 10x)).

Програмування. Для інтернет-магазину досить великий і складний етап, навіть якщо ми будемо його розробляти на коробковому продукті. Часто застосовують ітераційний підхід, коли спочатку розробляється ядро (система з мінімальним функціоналом), а потім до цього ядра допрацьовуються додаткові функції, таким чином, проект можна запустити в обмеженому функціоналі в будь-який момент після завершення ядра.

На даному етапі варто звернути особливу увагу на безпеку системи в цілому. Якщо використовується платна версія коробочки CMS, наприклад, 1С-Бітрікс – вона вже добре захищена і пройшла багато тестів з безпеки, весь новий функціонал потрібно буде обов'язково окремо протестувати на XSS, SQL injection та ін.

Для великих інтернет-магазинів важливо подбати про продуктивність системи, вони повинні витримувати великі навантаження. Знову ж коробкові CMS вже оптимізовані, а хороший виділений сервер буде працювати стабільно, так що це швидше відноситься до самописний CMS і модулів.

Важливо, щоб весь новий програмний код і особливо нові модулі коментувалися в коді. Для розробника це - правила хорошого тону, а для клієнта - мінімізація ризиків на той випадок, якщо з якихось причин доведеться міняти розробника.

Варто пам'ятати, що створений сайт буде просуватися, в тому числі і в пошукових системах, тому заздалегідь можна врахувати рекомендації по SEO від Google і Яндекс, та й взагалі, заздалегідь підготувати сайт інтернет-магазину до просування.

Тестування. Після того, як сам сайт готовий, його слід протестувати, щоб остаточно підтвердити якість створеного інтернет-магазину.

Для початку потрібно перевірити весь сайт на помилки (баги). Для цього спеціальна людина повинен пройти по сайту і спробувати його використовувати по максимуму, часто робиться разом з клієнтом. Всі знайдені помилки передаються розробникам для виправлення [4].

Потім можна провести тестування навантаження (*Load-testing*) і тестування на стійкість (*Stress-testing*), це робиться за допомогою спеціальних програмних засобів.

І в кінці обов'язкове тестування безпеки, коли спеціально навчений фахівець з безпеки буде намагатися зламати або нашкодити системі, таким чином шукаючи «дірки». Всі знайдені вразливості також передаються на доопрацювання.

Контент. На готовий сайт потрібно завантажити інформацію про продаваних продуктах (тексти, фото, відео і т.д.). Це зазвичай робиться в напівавтоматичному режимі: частина інформації автоматично вивантажується з сторонніх баз даних (наприклад, інформація з 1С), а частина завантажується в ручну через систему адміністрування.

Тексти до продуктів повинні бути створеними спеціально для продажів і, в ідеалі, ще й оптимізованими під пошукові системи.

Таким чином, сайт інтернет-магазину вийде зручним, красивим, буде направляти користувачів, мотивувати на здійснення покупок, готовим до

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

великих навантажень, з підвищеною безпекою, інтегрований в усі зовнішні системи підприємства. Залишилося тільки привести на цей чудовий сайт багато потенційних покупців і «справа в капелюсі», про що і говориться в наступному розділі.

Ринок безумовно є, і він досить значний, клієнтів багато, і вони охоче платять, темпи зростання тільки радують, і прогнози теж хороші. Створити дійсно прибутковий магазин зараз цілком реально, для цього потрібно врахувати три складові чарівної формули: комерційну частину, технічну частину і маркетингову частину. В результаті ми зможемо окупити вкладені інвестиції при оптимістичному прогнозі в перший квартал роботи, а при песимістичному – протягом року. У будь-якому випадку інтернет-магазини – дуже вигідний бізнес з високим коефіцієнтом *ROI*.

Розробка веб-сайту інтернет магазину є загальним рішенням щодо інвестування серед більшості власників бізнесу, роздрібної торгівлі, підприємців та більшості комерційних підприємств, щоб перетворити конкурентні переваги інтернет-покупок у стабільне джерело доходу через професійний інтернет-магазин.

Веб-сайти електронної комерції “Інтернет-магазини” дозволяють споживачам знаходити, порівнювати та купувати бажані товари чи послуги в Інтернеті зручно вдома за кілька хвилин, за кілька кліків за допомогою своїх смартфонів, планшетів та ПК, які приймають рішення інвестувати час, зусилля та гроші на веб-сайт електронної комерції розробити процес прийняття раціонального рішення щодо збільшення вашого бізнесу за рахунок збільшення продажів та збільшення доходів.

На додаток до зручного досвіду онлайн-покупок, існує безліч інших конкурентних переваг, пропонованих веб-сайтами інтернет магазинами, які включають, але не обмежуючись ними:

1. Збільшення охоплення бізнесу за допомогою націлювання на різні місцеві та міжнародні ринки
2. Залучення більшої кількості потенційних клієнтів та збільшення обсягу продажів завдяки професійному інтернет-магазину, який працює цілодобово та без вихідних
3. Зниження операційних та маркетингових витрат, необхідних для ведення та просування фізичного магазину
4. Відстеження та аналіз поведінки споживачів “інтересів відвідувачів вебсайту інтернет магазину” для налаштування вашої продукції та маркетингових кампаній
5. Створення більше можливостей для продажу та перехресних продажів за допомогою кампаній ремаркетингу.

Що стосується переваг розробки веб-сайту електронної комерції, щодня з'являється безліч ділових можливостей, коли з'являється професійний інтернетмагазин, налаштований відповідно до бізнес-потреб та цілей.

Сучасні технології розробки інтернет-магазинів

Інтернет-магазин є складним, динамічним веб-додатком. На даний момент існують безліч інструментів і технологій, що дозволяють реалізовувати такі програми. Інтернет-магазин можна реалізувати на таких мовах програмування, як *JavaScript, PHP, Java, Python, HTML, C, C++* і т.п.

Для зручної і спрощеної розробки інтернет-магазину існує клас систем управління контентом: широко використовуються системи *CMS (Content Management Systems)* [2].

CMS надають стандартні і базові набори певних функцій, придатних для створення сайтів. До того ж *CMS* мають безліч плагінів, готових до використання.

Як і будь-який інший програмний продукт, системи автоматизованого керування вмістом існують платні і безкоштовні, з відкритим і закритим вихідним кодом. Переваги платних систем складаються в постійній технічній підтримці з боку розробників, регулярних оновленнях продукту, а також більш високого ступеня безпеки в порівнянні з безкоштовними продуктами. У той же час системи, розповсюджуються безкоштовно, компенсують головний недолік платних *CMS* - їх вартість.

Переваги, які дає використання *CMS*, зручно розглянути під різними кутами: з точок зору розробника сайту і їх користувачів.

З позиції розробника сайту: завдяки наявності вже готових модулів, *CMS* дають можливість виробляти зміни в структурі сайтів якісно і оперативно шляхом компонування цих модулів. Використання модулів також забезпечує більш високу якість розробки, оскільки зменшує ймовірність появи помилок в програмному коді.

З точки зору користувача *CMS* пропонують насамперед зручний інтерфейс для редагування вмісту сайту, не вимагаючи основних знань в сфері веб-дизайну, *HTML* верстки і т.п.

На даний момент існує безліч готових систем керування вмістом сайту.

Найбільш відомі *CMS: WordPress, osCommerce, OpenCart та Joomla, Tilda*.

В ході роботи були розглянуті найбільш поширені системи.

1С-Бітрікс – система орієнтована на корпоративні сайти, інформаційні та довідкові портали, соціальні мережі, інтернет-магазинів, сайти ЗМІ, придатна для створення інших видів веб-ресурсів.

Для зберігання даних сайту використовується файлова система сервера і реляційна СУБД. Підтримуються наступні СУБД: *MySQL*, *Oracle*, *MS SQL*. Продукт працює на *Microsoft Windows* і *UNIX*-подібних платформах, включаючи *Linux*. “1СБітрікс: Управління сайтом” продається в однію з восьми складених фірмою розробником редакцій, що визначають функціональність системи і набір модулів.

Кількість модулів у встановлюваної системі залежить від редакції продукту. Головний недолік системи – вона є комерційною.

WordPress – система керування вмістом сайту з відкритим вихідним кодом, яка розповсюджується за ліцензією *GNU GPL*. *WordPress* написана на *PHP*, в якості бази даних використовує *MySQL*.

Сфера застосування – від блогів до досить складних новинних ресурсів і інтернет-магазинів. Вбудована система “тем” і “плагінів” разом з вдалою архітектурою дозволяє конструювати практично будь-які проекти. Система *WordPress* випущена під ліцензією *GPL* версії 2.

Універсальність *WordPress* прихована в сотнях доступних плагінів. Завдяки їм, функціонал цієї *CMS* наблизився до таких систем як “*Joomla*”. Однак, установка занадто великої кількості плагінів може негативно позначитися на швидкості завантаження сторінок.

Типи веб-сайтів інтернет магазинів

Перш ніж обговорювати процес проектування та розробки веб-сайтів інтернет магазинів, потрібно спочатку обговоримо основні типи веб-сайтів інтернет магазинів, щоб визначити найбільш підходящий тип для потреб та цілей перед початком процесу розробки.

Основними типами веб-сайтів інтернет магазинів є:

- *B2C* – бізнес-користувач
- *B2B* – бізнес-бізнес
- *C2C* – користувач-користувач

Інтернет-магазини *B2C*: Поширений тип підприємств електронної комерції, що має веб-сайт електронної комерції, присвячений показу та просуванню товарів для кінцевих споживачів, таких як одяг та технічна продукція, включаючи мобільні телефони, ПК чи меблі.

Платформи електронної комерції *B2B*: Підприємства розробляють платформи електронної комерції в Інтернеті для просування своїх продуктів та послуг для інших підприємств, одночасно збільшуючи їх охоплення та орієнтуючись на міжнародні ринки.

Магазини електронної комерції *C2C*: Сторонні магазини, такі як *eBay*, дозволяють звичайним приватним та роздрібним продавцям просувати та продавати свою продукцію в Інтернеті кінцевим споживачам.

Існують інші типи онлайн-платформ, де приватні особи та спеціалісти пропонують свої послуги для бізнесу, наприклад, платформи-фрілансери, які, як правило, не відповідають критеріям веб-сайту електронної комерції.

Початок бізнесу в галузі електронної комерції – це вибір, який підходить для різних типів комерційних, технологічних та промислових проектів, оскільки дає змогу рекламувати та продавати фізичні товари, а також цифрові товари, такі як онлайнкурси чи електронні книги, або ексклюзивний цифровий вміст, як стокові зображення.

2.2.1 Рекомендації на головній сторінці

Головна сторінка - це перше, що бачить користувач коли відвідує веб-сайт через прямий трафік. Оскільки ці відвідувачі не обов'язково шукають щось конкретне, рекомендації на головній сторінці мають на меті інформування клієнтів про останні пропозиції та знижки та демонстрацію своїх популярних продуктів.

Популярні продукти. Це дуже базова, але потужна логіка рекомендацій, яка чудово працює майже у всіх магазинах електронної комерції. Популярність продукту визначається найлегше за кількістю придбаних вами товарів. Проте, більш досконалі системи включають інші дані (кліки, перегляди, тощо) у свою логіку, щоб подати ще більш точні рекомендації.

У випадку із змістовними сайтами (новинними сайтами, відеопорталами) інші чинники, такі як час, витрачений на сторінку, прокручуваний відсоток, або час перегляду відео, також можуть бути важливими факторами популярності. На веб сайтах, що пропонують певну продукцію ця логіка простіша. На рисунку 2.1 приведено приклад рекомендації бестселлерів [26].

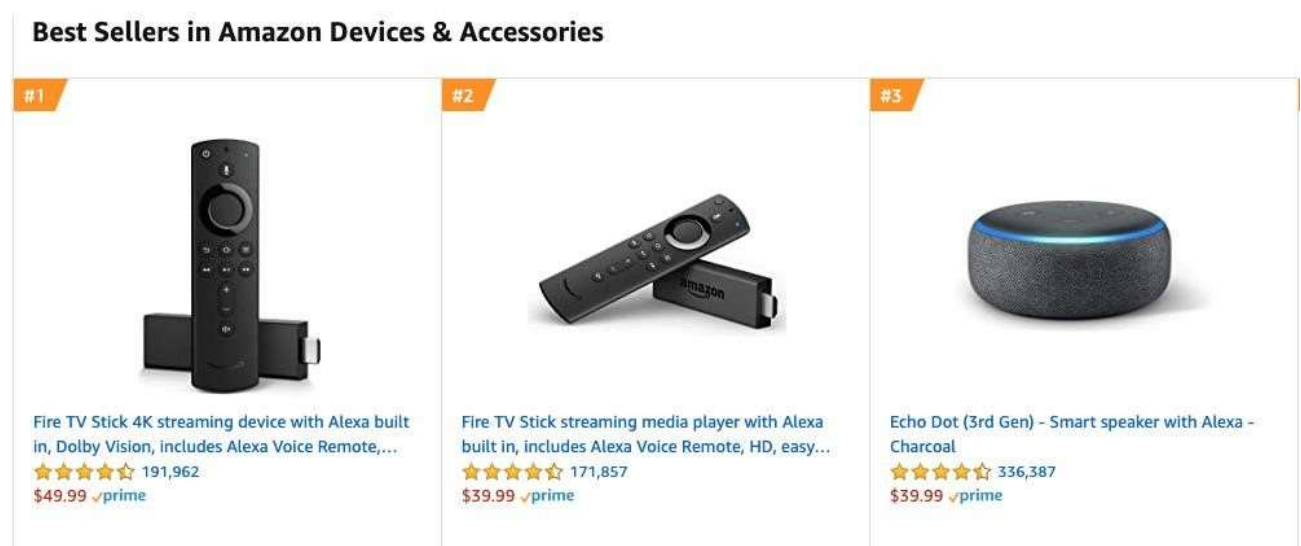


Рисунок. 2.1. Приклад рекомендації бестселерів

Рекомендації, що базуються на рейтингу. Іншим показником популярності може бути високий рейтинг продукту. Під рейтингом мається на увазі — явний рейтинг: оцінки, коментарі. З іншого боку, перегляд опису товару, покупки та перегляди називаються "неявним зворотним зв'язком", оскільки вони представляють точки даних, які створюються, коли користувач природно взаємодіє з платформою чи веб-сайтом. Системи рекомендацій, розгорнуті в реальному світі (а не в академічному), використовують неявні відгуки через велику кількість, багатство та структуру таких даних, на відміну від явних рейтингів, наприклад, які є надзвичайно дефіцитними (лише невеликий відсоток користувачів пише відгуки або навіть дають рейтинги). Тим не менш, хоча і веб-сайт чи платформа, можливо, не мають великої користі від рейтингів, користувачам це допомагає. Насправді, згідно з опитуванням BrightLocal, 88% користувачів довіряють рейтингу так само, як і особистим уподобанням.

Інколи, з огляду на звички та уподобання бази користувачів, відгуки можуть стати вирішальним чинником впливу на рішення про покупку. Тоді, позиції "Найвищий рейтинг" розміщуються також на головній сторінці. На рисунку 2.2 приведено приклад рекомендації товарів з найвищим рейтингом [27].

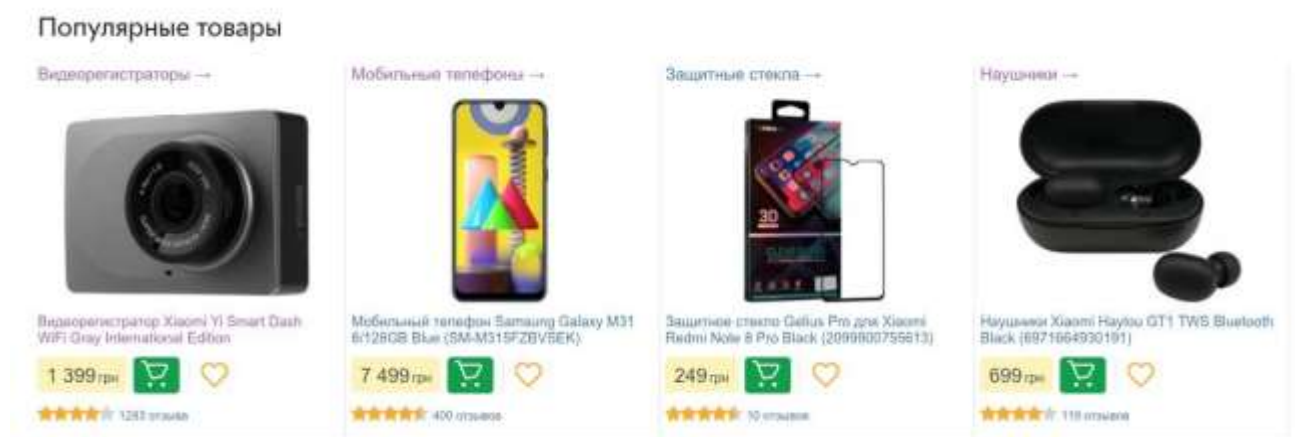


Рисунок. 2.2. Приклад рекомендації товарів з найвищим рейтингом

Персоніфіковані рекомендації. Списки персоналізованих рекомендацій показують різні продукти для кожного користувача, залежно від їх попередньої покупки та історії веб-перегляду. Проте персоналізовані рекомендації потребують значних обсягів поведінкових даних користувачів, яких система не має для нових відвідувачів (це називається проблемою холодного старту РС).

Для того, щоб забезпечити персоналізований користувацький досвід для користувачів які відвідують веб-сайт не вперше, а також задовольняти нових відвідувачів, найкращим рішенням у галузі є визначення "альтернативного сценарію" у випадку, коли про користувача мало що відомо, або невідомо нічого. На практиці це означає, що система виявляє, чи є достатньо даних про певного клієнта для надання йому персоналізованих пропозицій предметів. Якщо даних

немає, система рекомендацій "повернеться" до більш загальних рекомендацій (наприклад, популярні продукти) або логіки, яка фільтрує продукти за категорією, відображаючи в рекомендаціях тільки продукти з категорії, в якій були товари переглянуті користувачем в даній сесії.

Сценарій альтернативних рекомендацій слід планувати з великою обережністю і використовувати максимально ефективні дані для того, щоб забезпечити точні рекомендації для нових відвідувачів. На рисунку 2.3 приведено приклад персоналізованих рекомендацій [26].

Inspired by your Wish List [See more](#)



Рисунок. 2.3. Приклад персоналізованих рекомендацій

2.2.2 Рекомендації на сторінці продукту

Сторінка продукту чи сторінка інформації про продукт - це місце, де відвідувачі можуть знайти детальний опис продукту та його характеристик. На цій сторінці також зазвичай є можливість додати продукт до кошика або замовити його відразу. Основна мета рекомендацій на цих сторінках - показати найбільш релевантні елементи, щоб, таким чином, забезпечити "наступний крок" у пошуку користувачем товарів та заохотити його на наступні покупки на веб-сайті.

У більшості випадків, чим більше часу користувачі витрачають у інтернетмагазині, тим вище шанси на те, що вони будуть купувати товари.

Схожі продукти. Віджети зі списком схожих продуктів можуть базуватися на дуже різних даних. Найменш складним є проста фільтрація на основі категорій, яка може бути застосована навіть без рекомендаційного механізму (це, безперечно, відстає в продуктивності). Якщо поєднати цей простий метод фільтрування із подібністю на основі метаданих (описи, заголовки продуктів, теги, ціни тощо), можна значно підвищити продуктивність (наприклад, рекомендувати елементи однієї марки або того самого кольору з поточного категорії), але для цього потрібно буде мати розширені функціональні рекомендації, доступні на сайті. Одним з найефективніших реалізацій на основі

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

подібності є метод, що називається "спільна фільтрація між елементами" (метод, який був започаткований компанією Amazon). На рисунку 2.4 приведено приклад рекомендації схожих продуктів [26].

More items to consider [See more](#)



Рисунок. 2.4. Приклад рекомендації схожих продуктів на сторінці продукту

"Клієнт, який купив / переглянув це також купив" Колаборативна фільтрація. Колаборативна фільтрація в рекомендаціях щодо продуктів електронної комерції була вперше застосована Amazon (вони створили свій початковий патент для колаборативної фільтрації між елементами ще в 1999 році). Ідея колаборативної фільтрації проста - якщо користувач здійснював покупки товару або перегляди контенту, знайдемо користувачів зі схожими смаками, і порекомендувати нашому клієнтові те, що схожі на нього люди споживали, а клієнт ще ні. Це User-Based підхід.

Аналогічним чином можна подивитися на завдання з точки зору товару, і підібрати компліментарні товари до кошика клієнта, підвищивши середній чек, або замінивши відсутні на складі товари на аналог. Це Item-Based підхід[3]. На рисунку

2.5 приведено приклад колаборативної фільтрації [26].

Customers who bought this item also bought



Рисунок. 2.5. Приклад рекомендації колаборативної фільтрації

Item-to-item collaborative filtering рекомендації: Цей тип логіки рекомендацій в основному визначає подібність двох продуктів, переглядаючи, як

					КНУ.РМ.123.22.09.02.АРРС	Арк.
Арк.	№ документа	Підпис	Дата			

часто вони присутні разом у історії перегляду чи покупки користувачів. На практиці віджети з використанням такого роду рекомендацій мають назву, яка пояснює логіку рекомендацій: "Клієнти, які переглядали це, також переглянули ...". Моделі уподобань користувачів, які будуються цим алгоритмом, є дуже достовірними і специфічними до веб-сайту ваших користувачів, оскільки вони визначають кількісно реальні взаємодії між відвідувачами та платформою [8].

Персоналізовані рекомендації. Враховуючи контекст (продукт, який переглядається), разом з історією поточного користувача, рекомендації на сторінках продукту можуть бути персоналізовані ще більше. Для цього сценарію можна застосувати персоналізовані алгоритми колаборативного фільтрування. На рисунку 2.6 приведено приклад персоналізованої рекомендації [26].



Рисунок. 2.6. Приклад персоналізованої рекомендації

Personalized collaborative filtering рекомендації: Найпоширеніший спосіб персоналізованих рекомендацій - фокусуватися на середній подібності продуктів до останньої кількості продуктів, які користувач переглядає. Система використовує такий самий алгоритм знаходження подібності продуктів, але вона не порівнює кожен елемент з якимось одним, а з усіма елементами історії користувача. Зважування вихідного потоку залежно від часу взаємодії зазвичай підвищує точність.

2.2.3 Рекомендації для кошика

Рекомендація подібних або пов'язаних продукти з вмістом кошика відвідувача на сторінці кошика може бути дуже ефективним способом підвищення кількості продажів.

Рекомендації для кошика досягають клієнта в сприятливому психологічному стані, при якому він, швидше за все, вже вирішив здійснити покупку, тому він буде більш готовий відповісти «так» на подальші пропозиції. Існує декілька логічних рекомендацій, які дуже підходять для цього завдання. За дослідженням 2015 року, проведеного в 300 магазинах електронної комерції компанії Braillance та Marketingsherpa, віджети, які рекомендують додаткові

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

продукти до елементів у кошику користувача, були одними з 10 найбільш ефективних типів рекомендацій, базуючись на прибутку від таких рекомендацій.

Рекомендовані аксесуари. Рекомендовані аксесуари для продуктів можуть суттєво збільшити розмір та вартість замовлення на веб-сайті. Більше того, реалізація таких віджетів може бути досить простою, технічно кажучи. Однак, залежно від розміру каталогу та структури категорії, воно може бути більш довгим при втіленні, оскільки дуже важко автоматизувати процес рекомендації сумісних аксесуарів для кожного товару. Тому такі логіки, як правило, задаються вручну пов'язаними між собою категоріями.

Оптимально, рекомендувати одну категорію іншій безпосередньо (наприклад, iPhone -> Аксесуари для iPhone). Це набагато простіше для автоматизації та масштабування, але вимагає надзвичайно продуманого планування структури категорій. Одним з таких легко кодифікованих правил, які можуть визначати аксесуари по відношенню до продукту, є предмети, які часто купуються разом із ним, але коштують значно (в 2, 3, 10 разів) менше. Цей метод часто використовується, коли даних забагато для описування правил вручну. На рисунку 2.7 приведено приклад рекомендації товарів пов'язаних категорій [27].



Рисунок. 2.7. Приклад рекомендації товарів пов'язаних категорій

Часто купується разом. Відображення часто продаваних продуктів на сторінках кошика може бути дуже ефективним. Макет цієї сторінки також є важливим фактором. Якщо у вас є час і ресурси, тестування різних планів та дизайнів А / В може дати чудові результати та статистичні дані. Наприклад, Netflix широко використовує тести А / В для макетів домашнього екрана на різних платформах. На рисунку 2.8 приведено приклад рекомендації товарів які купуються разом з даним товаром [27].

Вместе с этим товаром покупают

Все категории Bluetooth-гарнитуры Держатели Чистящие средства Услуги к электронике TravelSim Док-станции

Смотреть все →

ТОП ПРОДАЖ  Bluetooth-гарнитура Jabra Talk 25 Multipoint (100-92310900-60) 999 грн 854 грн ★★★★★ 237 отзывов	 Держатель для телефона UFT + присоска 2 in 1 Black (lp05black) Заканчивается 398 грн ★★★★★ 112 отзывов	ТОП ПРОДАЖ  Чистящие салфетки ColorWay для очистки LCD/TFT мониторов 100 69 грн ★★★★★ 155 отзывов	ROZETKA  Установка защитного стекла на смартфон или планшет 99 грн ★★★★★ 110 отзывов	 Стартовый пакет TravelSim с пополненным счетом "Проба" 280 грн ★★★★★ 25 отзывов
---	---	---	--	--

Рисунок. 2.8. Пример рекомендації товарів які купуються разом з даним товаром

2.2.4 Рекомендації на сторінці категорій

Коли користувач переглядає сторінку категорії, він вже дав вам цінну інформацію про те, що він шукає. Тут ваша мета повинна полягати в наданні допомоги, тому йому не доведеться переглядати всю категорію, перш ніж надходити до потрібного товару або вмісту. Незважаючи на те, що найкращим варіантом у цьому сценарії є передовий фільтр та графічний пошук, рекомендації також можуть забезпечити цінність у цих випадках.

Нульовий результат і сторінка 404. Перехід до "нульового результату пошуку" або сторінка 404 часто означає завершення сеансу клієнта на сайті та, таким чином, потенційну втрату конверсії. Ці сторінки найчастіше є тими, що мають найвищий рівень виходу, а також однією з небагатьох, де швидкість виходу дійсно є дуже важливим показником.

Поле пошуку на 404 сторінках іноді може компенсувати UX. Відображення рекомендацій персоналізованих продуктів на цих сторінках також може допомогти клієнту повернутися до попередньої сторінки. На рисунку 2.9 приведено приклад списку переглянутих товарів на сторінці 404 [27].

Ошибка 404

Страница не найдена

Неправильно набран адрес или такой страницы на сайте больше не существует.

Перейдите на [главную страницу](#) или выберите подходящую категорию.



Просмотренные товары [Показать все](#) [Сохранить историю просмотра](#)

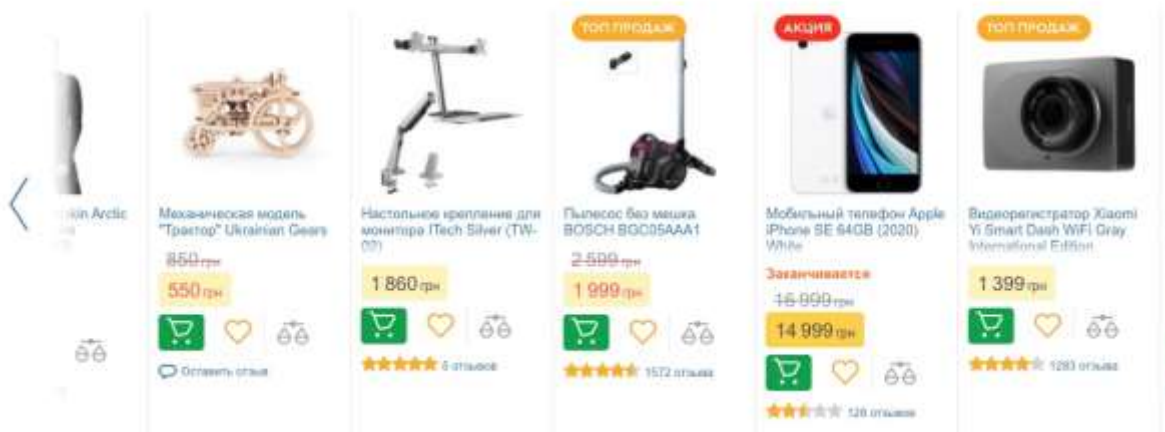


Рисунок. 2.9. Пример списка просмотренных товаров на странице 404

2.3 Метрики рекомендательных систем

Будь-яка рекомендаційна система допомагає вирішувати певну бізнес-задачу. А результат вимірюється особливими для конкретного бізнес-завдання способами: збільшенням кількості відвідувачів, кількості продаж і т.д. Однак якість рекомендаційного алгоритму таким способом виміряти занадто складно, адже бізнесметрики залежать від величезної кількості умов, серед яких рекомендаційний алгоритм сам по собі може виявитися десятою справою. Точність рекомендаційних алгоритмів досить складно оцінити, але існують деякі стандартні метрики, що дозволяють описати точність з математичної точки зору.

Оптимальний підхід, про який піде мова - це спробувати створити метрику безпосередньо під бізнес-завдання. Далі мова піде про конкретний приклад: як це зроблено для рекомендаційного сервісу кінофільмів imhonet.ru. Хоча цей підхід був перевірений саме на фільмах, але він досить загальний, в чому можна переконатися, замінивши слово фільм на що-небудь ще.

В задачі мова йде саме про список рекомендацій, а не про один елемент. Це тому, що система не в змозі отримати всю інформацію про бажані результати. Навіть пошуковик видає список посилань: він не знає повного контексту запиту.

					КНУ.РМ.123.22.09.02.APPC	Арк.
Арк.	№ документа	Підпис	Дата			

Якби він знав повний контекст (наприклад, повну інформацію в голові користувача), то, напевно, єдиний результат мав би сенс, а інакше частина роботи все одно залишається на плечах автора запиту. На рисунку. 2.10 зображено можливі варіанти реакції користувача на рекомендації.

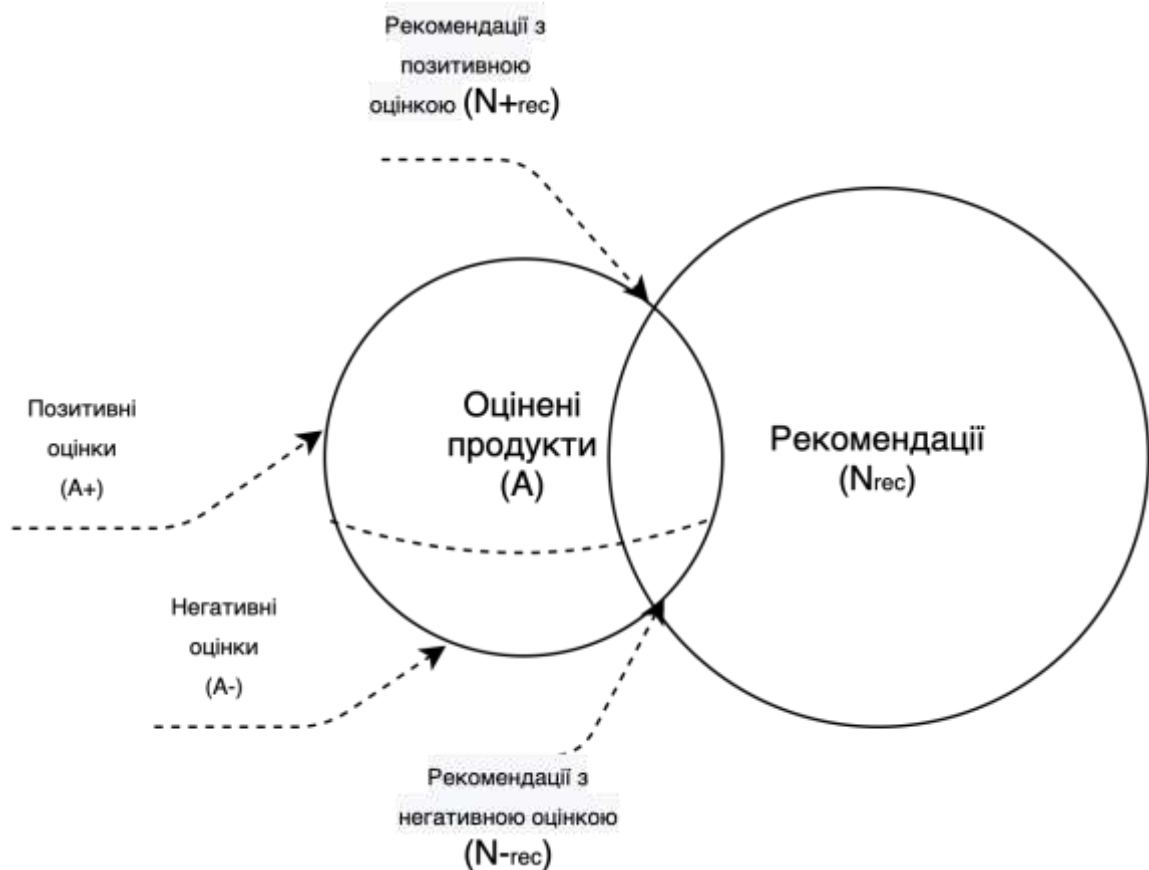


Рисунок. 2.10. Реакція користувача на рекомендації

Розглянемо 4 наступні ситуації. Hit (A+) – фільм рекомендувався і користувач відреагував на нього позитивно. Mishit (A-) – фільм рекомендувався, але реакція користувача на нього була негативною. Recommended but not rated – фільм рекомендувався, але оцінка користувача невідома ($N_{rec} \setminus A$). Rated but not recommended – Користувач позитивно відреагував на фільм, але цей фільм не рекомендувався ($A^+ \setminus N_{rec}$).

Hit. Якщо певний фільм рекомендувався користувачу, після чого користувач залишив позитивну оцінку про фільм, значить, ми поцілили точно. Саме такі випадки нас і цікавлять, тому назвемо їх успіхами і будемо максимізувати їхню кількість.

Mishit. Фільм, який рекомендувався і отримав негативну оцінку є прикладом випадків, яких ми повинні уникати. Можливо навіть, що уникнення таких випадків важливіше, ніж максимізація кількості випадків успіху. Але, практика показує, що такими випадками можна просто знехтувати, оскільки ймовірність збігу негативних оцінок елементів зі списку рекомендацій, яку дає певна РС, виявляється надто незначною, щоб помітно вплинути на значення

метрики. Тому, до негативних оцінок метрика буде нечутлива. Точніше, негативна оцінка буде еквівалентна просто відсутності оцінки у користувача.

Recommended but not rated. Якщо фільм рекомендувався, але позитивного відгуку у нас немає, то здається, що це нічого не означає. Людина могла не подивитися фільм або не оцінити його. На практиці виявляється, що таких випадків більшість. Тому, по-перше, ми завжди повинні рекомендувати з деяким надлишком (тобто взяти значення N_{rec} з запасом), по-друге, більшість людей, як правило, не оцінює кожен переглянутий фільм[14].

Rated but not recommended. Протилежний випадок – користувач позитивно оцінив фільм, який не рекомендувався. Тобто РС упустила можливість збільшити кількість успіхів. Поки такі можливості зустрічаються, можна покращувати точність РС.

Якщо всі елементи рекомендації містять позитивні оцінки, точність системи максимальна - покращувати більше нічого. На рисунку 2.11 зображено діаграму рекомендацій для ідеальної РС.



Рисунок. 2.11 РС з ідеальною точністю

Отже, поліпшення точності рекомендацій - це збільшення кількості успіхів в списку рекомендацій N_{+rec} (розмір якого, N_{rec} фіксований).

Метрика precision

Якщо замість кількості успіхів ми будемо використовувати значення precision (p), тобто розділимо N_{+rec} кількість успіхів на N_{rec} , то, по суті, нічого не зміниться: замість кількості успіхів будемо максимізувати величину, яка відрізняється від нього лише розподілом на константу:

$$p = \frac{N_{+rec}}{N_{rec}} \quad (2.1)$$

Зате, як ми побачимо трохи пізніше, цей поділ дозволить врахувати в метриці дуже суттєвий аспект завдання - залежність метрики від порядку рекомендацій в списку. Крім того, у значення p є зрозуміла логіка, яку можна описати імовірнісною мовою. Припустимо, що людина споживає наш продукт, а саме, переглядає всі елементи списку рекомендацій. Тоді p означає для нього ймовірність знайти в цьому списку потрібний елемент - такий, який задовольнить його в відносному майбутньому. Позначимо $precision$ для користувача u як p_u :

$$p_u = \frac{N_{+rec}(u)}{N_{rec}} \quad (2.2)$$

Доречно узагальнити цю формулу на всю нашу аудиторію (або вимірювану вибірку) користувачів ($Users$):

$$P_{Nrec} = \frac{mean(p_u)}{u \in Users} = \frac{1}{|Users|} * \sum_{u \in Users} \frac{N_{+rec}(u)}{N_{rec}} = \frac{N_{+rec}}{N_{rec} * |Users|} \quad (2.3)$$

Кожен дивиться свій власний список і вибирає те, що йому потрібно, а

P_{Nrec} показує середню ймовірність успіху для всіх випадків. Значення в правій частині - це тепер загальна кількість успіхів у всій вибірці.[14]
Дисконтування та метрика AUC

До сих пір ми оцінювали список елементів як ціле, але ми прекрасно знаємо, що його початок важливіший, ніж хвіст. У всякому разі, якщо список не надто короткий. Метрика, яка це враховує, а значить залежить від порядку елементів у списку, називається дисконтованою. Початок списку важливіше, ніж хвіст, через нерівномірний розподіл уваги користувачів - люди майже завжди дивляться на перший елемент, дещо рідше дивляться і на перший і на другий і т.д. Це означає, що для правильного дисконтування нам потрібна деяка поведінкова модель. Ми уявляємо собі усередненого користувача, який переглядає елементи по черзі, починаючи з першого, а потім, в певний момент, перестає це робити.

Ми не можемо ідентифікувати конкретного користувача, та й не хочемо. Іноді один і той же чоловік захоче переглянути список з 2 елементів, а іноді з 20. Було б непогано роздобути середні ймовірності переходу від кожного елемента до наступного, але нам вистачить даних навіть трохи простіших.

Припустимо, що у нас є ймовірність того, що довільна людина перегляне список з N елементів: w_N для будь-якого розумного N . Тоді можна за формулою повної ймовірності усереднити значення P_N , розуміючи під P_N середнє значення

ймовірності успіху саме для тієї частини нашої аудиторії, яка переглянула список довжиною N . У нашій термінології це буде виглядати так:

$$AUC = \sum_{N_{rec}=1..N} w_{N_{rec}} \cdot P_{N_{rec}} \quad (2.4)$$

Виходить, що значення AUC оцінює середню ймовірність успіху персональних списків рекомендацій в більш реальних умовах - коли списки переглядають живі люди, увагу яких розподілено нерівномірно. Зауважимо, що щоб отримати величину AUC, ми використовували залежність величини $P_{N_{rec}}$ (precision) від N_{rec} - розміру списку рекомендацій, який трохи раніше ми фіксували [14].

2.4 Рекомендаційна система на основі алгоритму CCO (Correlated CrossOccurrence)

У сфері електронної комерції часто явні відгуки користувачів про продукти, або їх враження у іншому вигляді, що були явно сказані є недоступними. Навіть, якщо дані явного відгуку є доступні, їх кількість надзвичайно мала по відношенню до кількості користувачів та кількості продуктів. Кажуть, що матриця КористувачПродукт є розрідженою.

Сфера електронної комерції характерна великими об'ємами даних поведінки користувача. Саме поведінка може найчастіше сказати що користувачу подобається, аніж сам користувач. З даних поведінки користувача в сервісі електронної комерції можна дістати дані неявного відгуку, їх ще називають індикаторами. Індикатор вказує на певний рівень вподобання продукту користувачем з певною ймовірністю. З поведінкових даних користувача можна, як правило, розпізнати дані більш як одного типу індикатора.

Рекомендаційна система, яка правильно використовує дані усіх доступних типів індикаторів вважається якісною.

Розглянуті у цьому розділі гібридні рекомендаційні системи працюють переважно з даними явного відгука, або ж потребують переведення даних явного відгука в дані неявного відгука, що не є тривіальною задачею, коли типів таких відгуків більше ніж 1. У останньому випадку припускають, що кожен тип індикатора має свою вагу, наприклад: купівля – 5, додавання товару до кошика – 3, перегляд – 1. Але це не завжди вірно. Серед таких даних багато шуму, особливо у випадку індикаторів з низьким пріоритетом, наприклад, як перегляд товару. Дані потрібно фільтрувати. Алгоритм Correlated Cross-Occurrence

призначений для створення моделей РС для даних багатьох типів індикаторів. Персоналізовані рокомендації можна обчислити за формулою:

$$r = (P^T P)h_p \quad (2.5)$$

r – рекомендації,

h_p – дані історії користувача певного типу індикатора (купівля продукту, наприклад),

P – матриця історія усіх користувачів головного типу індикатора, рядки є користувачі, кононки є продуктами.

У даній формулі вираз $(P^T P)$ порівнює продукти між собою використовуючи (Log Likelihood Ratio) LLR кореляційний тест. Назвемо матрицю $(P^T P)$ матрицею індикатора головної події (наприклад, купівля). Рядками та колонками у цій матриці є продукти, а елементами similation/correlation оцінка. LLR знаходить аномальні патерни купівлі продуктів разом, фільтруючи події з історії користувача, які не надають користі в обчисленні рекомендацій.

Вираз вище можна інтерпретувати наступним чином. Береться історія користувача h_p та порівнюється з рядками матриці $(P^T P)$. Далі TF-IDF підхід може бути використано для зменшення впливу занадто популярних продуктів. Знаходяться найближчі елементи до елементів у історії користувача. Потрібно відсортувати ці елементи – продукти за значенням подібності, та вибрати перших k елементів. Це і є рекомендації.

2.5 Інформаційна модель інтернет-магазину з адаптивною пропозицією товарів

Розробка інформаційної моделі інтернет-магазину з адаптивною пропозицією товарів потрібна для подальшої роботи з рядом процесів та сутностей, які формують предметну область інтернет-магазину з семантичним аналізом профілів товарів, що потрібно для розробки методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками, який дозволяє за значеннями оцінок сумарного лінгвістичного збігу профілів товарів та сумарного семантичного збігу профілів товарів виконувати формування множин подібних товарів відповідно до обраного користувачем товару.

Інформаційна модель інтернет-магазину з адаптивною пропозицією товарів містить формальне подання всіх необхідних сутностей для адаптивної пропозиції товарів за семантичними ознаками користувачам інтернет-магазину.

Отож, інформаційна модель інтернет-магазину з адаптивною пропозицією товарів ІМ має наступний вигляд:

					КНУ.РМ.123.22.09.02.АРРС	Арк.
	Арк.	№ документа	Підпис	Дата		

$$IM = \langle T, CT, O, SO, U, TU, P, G, SG, W, WD, WK, WKT, WDT \rangle, (2.6)$$

де T – множина товарів визначених категорій ct , CT – множина категорій товарів t , O – множина замовлень, SO – множина статусів замовлень o , U – множина користувачів, TU – множина типів користувачів u , P – множина постачальників, G – множина доставок, SG – множина статусів доставок g , W – множина слів профілів товарів t , WD – множина оцінок важливості слів у описах профілів товарів t , WK – множина оцінок зустрічованості слів у описах профілів товарів t , WKT – множина оцінок подібності пар товарів за оцінками сумарних лінгвістичних збігів профілів товарів wk , WDT – множина оцінок подібності пар товарів за оцінками сумарних семантичних збігів профілів товарів wd .

Отже, розроблена інформаційна модель інтернет-магазину з адаптивною пропозицією товарів містить формальне подання всіх необхідних сутностей для адаптивної пропозиції товарів за семантичними ознаками користувачам інтернет-магазину.

2.6 Схема адаптивної пропозиції товарів у інтернетмагазині за семантичними ознаками

Метод адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками дозволяє за значеннями оцінки сумарного лінгвістичного збігу профілів товарів, обрахованої за порівняннями кількостей появ ідентичних слів, та оцінки сумарного семантичного збігу профілів товарів, обрахованої з використанням дисперсійного оцінювання позицій появ слів, виконувати формування множин подібних товарів відповідно до обраного користувачем товару. На Рисунку 2.12 зображено схему кроків методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками.

Вхідні дані складає множина профілів товарів інтернет-магазину із їх описами, обраний користувачем товар (чи товари) для пошуку подібних товарів та потрібний кількісний обсяг множини подібних товарів m .

На Кроці 1 для кожного товару з числа обраних виконується визначення кількостей появ та оцінок дисперсії слів. Для цього спершу виконується перетворення опису профілю товару в упорядковану множину слів, за якою відбувається формування множини унікальних слів для товару. Потім для кожного слова із множини унікальних слів виконується визначення кількостей появ кожного слова із множини. Окремо для подальшого обрахунку оцінок дисперсії відбувається визначення позицій появ кожного слова із множини. Після цього виконується обрахунок оцінок дисперсії для кожного слова із множини унікальних слів.

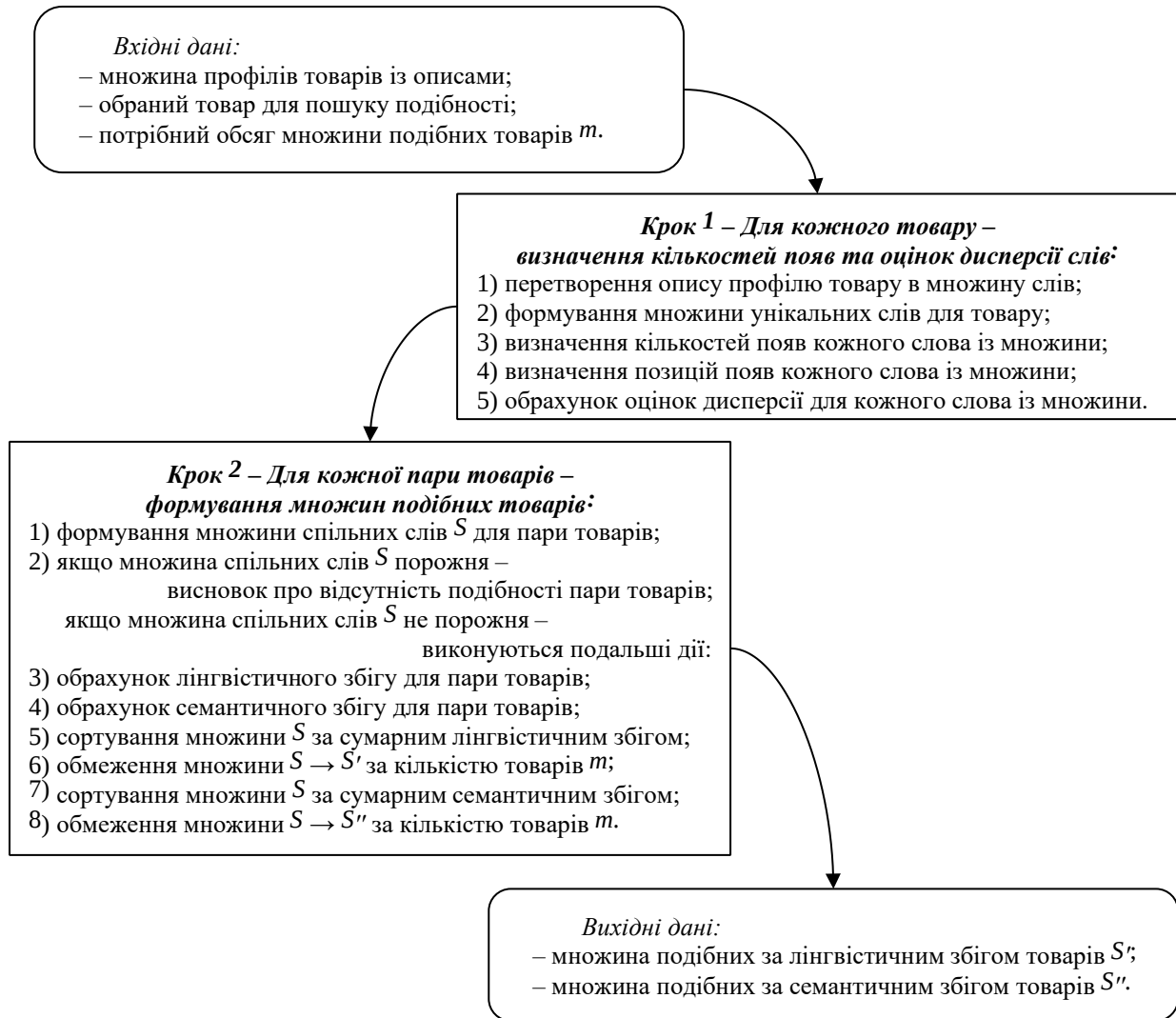


Рисунок 2.12 – Схема методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками

На Кроці 2 методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками для кожної пари товарів виконується формування множин подібних товарів. Для цього спершу порівнюються множини унікальних слів описів цих товарів, за результатом чого відбувається формування множини спільних слів S для пари товарів. За результатом, якщо множина спільних слів S порожня то формується висновок про відсутність подібності пари товарів. Якщо ж множина спільних слів S не порожня, то виконуються подальші дії.

Зокрема спершу проводиться обрахунок лінгвістичного збігу для пари товарів. Після цього виконується обрахунок семантичного збігу для пари товарів. За результатом обрахунку лінгвістичного збігу для пари товарів виконується сортування множини S за сумарним лінгвістичним збігом й відбувається обмеження цієї множини $S \rightarrow S'$ за кількістю товарів m . За результатом обрахунку семантичного збігу для пари товарів, виконується сортування множини S за сумарним семантичним збігом й відбувається обмеження множини $S \rightarrow S''$ за кількістю товарів m . Це визначає вихідні дані методу.

Вихідні дані методу адаптивної пропозиції товарів у інтернетмагазині за семантичними ознаками складають множина подібних за лінгвістичним збігом товарів S' та множина подібних за семантичним збігом товарів S'' .

Таким чином, розроблено метод адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками, який дозволяє за значеннями оцінки сумарного лінгвістичного збігу профілів товарів, обрахованої за порівняннями кількостей появ ідентичних слів, й оцінки сумарного семантичного збігу профілів товарів, обрахованої із використанням дисперсійного оцінювання позицій появ слів, виконувати формування множин подібних товарів відповідно до обраного користувачем товару.

2.7 Функції інформаційної системи автоматизованого пошуку ключових семантичних одиниць у цифрових текстах

Для прикладного дослідження розробленого методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками необхідно створити інформаційну систему автоматизованої пропозиції товарів за семантичними ознаками на основі інтернет-магазину, яка використовує за призначенням даний метод та дозволяє за значеннями оцінки сумарного лінгвістичного збігу профілів товарів, обрахованої за порівняннями кількостей появ ідентичних слів, та оцінки сумарного семантичного збігу профілів товарів, обрахованої з використанням дисперсійного оцінювання позицій появ слів, виконувати формування множин подібних товарів відповідно до обраного користувачем інтернет-магазину товару й забезпечує відповідний необхідний функціонал.

Інформаційна система автоматизованої пропозиції товарів за семантичними ознаками на основі інтернет-магазину має надавати можливість за відомостями обраного користувачем товару, описами наявних товарів та кількістю необхідних пропозицій автоматизовано одержувати дві множини подібних до обраного користувачем товарів, одна з яких формується за оцінками сумарних лінгвістичних збігів профілів товарів, а друга – за оцінками сумарних семантичних збігів профілів товарів.

Відповідно до п.2.2, розроблена з використанням методу адаптивної пропозиції товарів інформаційна система автоматизованої пропозиції товарів за семантичними ознаками на основі інтернет-магазину мусить виконувати наступні функції:

робота адміністратора інтернет-магазину з прайсом товарів (перегляд переліку, додавання, редагування та відмітки наявності товарів) ;

робота адміністратора інтернет-магазину з даними профілю (перегляд переліку, докладний перегляд обраного профілю, активація профілів);

робота адміністратора інтернет-магазину з замовленнями (перегляд переліку замовлень, пошук, редагування, зміна статусів);

робота профілю користувача інтернет-магазину з даними профілю (реєстрація, авторизація, редагування даних профілю);

робота профілю користувача інтернет-магазину з товарами (перегляд переліку, докладний перегляд обраного товару, пошук);

робота профілю користувача інтернет-магазину з замовленнями (переміщення товарів у кошик, редагування вмісту кошику, оформлення замовлення);

автоматизоване формування множини слів для кожного товару;

автоматизоване визначення кількостей появ та оцінок дисперсії для кожного слова із множини;

автоматизоване формування множини спільних слів для кожної пари товарів;

автоматизований обрахунок лінгвістичного та семантичного збігу для кожної пари товарів;

автоматизований обрахунок сумарних лінгвістичного та семантичного збігів для пар товарів;

автоматизоване сортування за сумарним лінгвістичним збігом, обмеження кількості подібних товарів;

автоматизоване сортування за сумарним семантичним збігом, обмеження кількості подібних товарів;

автоматизоване закріплення за товарами множин подібних товарів.

Таким чином, наведено функції, які має виконувати інформаційна система автоматизованої пропозиції товарів за семантичними ознаками на основі інтернет-магазину для прикладного дослідження розробленого методу адаптивної пропозиції товарів. Розроблена інформаційна система використовує даний метод та дозволяє за значеннями оцінки сумарного лінгвістичного збігу профілів товарів та оцінки сумарного семантичного збігу профілів товарів виконувати формування множин подібних товарів відповідно до обраного користувачем інтернет-магазину товару й забезпечує відповідний необхідний функціонал.

За коректного виконання інформаційною системою визначених функцій, можна робити висновок щодо придатності методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками до прикладного використання.

2.8 Математичні моделі обрахунку сумарного лінгвістичного збігу та сумарного семантичного збігу описів профілів товарів

Математичні моделі обрахунку сумарного лінгвістичного збігу та сумарного семантичного збігу описів профілів пари товарів у інтернетмагазині призначені для використання у методі адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками й дозволяють визначати числові оцінки подібності двох товарів за результатом порівняння відповідно кількостей появ та оцінок дисперсії кожного слова із множин унікальних слів описів профілів цих товарів.

Сумарний лінгвістичний збіг K обраної пари описів профілів товарів залежить від кількості появ слів, що співпали у двох описах товарів:

$$K = \sum_{i=1}^n K_i, \quad K_i = \frac{1}{1 + |M_{i,1} - M_{i,2}|} \quad (2.7)$$

де n – кількість оригінальних слів i що співпали, K_i – лінгвістичний збіг слова i в описах товарів №1 та №2, $M_{i,1}$ – кількість появ слова i в описі товару №1, $M_{i,2}$ – кількість появ слова i в описі товару №2.

При повному співпадинні кількості появ слова i в описі обох товарів, буде визначено лінгвістичний збіг слова $K_i = 1$.

Сумарний семантичний збіг D обраної пари описів профілів товарів залежить від оцінки семантичної важливості слів, що співпали у двох описах товарів:

$$D = \sum_{i=1}^n D_i, \quad D_i = \frac{1}{1 + |DE_{i,1} - DE_{i,2}|} \quad (2.8)$$

де n – кількість оригінальних слів i що співпали, D_i – семантичний збіг слова i в описі 2, $DE_{i,1}$ – семантична важливість слова i в описі товару №1, $DE_{i,2}$ – семантична важливість слова i в описі товару №2.

При повному співпадинні оцінки семантичної важливості слова i в описі обох товарів, буде визначено семантичний збіг слова $D_i = 1$.

Використана у (2.3) семантична важливість слова в описі товару DE (Disperse Evaluation) розраховується наступним так [19, 20]:

$$DE_A = \frac{\sqrt{(\Delta A^2) - (\Delta A)^2}}{(\Delta A)}, \quad (2.9)$$

де ΔA – середнє значення послідовності $\Delta A_1, \Delta A_2, \Delta A_k, (\Delta A_2)$ – середнє значення послідовності $\Delta A_{12}, \Delta A_{22}, \Delta A_{k2}$.

При цьому A є словом й у даному випадку позначається як Ank , де індекс $k = 1, 2, \dots, k$ це порядковий номером появи цього слова у тексті, n – номер цього слова в тексті.

Таким чином, розроблені математичні моделі обрахунку сумарного лінгвістичного збігу і сумарного семантичного збігу описів профілів пари товарів в інтернет-магазині призначені для використання у методі адаптивної пропозиції товарів в інтернет-магазині за семантичними ознаками й дозволяють визначати числові оцінки подібності двох товарів за результатом порівняння відповідно кількостей появ й оцінок дисперсії кожного слова із множин унікальних слів описів профілів цих товарів.

Висновки за розділом

В розділі розроблено інформаційну модель інтернет-магазину з адаптивною пропозицією товарів, яка містить формальне подання всіх необхідних сутностей для адаптивної пропозиції товарів за семантичними ознаками користувачам інтернет-магазину.

Наведено функції, які має виконувати інформаційна система автоматизованої пропозиції товарів за семантичними ознаками на основі інтернет-магазину для прикладного дослідження розробленого методу адаптивної пропозиції товарів. Розроблена інформаційна система використовує даний метод та дозволяє за значеннями оцінки сумарного лінгвістичного збігу профілів товарів та оцінки сумарного семантичного збігу профілів товарів виконувати формування множин подібних товарів відповідно до обраного користувачем інтернет-магазину товару й забезпечує відповідний необхідний функціонал. За коректного виконання інформаційною системою визначених функцій, можна робити висновок щодо придатності методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками до прикладного використання.

3. РЕКОМЕНДАЦІЙНИЙ СЕРВІС ДЛЯ ЕЛЕКТРОННОЇ КОМЕРЦІЇ

3.1 Опис технологій, що використовуються

Вимоги до сучасних РС досить високі, адже об'єм даних, що має оброблюватися неймовірно великий, і, разом з тим, користувач не буде задоволений, якщо йому треба чекати кілька днів для отримання рекомендацій. Система має бути **масштабованою**, тобто, придатною для її використання з великими об'ємами даних. За рахунок використання багатьох індикаторів **якості рекомендацій**, які надає система значно підвищується. Ще одним критерієм є **швидке отримання рекомендацій**. Велика кількість запитів отримання рекомендацій надходить до РС на одиницю часу, усі запити мають обслуговуватися в режимі реального часу, щоб користувач мав позитивний досвід користування сервісом електронної комерції.

У даному розділі описується програмний продукт який являє собою РС що задовольняє вищевказаним потребам.

На ринку з'являється все більше і більше open-source технологій для вирішення задач збереження, обробки та пошуку даних. Це надає можливість не винаходити велосипед при вирішенні нової технічної задачі. Нові рішення потребують багато ресурсів та у випадку з невеликими компаніями не завжди будуть кращі за існуючі. Правильнішим підходом буде розбити таку задачу на підзадачі, та використовувати відповідні існуючі технології для вирішення підзадач де це можливо. Реалізацію РС можна розділити на декілька підсистем:

Система збору вхідних даних. Як було описано у розділі 1 РС має вхідний інтерфейс та вихідний інтерфейс. Ця система має реалізувати вхідний інтерфейс РС користувача, у нашому випадку вхідними даними являються індикатори подій користувача пов'язаних з певним продуктом. Система збору вхідних даних повинна мати інтерфейс для завантаження одиничних індикаторів подій та для великого набору таких даних. Система має швидко поглинати дані та надавати результат операції. Для вхідних інтерфейсів РС часто використовують пошукові движки, наприклад Elastic Search, або Apache Solr. Вони призначені для обробки великої кількості запитів та гарно масштабуються. У проекті для збереження індикаторів взаємодії використовується Elasticsearch.

					КНУ.РМ.123.22.09.03.РСЕК			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Коржов.				РЕКОМЕНДАЦІЙНИЙ СЕРВІС ДЛЯ ЕЛЕКТРОННОЇ КОМЕРЦІЇ	Літера	Аркуш	Аркушів
Перевірив	Кумченко							
Н.контроль	Кузнєцов					КІ-21м		
Затвердив	Купін							

Система обробки даних (тренування моделі). Обробка даних полягає в перетворенні даних індикаторів взаємодії в рекомендації. Існують алгоритми які вирішують цю задачу в режимі реального часу, але на великих об'ємах даних такі алгоритми втрачають свою швидкість або точність. Переважно використовують спосіб тренування моделей офлайн. У проекті також використаний офлайн підхід тренування моделі. Для обчислень використовується *алгоритм спільного входження* з бібліотеки Apache Mahout, який читає вхідні дані з HDFS та виконується на розподіленій системі обчислень Apache Spark.

Система отримання рекомендацій. Результати алгоритму спільного входження записуються в Elasticsearch який слугує також і вихідним інтерфейсом PC користувача. Розглянемо основні сторонні технології, які використовуються у проекті.

Пошуковий движок Elastic Search

Elasticsearch - це швидкий, горизонтально масштабований і безкоштовний пошуковий движок (гібрид NoSQL бази даних наділеної багатьма мовами запитів для неї). Користувач взаємодіє з сервісом через REST API. Elasticsearch сервіс представляє собою кластер, що складається з вузлів (нод). Дані зберігаються в індексах (які являються аналогом бази даних) та типах (які являються аналогом таблиць). Елементом даних є документ, який представляє собою JSON документ. Знаючи які пошукові запити даних будуть виконуватися користувач може задати правильний мапінг для кожного поля документа, що допоможе досягнути великої швидкості відповіді. Процес збереження документів у Elasticsearch називається індексацією. Кожен вузол Elasticsearch кластера бере участь у функціях індексації та приєднання до кластера, тобто вузол буде брати участь у певному пошуковому запиті шляхом пошуку даних, які він зберігає. Кожен вузол кластера може обробляти запити HTTP для клієнтів, які хочуть надіслати запит до кластера. Це робиться за допомогою HTTP REST API, який кластер викриває. Даний вузол потім отримує цей запит і несе відповідальність за координування решти роботи. Крім того, даний вузол в рамках кластера знає про кожен вузол кластера і здатний пересилати запити до даного вузла за допомогою транспортного шару, тоді як рівень HTTP використовується виключно для спілкування з зовнішніми клієнтами. Всі вузли за замовчуванням приймають запити HTTP від клієнтів. Кожен вузол також може бути призначений як, так званий, основний вузол за замовчуванням. Головний вузол - це вузол, який відповідає за координацію змін у кластері, наприклад, додавання або видалення вузлів, створення чи видалення індексів тощо. Цей основний вузол оновлює стан кластера.[19]

Система розподілених обчислень Apache Spark

Apache Spark (від англ. Spark - іскра, спалах) - програмний каркас з відкритим вихідним кодом для реалізації розподіленої обробки неструктурованих і слабо структурованих даних, входить в екосистему проектів Hadoop. На відміну від класичного обробника з ядра Hadoop, що реалізує дворівневу концепцію MapReduce з дисковим сховищем, Spark використовує спеціалізовані примітиви для рекурентної обробки в оперативній пам'яті,

					KNU.PM.123.22.09.03.PCEK	Арк.
	Арк.	№ документа	Підпис	Дата		

завдяки чому дозволяє отримувати значний вииграш в швидкості роботи для деяких класів задач, зокрема, можливість багаторазового доступу до завантажених в пам'ять даних робить бібліотеку привабливою для алгоритмів машинного навчання. Проект надає програмні інтерфейси для мов Java,

Scala, Python, R. Спочатку написаний на Scala, згодом додана істотна частина коду на Java для надання можливості написання програм безпосередньо на Java. Spark складається з ядра і кількох розширень, таких як Spark SQL (дозволяє виконувати SQL-запити над даними), Spark Streaming (надбудова для обробки поточкових даних), Spark MLlib (набір бібліотек машинного навчання), GraphX (призначений для розподіленої обробки графів). Spark може працювати в середовищі кластера Hadoop під керуванням YARN та без компонентів ядра Hadoop, він підтримує кілька розподілених систем для зберігання даних - HDFS, OpenStack Swift,

NoSQL-СУБД Cassandra, Amazon S3. [20]

Розподілена файлова система HDFS

HDFS (Hadoop distributed file system) – розподілена файлова система, аналогічна файловій системі Linux, але фізично розгорнута на багатьох серверах. Ідея в тому, щоб можна було зберігати великі об'єми даних на машинах з відносно невеликими характеристиками. HDFS підтримує реплікацію даних. Архітектура системи складається з 3 видів демон процесів: NameNode, DataNode,

SecondaryNameNode. [22] На рисунку 3.1 зображено архітектуру файлової системи HDFS[28].

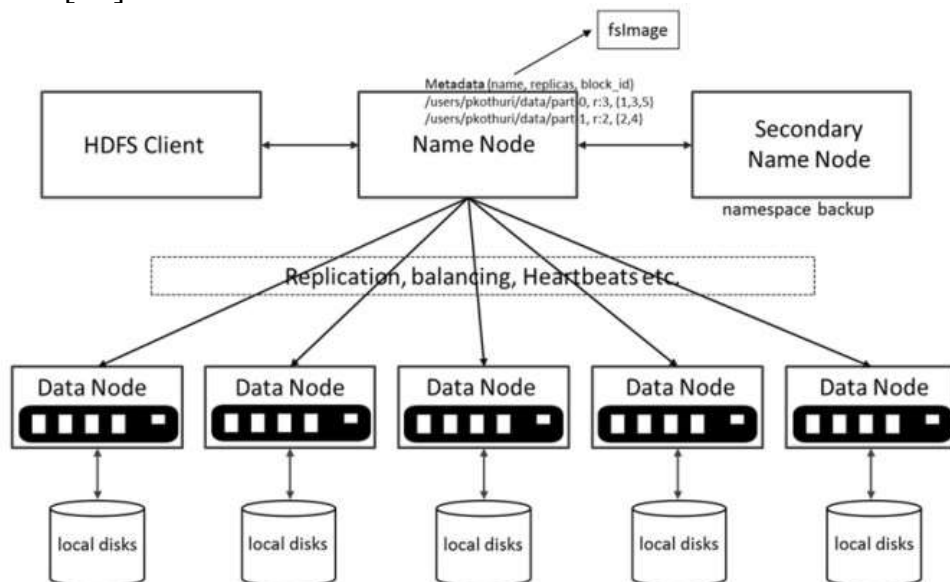


Рисунок. 3.1. Архітектура розподіленої файлової системи HDFS

Кожен файл даних в системі розбивається на блоки. За збереження блоків відповідає процес **DataNode**, а **NameNode** зберігає метадані про кожен файл з даними в системі: кількість та локації блоків кожного файлу даних. **NameNode** являється, так званим, **single point of failure**. Якщо **NameNode** втрачає метадані про файли з будь

яких причин, файли з даними неможливо відновити. Для страхування існує процес SecondaryNameNode, який періодично копіює інформацію з NameNode і зберігає її. SecondaryNameNode може відновити стан NameNode. HDFS Client – це програма, яка відправляє запити по запису чи читанню даних в HDFS, вона звертається напямку до NameNode. [23]

Бібліотека Apache Mahout

Apache Mahout – бібліотека алгоритмів машинного навчання, які виконуються на системах розподілених обчислень таких як Hadoop MapReduce та Apache Spark. Mahout вміщує ряд алгоритмів для реалізації рекомендаційних систем як на Hadoop так і на Spark. [24] Для Spark підтримується 2 алгоритма: spark-itemsimilarity та sparkrowsimilarity.

spark-itemsimilarity – алгоритм колаборативної фільтрації, який на основі неявних взаємодій користувачів з продуктами будує матрицю схожості продуктів між собою. Цікаво, що алгоритм підтримує 2 індикатори взаємодій: основний (наприклад, користувач придбав продукт), та другорядний (наприклад, користувач переглянув деталі продукту). Ідея в тому, що ми хочемо рекомендувати події, що позначаються основним індикатором (купівлю продуктів, наприклад), а події, на які вказує другий індикатор не завжди означають, що користувач позитивно відреагував на продукт. Наприклад, перегляд детального опису продукту не завжди означає, що користувач хоче придбати цей продукт. Алгоритм відфільтровує взаємодії з другого індикатора, які не корелюють з першим індикатором та будує дві матриці подібностей продуктів: за першим індикатором та за другим. Маючи ці дві матриці можна реалізувати якісну РС додавши бізнес логіку поверх отримання рекомендацій з матриць. Цей алгоритм використовується у проекті, описаному в даному розділі. [25] spark-rowsimilarity – алгоритм фільтрації за контентом, який будує матрицю схожих продуктів на основі атрибутів кожного продукту. Підходить для створення рекомендаційних систем текстових документів, книг, музики та інших об'єктів, що не мають чітких атрибутів. [25]

Можливості та характеристики РС

Оскільки сучасні РС – це, безумовно, проекти у сфері Big Data технологій, то архітектура РС була спроектована з можливістю масштабування системи у майбутньому з мінімальними затратами.

Завдяки використанню пошукового движка Elasticsearch РС ми можемо отримати рекомендації в режимі реального часу.

Головною перевагою даної РС перед аналогами є можливість обробки багатьох типів індикаторів, що значно покращує якість рекомендацій. Кожен тип індикатор має пріоритет. Ідея в тому, що РС рекомендує події індикатора найвищого типу, дані індикаторів усіх інших типів фільтруються, залишивши тільки ті події, які мають високу кореляцію з подіями головного індикатора. Для кожного типу індикатора будується модель. Кожна модель може надавати рекомендації для певної підмножини користувачів. Список рекомендацій для користувача формується певним чином, описаним далі у цьому розділі.

3.2 Архітектура програмного продукту

Умовно архітектуру PC можна розділити на 2 компоненти: Online layer та Offline layer. Online layer надає можливість отримувати рекомендації з мінімальною затримкою через REST інтерфейс. [12]

3.2.1 Online рівень

Оскільки зі збільшенням даних збільшується і час тренування моделей, то тренувати нову модель після кожного нового рейтингу, зробленого користувачем, стає неможливим через величезне навантаження. Це потребує розподілення системи на real-time та batch рівні. Real-Time або онлайн рівень означає, що система надає рекомендації в online режимі. Матриця подібності продуктів, модель, створюється кожну добу, періодично, тобто офлайн. Архітектуру системи зображено в додатку В.

Система складається з 4 компонентів, помічених цифрами від 1 до 4 в кільці та інтегрованих сторонніх технологій, помічених цифрами 5, 6, 7 у кільці. Товстими стрілками помічено шлях пересування трансформованих вхідних даних системи. Тонкими стрілками позначено операції пов'язані з обробкою метаданих інформації про вхідні дані. Кожна стрілка позначена буквою англійського алфавіту від —a до —i.

Система має мікросервісну архітектуру, що означає – кожен сервіс виконує свою бізнес-задачу.

Цифрою 1 позначено Менеджер Подій (Event Manager). Це REST сервіс, який слугує вхідним інтерфейсом PC. Цей сервіс спроектовано для обробки даних з великою пропускну здатністю. Користувачі через REST API завантажують нові події (event) в систему. *Подія (event)* – це певна подія в часі, яка сталася на сайті електронної комерції і вказує на вираження інтересу певного рівня користувачем до продукту. Типові події на сайті електронної комерції:

- Користувач **u** переглянув інформацію про продукт **p**;
- Користувач **u** додав до кошика продукт **p**;
- Користувач **u** купив продукт **p**;

Подію можна описати терміном індикатор який був розглянутий у розділі 2. Індикатором у даному випадку являється кортеж **(u, type, p)**, де користувач з ідентифікатором **u** виразив відношення певного рівня **type** до продукту з ідентифікатором **p**. Event Manager індексує отримані події в ElasticSearch, процес індексування зображено стрілкою з буквою **a**. Як було зазначено, система підтримує багато типів індикаторів подій. Перегляд інформації про товар, додавання товару до кошика та купівля товару – це все різні типи індикаторів, які означають різний рівень інтересу до продукту користувачем. Кожен тип індикатора має певний пріоритет. Індикатор з найвищим пріоритетом вважається типом індикатора, події якого система буде рекомендувати користувачам. Усі інші типи індикаторів вважаються другорядними і лише доповнюють опис

					КНУ.РМ.123.22.09.03.РСЕК	Арк.
Арк.	№ документа	Підпис	Дата			

поведінки користувача. Але, варто зазначити, що не всі події індикатора другорядного типу означають, що користувач виявляє інтерес до товару. Наприклад, якщо користувач переглянув інформацію про десятки предметів меблів, це не означає, що в усіх них він зацікавлений і, що він прагне рекомендацій схожих предметів. Можливо користувач просто шукає певний товар переглядаючи багато альтернатив. Саме тому система фільтрує події усіх індикаторів окрім першого з найвищим пріоритетом, залишивши тільки ті події, які мають високу кореляцію з подіями індикатора першого типу. Окрім завантаження подій Event Manager має інтерфейс для конфігурації пріоритету індикаторів. Ці метадані зберігаються в Elasticsearch, що зображено стрілкою з буквою *f*.

Цифрою 5 позначено пошуковий движок Elasticsearch – один із сторонніх сервісів використаних для реалізації проекту. Головні задачі Elasticsearch: зберігати події індикаторів усіх типів, зберігати інформацію про типи індикаторів та їх пріоритет, зберігати моделі, треновані алгоритмом Mahout spark-itemsimilarity для подій типу кожного індикатора.

Цифрою 2 позначено Сервіс Рекомендацій (Recommender), який відповідає за вихідний інтерфейс РС. Цей сервіс спроектовано для обробки даних з великою пропускнуою здатністю. Сервіс представляє собою REST API для отримання рекомендацій. Стрілкою з буквою *d* позначено шлях, через який сервіс отримує метадані про типи індикаторів. Стрілкою з буквою *e* позначено шлях через який Recommender виймає списки рекомендацій отримані від моделі кожного індикатора доти, доки це потрібно. Модель для кожного індикатора зберігається у Elasticsearch у своєму індексі який вміщує ідентифікатор продукту, список рекомендованих цією моделлю ідентифікаторів продуктів для даного продукту та список ваг для таких рекомендацій. Обидва списки відсортовано в порядку спадання.

Приклад:

```
{
  "objectId": "359844",
  "recommendations":["463550", "101075", "333244"]
  "preferences":[19.3165, 18.316588, 17.23454]
}
```

Алгоритм отримання рекомендацій можна описати наступним чином:

```
N // кількість рекомендацій, які треба отримати
user // користувач, для якого треба отримати рекомендації
indicators // список індикаторів відсортований за спаданням пріоритету
recommendations = [] // пустий список рекомендацій

k = 0
#1 Для індикатора indicators[k] отримати N рекомендацій для користувача user
    tmp = getRecommendations[indicators[k]].limit(N) // список рекомендацій від моделі індикатора k
    recommendations += tmp
    якщо рекомендацій для користувача менше ніж потрібно recommendations.size < N
    k += 1
    якщо k == indicators.size =>
        повернути рекомендації recommendations
    інакше =>
        N -= tmp.size
        якщо T
            перейти до #1
```

Рисунок 3.2 Алгоритм отримання рекомендацій

					КНУ.РМ.123.22.09.03.РСЕК	Арк.
	Арк.	№ документа	Підпис	Дата		

Для кожного типу індикатора у відсортованому за спаданням пріоритету списку індикаторів отримувати певну кількість рекомендацій, якщо у всіх моделях для індикаторів з вищим пріоритетом рекомендацій недостатньо. Такий простий алгоритм забезпечує досить високу ефективність, та високу швидкість.

3.2.2 Offline рівень

Асинхронність в комунікації між різними підсистемами великих систем допомагає досягти простоти в реалізації та гнучкості в проектуванні таких систем. Elasticsearch являється проміжною ланкою, що з'єднує online та offline рівні РС. Offline процеси асинхронно читають журнал подій користувача, або дані індикаторів користувачів та тренують моделі, які потім так само асинхронно записуються в Elasticsearch. Online рівень асинхронно записує дані індикаторів користувачів та асинхронно читає рекомендації з тренуваних моделей.

Цифрою 3 позначено сервіс запуску тренування моделей індикаторів Job Runner. Цей сервіс представляє собою REST інтерфейс для запуску процесу тренування. Сторонній сервіс планувальник має відповідальність періодично відправляти запит на тренування моделі. Часто використовують період довжиною в 1 день, тобто моделі оновлюються один раз на добу. Це допомагає знизити навантаження на систему. Буквами *h* та *i* позначена взаємодія Job Runner сервісу з ElasticSearch та задачею тренування моделі, що буде розглянута далі. Ці взаємодії полягають у читанні метаданих про індикатори та запуску процесу тренування, який приймає певні параметри, що залежать від індикатора.

Цифрою 4 позначено компонент системи, який реалізує алгоритм спільного входження, який був розглянутий раніше. Як реалізацію алгоритма було використано Mahout spark-itemsimilarity. spark-itemsimilarity утиліта являється Spark Application, тобто для її запуску потрібно середовище Apache Spark, яке зображено цифрою 7.

Основними параметрами утиліти являються наступні:

- Шлях до вхідного файлу даних індикаторів поведінки користувача
- Шлях, куди записати файли тренуваної моделі
- Назва основного індикатора (наприклад, —purchase|)
- Назва другорядного індикатора (наприклад, —view|)
- Spark master URL, параметр для запуску в середовищі Spark

Вхідними та вихідними шляхами являються директорії у файлової системі HDFS, яка позначена цифрою 6.

Job Runner запускає компонент, позначений цифрою 4, який, в свою чергу, запускає виконання наступних задач:

1. Імпорт даних індикаторів користувачів у папку hdfs:///events. Цей процес виконується в середовищі Spark.
2. Запуск spark-itemsimilarity процесів рівній кількості другорядних індикаторів, або одного такого процесу, якщо другорядних індикаторів нема. Ці процеси виконуються паралельно в середовищі Spark. Результати записуються в директорію hdfs:///model-<ім'я індикатора>/. Результатом кожного такого

					КНУ.РМ.123.22.09.03.РСЕК	Арк.
	Арк.	№ документа	Підпис	Дата		

процесу є 2 матриці: *similarity-matrix* та *cross-similarity-matrix*. Перша матриця представляє матрицю схожості продуктів отриманих методом колаборативної фільтрації від даних головного індикатора. Матриця *cross-similarity-matrix* представляє матрицю схожості продуктів отриману від даних другорядного індикатора, що корелюють між даними основного індикатора.

3. Імпорт матриць в Elasticsearch. Цей процес виконується в середовищі Spark.

Буквою *b* позначено копіювання даних індикаторів користувача в певний час до HDFS. Буквою *c* позначено імпорт цих даних до середовища Spark та їх обробка з допомогою Mahout spark-itemsimilarity. Буквою *d* позначено запис тренуваних моделей – матриць до Elasticsearch.

Інтерфейс сервіса Event Manager

Вхідний інтерфейс PC складається з інтерфейсу для завантаження даних та інтерфейсу конфігурації типів індикаторів та їх пріоритетів.

Нижче зображено REST інтерфейс для конфігурації типів індикаторів та їх пріоритетів.

POST <http://<host>:7777/api/model>

```
Request:
{
  "primaryIndicator": "transaction",
  "secondaryIndicators": [
    {
      "name": "addtocart",
      "priority": 1
    },
    {
      "name": "view",
      "priority": 2
    }
  ]
}
Response:
None
```

Рисунок 3.3 API Event Manager

Поле *primaryIndicator* вказує на назву основного індикатора. Поле *secondaryIndicators* вказує на список другорядних індикаторів, де *name* – ім'я індикатора, а *priority* –

його пріоритет серед інших другорядних індикаторів. Менше значення поля *priority* означає вищий пріоритет.

Отримати конфігурацію індикаторів можна використовуючи наступне API: POST <http://<host>:7777/api/model>

```

Request:
None
Response:
{
  "primaryIndicator": "transaction",
  "secondaryIndicators": [
    {
      "name": "addtocart",
      "priority": 1
    },
    {
      "name": "view",
      "priority": 2
    }
  ]
}

```

Рисунок 3.4 API конфігурації індикаторів

REST інтерфейс для завантаження даних індикаторів має вигляд:
 POST <http://<host>:7777/api/events>

```

Request:
{
  "subjectId": "u1",
  "objectId": "i1",
  "indicator": "view"
}
Response:
None

```

Рисунок 3.5 API завантаження даних індикаторів

Де subjectId – ідентифікатор користувача, objectId – ідентифікатор продукту, indicator – ім'я індикатора події.

Інтерфейс сервіса Recommender

Рекомендації на сторінці продукту показують найкращу конверсію. Вихідний інтерфейс системи має REST API для отримання рекомендацій:

POST <http://<host>:7778/api/recommendation/similarObjects>

```

Request:
{
  "objectId": "i2",
  "size": 10
}
Response:
{
  "objectId": "i2",
  "recommendedObjectIds": [
    "i3",
    "i32",
    "i6",
    "i1",
    "i323",
    "i457",
    "i332",
    "i33",
    "i67"
  ]
}

```

Рисунок 3.6 API сервіса Recommender

У запиті поле `objectId` вказує на ідентифікатор продукту, для якого потрібно отримати схожі продукти в рекомендаціях, а поле `size` вказує на кількість таких рекомендацій. Результат вміщує 2 поля: `objectId` аналогічне полю `objectId` в запиті, поле `recommendedObjectIds` вміщує рекомендовані ідентифікатори продуктів.

Сервіс також дозволяє видалити усі рекомендації за допомогою наступного API:

DELETE <http://<host>:7778/api/recommendation>

Інтерфейс сервіса Job Runner

Для запуску тренування нової моделі система має окремий сервіс, цей сервіс розгорнутий на віртуальній машині, на якій запускаються `bash` скрипти для тренування моделі. При звертанні до наступного API Job Runner запускає скрипти, які в свою чергу запускають Spark Application:

POST <http://<host>:7779/api/model/train>

```

Request:
None
Response:
{
  "tasks": [
    {
      "name": "transaction"
    },
    {
      "name": "addtocart"
    },
    {
      "name": "view"
    }
  ],
  "startedAt": 1541952263340,
  "elapsed": 3,
  "hasFailures": false
}

```

Рисунок 3.7 Job Runner API

					КНУ.РМ.123.22.09.03.РСЕК	Арк.
Арк.	№ документа	Підпис	Дата			

У відповіді сервіс надає статус тренування моделі для кожного типу індикаторів, цей статус позначається у полі `tasks`, де `name` ім'я індикатора. Поле `startedAt` вказує на час в мілісекундах, коли тренування моделей було почато. Поле `elapsed` вказує на час в мілісекундах, який пройшов після запуску тренування моделі. Поле `hasFailures` вказує, чи була помилка тренування якоїсь із моделей. Якщо це поле `true`, то біля поля `name` з'являється поле `message`, де задається причина помилки. Тренування моделі неможливе, якщо уже запущено процес тренування. Коли тренування завершується, то у JSON статусу з'являється поле `isFinished = true`.

Асинхронно отримати статус тренування можна за допомогою наступного API. Цей підхід називається `polling`, коли клієнт періодично відправляє запит для перевірки статусу процесу. GET <http://<host>:7779/api/model/train>

```
Request:
None
Response:
{
  "tasks": [
    {
      "name": "transaction",
      "completedAt": 1541952555813
    },
    {
      "name": "addtocart",
      "completedAt": 1541952556100
    },
    {
      "name": "view"
    }
  ],
  "startedAt": 1541952263340,
  "elapsed": 712853,
  "hasFailures": false
}
```

Рисунок 3.8 API статусу тренування

JSON у відповіді статусу має таку ж саму структуру, як і у відповіді API для тренування моделі.

3.3 Тестування з набором даних поведінки користувача на веб-сайті електронної комерції

Для тестування проекту було використано набір даних, який було зібрано з вебсайту електронної комерції. Цей набір даних можна завантажити з вебсайту Kaggle [11]

Набір даних складається з трьох файлів: файл із даними про дії користувача (`events.csv`), файл із властивостями продуктів з каталогу (`itemproperties.csv`) та файл, який описує дерево категорій (`categorytree.csv`). Дані були зібрані з реального вебсайту електронної комерції. Це необроблені дані, тобто без будь-яких змістових перетворень, проте всі значення хешируються через конфіденційність.

Дані про поведінку, тобто такі події, як кліки, додавання у корзину, покупки, являють собою взаємодії, зібрані протягом 4,5 місяців. Відвідувач може здійснити три типи подій, а саме: «перегляд», «додаток» або «транзакція»

					КНУ.РМ.123.22.09.03.РСЕК	Арк.
Арк.	№ документа	Підпис	Дата			

(покупка). Загалом є 2 756 101 події, включаючи 2 664 312 перегляди, 69 332 додати до візків та 22 457 транзакцій, здійснених 1 407 580 унікальними відвідувачами. Приблизно 90% властивостей подій можна знайти у файлі «item_properties.csv».

Наприклад:

"1439694000000,1, view, 100", означає visitorId = 1, переглянув елемент з id = 100 о 1439694000000 (часова мітка Unix)

"1439694000000,2, транзакція, 1000,234" означає, що відвідувач з Id = 2 придбав товар з id = 1000 в транзакції з id = 234 о 1439694000000 (часова мітка Unix)

Файл із властивостями продуктів (item_properties.csv) включає 20 275 902 рядків, тобто різні властивості, описуючи 417 053 унікальних елементів. Файл розділений на 2 файли через обмеження розміру файлу. Оскільки властивість елемента може змінюватись у часі (наприклад, ціна змінюється з часом), кожен рядок у файлі має відповідну часову позначку. Іншими словами, файл складається з об'єднаних знімків на кожен тиждень у файлі з даними про поведінку. Однак якщо властивість елемента є постійною протягом спостережуваного періоду, у файлі буде присутній лише один момент знімка. Наприклад, у нас є три властивості для одного елемента та 4 знімки на тиждень, як нижче:

timestamp,itemid,property,value 1439694000000,1,100,1000

1439695000000,1,100,1000

1439696000000,1,100,1000

1439697000000,1,100,1000

1439694000000,1,200,1000

1439695000000,1,200,1100

1439696000000,1,200,1200

1439697000000,1,200,1300

1439694000000,1,300,1000

1439695000000,1,300,1000

1439696000000,1,300,1100

1439697000000,1,300,1100

Після злиття знімка це виглядає так:

1439694000000,1,100,1000

1439694000000,1,200,1000

1439695000000,1,200,1100

1439696000000,1,200,1200

1439697000000,1,200,1300

1439694000000,1,300,1000

1439696000000,1,300,1100

Оскільки властивість = 100 є постійною протягом часу, властивість = 200 має різні значення для всіх знімків, властивість = 300 було змінено один раз. Файл властивостей елемента містить стовпчик часових позначок, оскільки всі вони залежать від часу, та властивості можуть змінюватися з часом, наприклад ціна, категорія тощо. Спочатку цей файл складався із знімків кожного тижня у файлі

					КНУ.РМ.123.22.09.03.РСЕК	Арк.
	Арк.	№ документа	Підпис	Дата		

подій і містив понад 200 мільйонів рядків. Таким чином, постійні значення відображатимуться лише один раз у файлі. Ця дія зменшила кількість рядків у 10 разів. Усі значення у файлі "item_properties.csv", виключаючи властивості "categoryid" та "available", були хешіровані. Значення властивості "categoryid" містить ідентифікатор категорії предмета. Значення властивості "available" містить доступність елемента, тобто 1 означає, що елемент був доступний, інакше 0.

Усі числові значення на початку були позначені знаком "n" і мають 3-х точну точність після десяткових знаків, наприклад, " 5 "стане" n5.000 ", "-3.67584 "стане" n3.675 ". Усі слова в текстових значеннях були нормалізовані (процедура Stemming [10]) та хешіровані, цифри оброблялися як вище, напр. текст "Hello world 2020!" стане "24214 44214 n2020.000"

Файл ієрархії категорій має 1669 рядків. Кожен рядок у файлі вказує дочірню categoryId та відповідну батьківську. Наприклад: Рядок "100,200" означає, що categoryId = 100 належить categoryId = 200 Рядок "300" означає, що categoryId не має батьківського вузла. У додатку А можна переглянути проаналізовані дані: товари, що були найчастіше переглянуті або придбані.

Проаналізуємо найбільш популярні товари за різними показниками: покупка, додавання у корзину, перегляд (Рисунок 3.9, 3.10, 3.11)

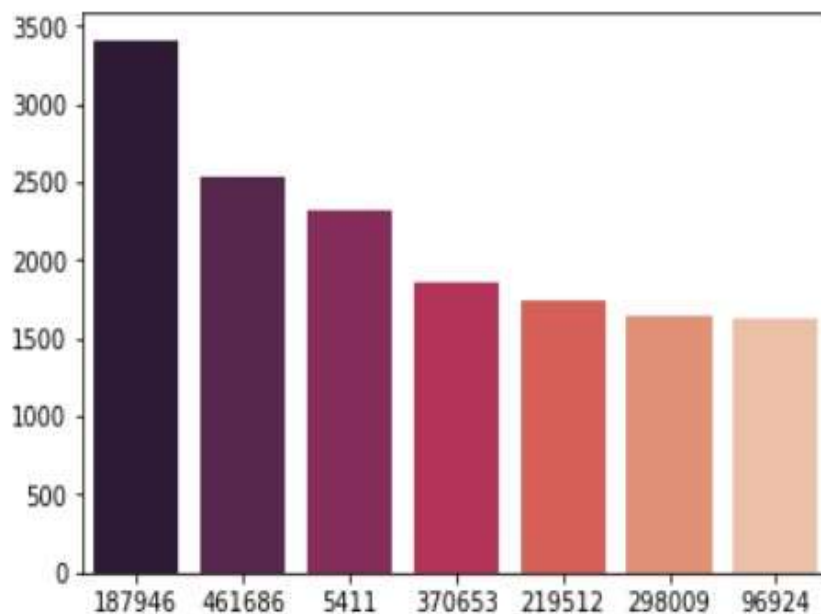


Рисунок 3.9 Позиції, що були найчастіше переглянуті. Вісь x – унікальний ідентифікатор продукту, вісь y – кількість переглядів.

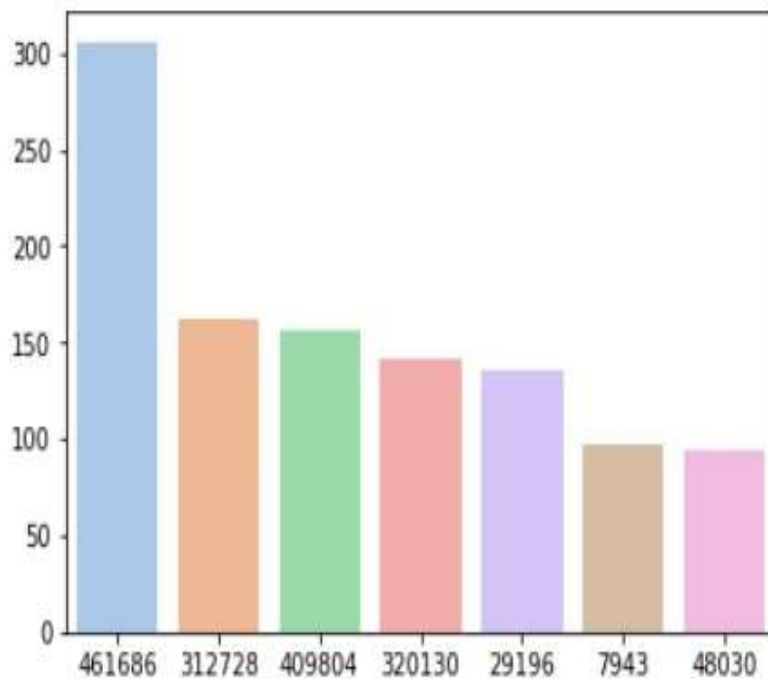


Рисунок 3.10 Позиції, що були найчастіше додані у кошик. Вісь x – унікальний ідентифікатор продукту, вісь y – кількість додавань у кошик.

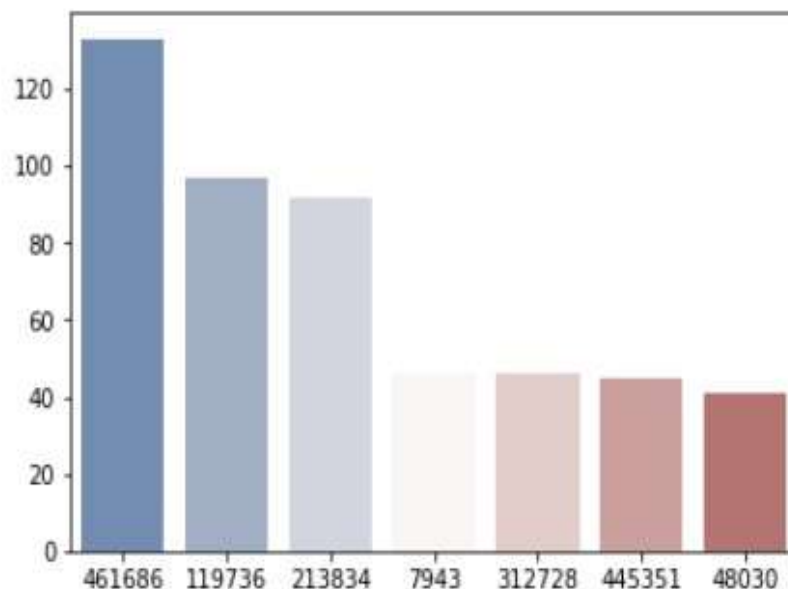


Рисунок 3.11 Позиції, що були найчастіше куплені. Вісь x – унікальний ідентифікатор продукту, вісь y – кількість транзакцій.

Для тестування системи було використано файл events.csv. Поле transactionid було ігноровано. Розмір файла events.csv складає 32.88 МВ. Кількість подій купівлі, додавання до кошика та перегляду інформації про товар складає 2 756 101. [11]

Усі ці події було завантажено в систему через Events Service batch API, описане у попередньому пункті. Для завантаження набору даних

використовуються скрипт додатку А. Завантаження набору даних займає 3 хвилини. Тренування моделі займає 7 хвилин.

Отримуємо 7 рекомендацій для продукту з id 14532:

POST <http://localhost:7778/api/recommendation/similarObjects>

Request: {

“objectId”: “14532”

“size”: 7

}

Response: {

“objectId”: “14532”

“recommendedObjectIds: [

“87453”,

“64562”,

“901462”,

“52802”,

“632356”,

“2452”,

“632”

]

Для перевірки точності РС найкраще підходить А/В тестування, що не входить в рамки даного дипломного проекту.

3.4 Дослідження методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками

Для прикладного дослідження розробленого методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками було створено інформаційну систему автоматизованої пропозиції товарів за семантичними ознаками на основі інтернет-магазину, яка використовує за призначенням даний метод та дозволяє за значеннями оцінки сумарного лінгвістичного збігу профілів товарів, обрахованої за порівняннями кількостей появ ідентичних слів, та оцінки сумарного семантичного збігу профілів товарів, обрахованої з використанням дисперсійного оцінювання позицій появ слів, виконувати формування множин подібних товарів відповідно до обраного користувачем інтернет-магазину товару й забезпечує відповідний необхідний функціонал.

Проведене прикладне дослідження методу адаптивної пропозиції товарів у інтернет-магазині за семантичними ознаками за створеною інформаційною системою в вигляді інтернет-магазину традиційної архітектури із програмним модулем для автоматизованої пропозиції товарів за семантичними ознаками виявило, що створений метод забезпечує можливість автоматизованої адаптивної пропозиції товарів у інтернетмагазині за семантичними та лінгвістичними ознаками описів профілів товарів, яка може бути застосована навіть у випадках відсутності статистики по діях із товарами користувачів інтернет-магазину.

					КНУ.РМ.123.22.09.03.РСЕК	Арк.
	Арк.	№ документа	Підпис	Дата		

Таблиця 3.1 – Ефект від використання методу адаптивної пропозиції товарів для різних випадків діяльності інтернет-магазину

Ситуація	Ефект від використання розробленого методу	Існуючі методи, які можуть бути застосовані одночасно
Відсутня статистика активності користувачів щодо товарів	Використання методу для пропозицій користувачам подібних товарів шляхом підбору товарів із семантично подібними описами властивостей дозволяє компенсувати відсутність статистики активності користувачів щодо товарів	Пропозиція товарів з обраної категорії. Пропозиція товарів із складеного вручну переліку. Пропозиція випадкових товарів з асортименту.
Додано нові товари (локальна відсутність статистики)		
Змінено властивості існуючих товарів (локальна некоректність статистики)		
Є статистика активності користувачів щодо товарів	Використання методу дозволяє покращити можливості інтернетмагазину з пропозицій користувачам подібних товарів шляхом підбору товарів із семантично подібними описами властивостей	Пропозиція товарів згідно статистики переходів попередніх покупців. Пропозиція товарів згідно статистики переглядів попередніх покупців. Пропозиція товарів згідно статистики покупок попередніх покупців.

У Таблиці 3.1 наведено ефект від використання методу адаптивної пропозиції товарів для різних випадків діяльності інтернет-магазину. Виявлено, що найбільший ефект від застосування методу спостерігається у випадках, коли в інтернет-магазині додано нові товари чи змінено властивості існуючих товарів (як наслідок – локальна відсутність статистики), відповідно найбільша ефективність визначається коли взагалі відсутня статистика активності користувачів щодо товарів. В таких випадках використання методу для пропозицій користувачам подібних товарів шляхом підбору товарів із семантично подібними описами властивостей дозволяє компенсувати відсутність статистики активності користувачів щодо цих товарів. Це суттєво

покращує якість пропозицій, на відміну від традиційних підходів у таких випадках, які полягають в пропозиції товарів з обраної категорії, пропозиції товарів із складеного вручну переліку або пропозиції випадкових товарів з асортименту.

У випадку ж, коли в інтернет-магазині наявна статистика активності користувачів щодо товарів, використання методу дозволяє покращити можливості інтернет-магазину з пропозицій користувачам подібних товарів шляхом підбору товарів із семантично подібними описами властивостей. Цей підхід здатний доповнити традиційні підходи, які полягають в пропозиції товарів згідно статистики переходів попередніх покупців, пропозиції товарів згідно статистики переглядів попередніх покупців та пропозиції товарів згідно статистики покупок попередніх покупців.

Підсумовуючи, можна визначити наступні позитивні риси використання розробленого методу адаптивної пропозиції товарів для різних випадків діяльності інтернет-магазину:

□ обрахунок семантичної подібності методом адаптивної пропозиції товарів можна виконувати не кожен раз при зверненні користувача, а один раз для кожної пари товарів, причому це можна виконувати заздалегідь;

□ метод демонструє високу стійкість за коригування прайсу, оскільки додавання чи видалення товарів не впливають вже виконані обрахунки семантичної подібності товарів;

□ інші існуючі методи пропозицій товарів у інтернет-магазинах можуть бути застосовані одночасно з розробленим методом і не протирічать його використанню.

Висновки за розділом

Метриками оцінки якості рекомендацій є порівняння продажів певних товарів до та після запуску системи, визначення відсотку конверсії на запропонованому товарі, відсоток покупок таких продуктів, тощо. Оскільки, особливість рекомендаційної системи саме у тому, що вона використовує різні показники та «неявні» відгуки для визначення найкращого товару для певної особи, повністю оцінити її якість можливо лише при тестуванні на досвіді реальних користувачів

При прикладному застосуванні розробленого методу його позитивною рисою є те, що обрахунок семантичної подібності можна виконувати не кожен раз при зверненні користувача, а один раз для кожної пари товарів, причому це можна виконувати заздалегідь. Метод демонструє високу стійкість за коригування прайсу, оскільки додавання чи видалення товарів не впливають вже виконані обрахунки семантичної подібності товарів. Інші ж існуючі методи пропозицій товарів у інтернет-магазинах можуть бути застосовані одночасно із розробленим методом і не протирічать його використанню.

ВИСНОВКИ

Рекомендаційні системи є системами фільтрації інформації. У сучасні дні складно уявити своє життя без таких помічників.

В роботі було розглянуто основні поняття теорії рекомендаційних систем та функції, які вони виконують. Були визначено типи даних з якими працюють РС: користувачі, продукти, рейтинги. Було описано основні алгоритми, та підходи до обчислення рекомендацій. Було порівняно сучасні методи до обчислення рекомендацій та визначено переваги і недоліки кожного з них.

Було поглиблено розглянуто підходи до побудови РС. Для того, щоб покрити недоліки одного алгоритму перевагами іншого використовують гібридні підходи до обчислення рекомендацій. Також, було визначено точність РС та основні її метрики, такі як: precision, recall та AUC. Для оцінки точності РС потрібно мати механізм для оцінки похибки її результатів. У цьому розділі було запропоновано новий підхід до створення рекомендаційних систем у сфері електронної комерції, який використовує алгоритм ССО.

Реалізація включила у себе проектування системи мікросервісів, які отримують та зберігають дані, виконують запити на тренування моделі та надають готові рекомендації.

За результатом прикладного дослідження виявлено, що значний ефект від застосування методу спостерігається у випадках, коли в інтернетмагазині додано нові товари чи змінено властивості існуючих товарів, відповідно найбільша ефективність визначається тоді, коли взагалі відсутня статистика активності користувачів щодо товарів. Це суттєво покращує якість пропозицій, на відміну від традиційних підходів у таких випадках, які полягають в пропозиції товарів з обраної категорії, пропозиції товарів із складеного вручну переліку або пропозиції випадкових товарів з асортименту. У випадку ж, коли в інтернет-магазині наявна статистика активності користувачів щодо товарів, використання методу дозволяє покращити можливості інтернет-магазину з пропозицій користувачам подібних товарів шляхом підбору товарів із семантично подібними описами властивостей.

Позитивною при прикладному застосуванні розробленого методу рисою є те, що обрахунок семантичної подібності можна виконувати не кожен раз при зверненні користувача, а один раз для кожної пари товарів, причому це можна виконувати заздалегідь. Метод демонструє високу стійкість за коригування прайсу товарів, оскільки додавання чи видалення товарів не впливають вже виконані обрахунки семантичної подібності товарів. Інші ж існуючі методи пропозицій товарів у інтернет-магазинах можуть бути застосовані одночасно з розробленим методом й не протирічать його використанню.

					КНУ.РМ.123.22.09.В			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Коржов				ВИСНОВКИ	Літера	Аркуш	Аркушів
Перевірив	Кумченко							
Н.контроль	Кузнецов					КІ-21м		
Затвердив	Купін							

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Recommender systems introduction. [Електронний ресурс] - Режим доступу до ресурсу: <https://www.coursera.org/learn/recommender-systemsintroduction/home/week/1>
2. Aggarwal C. Recommender Systems / Charu Aggarwal., 2016. – 498 с.
3. Введение в рекомендательные системы [Електронний ресурс] - Режим доступу до ресурсу <https://habr.com/ru/post/476222/>
4. 10 Product Recommendation Techniques to Improve UX and Conversions [Електронний ресурс] - Режим доступу до ресурсу: <https://conversionxl.com/blog/productrecommendations/>
5. The Netflix Prize Rules. [Електронний ресурс] - Режим доступу до ресурсу: <http://www.netflixprize.com/rules.html>
6. Recommender Systems: An Introduction / C.Dietmar Jannach, M. Zanker, A. Felfernig, A. Friedrich. – 2010. – №2. – С. 315.
7. Forbes - How Much Data Do We Create Every Day [Електронний ресурс] – Режим доступу до ресурсу: <https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-createevery-day-the-mind-blowing-stats-everyone-should-read/#3f7a3fe460ba>
8. Hybrid Recommender Systems: Survey and Experiments. [Електронний ресурс] - Режим доступу до ресурсу: <https://pdfs.semanticscholar.org/5880/b9bc3f75f4649b8ec819c3f983a14fca9927.pdf>
9. Approaches, Issues and Challenges in Recommender Systems: A Systematic Review [Електронний ресурс] - Режим доступу до ресурсу: https://www.researchgate.net/publication/312507062_Approaches_Issues_and_Challenges_in_Recommender_Systems_A_Systematic_Review
10. Stemming [Електронний ресурс] – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Stemming>
11. Retailrocket recommender system dataset [Електронний ресурс] – Режим доступу до ресурсу: <https://www.kaggle.com/retailrocket/ecommercedataset#events.csv>
12. Andrew Havens. Beginners guide to creating a REST API [Електронний ресурс]. – Режим доступу: <http://www.andrewhavens.com/posts/20/beginners-guide-to-creating-a-rest-api/>
13. Янг Б. Объектноориентированный анализ и проектирование с примерами приложений. 3е издание [Текст, електронний ресурс] / Бобби Янг, Джим Коналлен, Гради Буч. – М.: Вильямс, 2008. – 720 с.
14. Метрики рекомендательной системы Imhonet.ru. [Електронний ресурс] Режим доступу до ресурсу: <https://habrahabr.ru/company/dca/blog/281066/>
15. McLeod D. Collaborative Filtering for Information Recommendation Systems [Електронний ресурс] / D. McLeod, A. Y. Chen // Research Reports. – 2009.

					КНУ.РМ.123.22.09.СВД	Арк.
	Арк.	№ документа	Підпис	Дата		

- Режим доступу до ресурсу:
http://research.create.usc.edu/cgi/viewcontent.cgi?article=1101&context=nonpublished_reports.
16. Koren Y. Matrix Factorization Techniques for Recommender Systems / Y. Koren, B. Robert, V. Chris. // Journal Computer. – 2009. – №42 (8). – pp. 30–37.
 17. Dunning T. Practical Machine Learning: Innovations in Recommendation / T. Dunning, E. Friedman, A. Felfernig // O'Reilly Media, Inc. , 2014. – 56 с.
 18. Lucene documentation [Електронний ресурс] – Режим доступу до ресурсу: https://lucene.apache.org/core/7_5_0/index.html
 19. Elasticsearch documentation [Електронний ресурс] – Режим доступу до ресурсу: <https://www.elastic.co/guide/index.html>
 20. Elasticsearch Spark support [Електронний ресурс] – Режим доступу до ресурсу: <https://www.elastic.co/guide/en/elasticsearch/hadoop/current/spark.html>
 21. Apache Spark Documentation. [Електронний ресурс] – Режим доступу до ресурсу: <https://spark.apache.org/docs/2.2.1/>
 22. HDFS User Guide [Електронний ресурс] – Режим доступу до ресурсу: <https://hadoop.apache.org/docs/r2.4.1/hadoop-project-dist/hadoopdfs/HdfsUserGuide.html>
 23. Hadoop tutorial [Електронний ресурс] – Режим доступу до ресурсу: <https://www.tutorialspoint.com/hadoop/>
 24. Mahout documentation [Електронний ресурс] – Режим доступу до ресурсу: <https://mahout.apache.org/>
 25. Building a Correlated Cross-Occurrence (CCO) Recommenders with the Mahout CLI [Електронний ресурс] – Режим доступу до ресурсу: <http://mahout.apache.org/users/recommender/intro-cooccurrence-spark.html>
 26. Amazon [Електронний ресурс] – Режим доступу до ресурсу: <https://www.amazon.com/>
 27. Rozetka.ua [Електронний ресурс] – Режим доступу до ресурсу: <https://rozetka.com.ua/>
 28. An introduction to HDFS [Електронний ресурс] – Режим доступу до ресурсу: <https://dzone.com/articles/an-introduction-to-hdfs>
 29. Семантичний аналіз і способи представлення сенсу тексту у комп'ютерній лінгвістиці. URL: <https://cyberleninka.ru/article/n/semanticheskij-analiz-i-sposoby-predstavleniyasmysla-teksta-v-kompyuternoy-lingvistike/viewer>
 30. Ключові слова та їх функції у науковому тексті. URL: <https://cyberleninka.ru/article/n/klyuchevye-slova-i-ih-funktsii-v-nauchnomtekste/viewer>
 31. Текст як конститутивна одиниця дискурсу. URL: [http://xn-e1aajfpcds8ay4h.com.ua/files/82_05\(1\).pdf#page=6](http://xn-e1aajfpcds8ay4h.com.ua/files/82_05(1).pdf#page=6)
 32. Ключові слова: основні аспекти визначення (на матеріалі англійської мови). URL: <http://olj.onua.edu.ua/index.php/olj/article/view/668/355>

33. Семантичний аналіз тексту URL:
<https://smychnyk.name/semantychnyy-analiz-tekstu.html>
34. Аркадій Анатолійович Зеленський. Актуальність дослідження програм семантичного аналізу текстів та огляд методів їх реалізації. С. 63-67. URL:
<http://ceur-ws.org/Vol-2292/paper05.pdf>
35. Windows Presentation Foundation URL:
https://uk.wikipedia.org/wiki/Windows_Presentation_Foundation
36. Розробка веб-застосунків URL: <https://webstudio2u.net/ua/sitedevelop/641-razrabotka-veb-prilozheniy.html>
37. 2017 Retrospective: A Monumental Year for the App Economy 2017 Retrospective: App Annie Content — 2017.
38. Microsoft Visual Studio Professional URL:
<https://itpro.ua/product/visual-studio-professional/?tab=description>

Додаток А

```
import numpy as np # лінійна алгебра
import pandas as pd # обробка даних, CSV I/O (e.g. pd.read_csv)
import seaborn as sns
import matplotlib.pyplot as plt

# Дані знаходяться у директорії "../input/" .

import os
print(os.listdir("../input"))
```

```
['item_properties_part1.csv', 'events.csv', 'item_properties_part2.csv', 'category_tree.csv']
```

```
items1 = pd.read_csv('../input/item_properties_part1.csv')
items2 = pd.read_csv('../input/item_properties_part2.csv')
items = pd.concat([items1, items2])
items.head(10)
```

	timestamp	itemid	property	value
0	1435460400000	460429	categoryid	1338
1	1441508400000	206783	888	1116713 960601 n277.200
2	1439089200000	395014	400	n552.000 639502 n720.000 424566
3	1431226800000	59481	790	n15360.000
4	1431831600000	156781	917	828513
5	1436065200000	285026	available	0
6	1434250800000	89534	213	1121373
7	1431831600000	264312	6	319724
8	1433646000000	229370	202	1330310
9	1434250800000	98113	451	1141052 n48.000

Розмір даних

```
items.shape
```

```
(20275902, 4)
```

+ Code

+ Markdown

```
import datetime
times = []
for i in items['timestamp']:
    times.append(datetime.datetime.fromtimestamp(i//1000.0))
```

```
items['timestamp'] = times
```

```
items.head(10)
```

i1:

	timestamp	itemid	property	value
0	2015-06-28 03:00:00	460429	categoryid	1338
1	2015-09-06 03:00:00	206783	888	1116713 960601 n277.200
2	2015-08-09 03:00:00	395014	400	n552.000 639502 n720.000 424566
3	2015-05-10 03:00:00	59481	790	n15360.000
4	2015-05-17 03:00:00	156781	917	828513
5	2015-07-05 03:00:00	285026	available	0
6	2015-06-14 03:00:00	89534	213	1121373
7	2015-05-17 03:00:00	264312	6	319724
8	2015-06-07 03:00:00	229370	202	1330310
9	2015-06-14 03:00:00	98113	451	1141052 n48.000

```
# Завантажуємо dataset дій користувача
events = pd.read_csv('../input/events.csv')
```

[+ Code](#)
[+ Markdown](#)

```
events.head(10)
```

i1:

	timestamp	visitorid	event	itemid	transactionid
0	1433221332117	257597	view	355908	NaN
1	1433224214164	992329	view	248676	NaN
2	1433221999827	111016	view	318965	NaN
3	1433221955914	483717	view	253185	NaN
4	1433221337106	951259	view	367447	NaN
5	1433224086234	972639	view	22556	NaN
6	1433221923240	810725	view	443030	NaN
7	1433223291897	794181	view	439202	NaN
8	1433220899221	824915	view	428805	NaN
9	1433221204592	339335	view	82389	NaN

```
events.shape
```

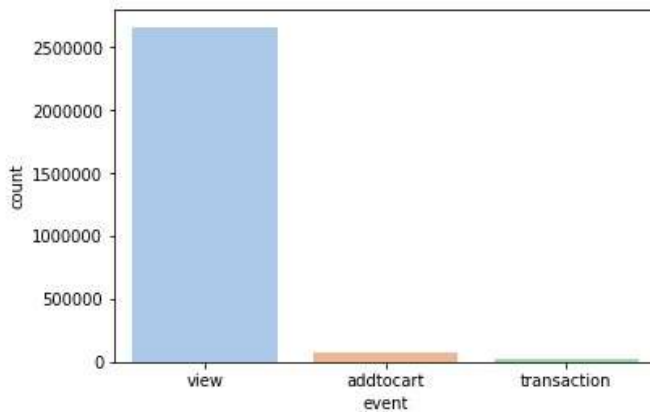
```
1): (2756101, 5)
```

Проаналізуємо види дій користувача

```
1: print(events['event'].value_counts())
   sns.countplot(x='event', data=events, palette="pastel")
```

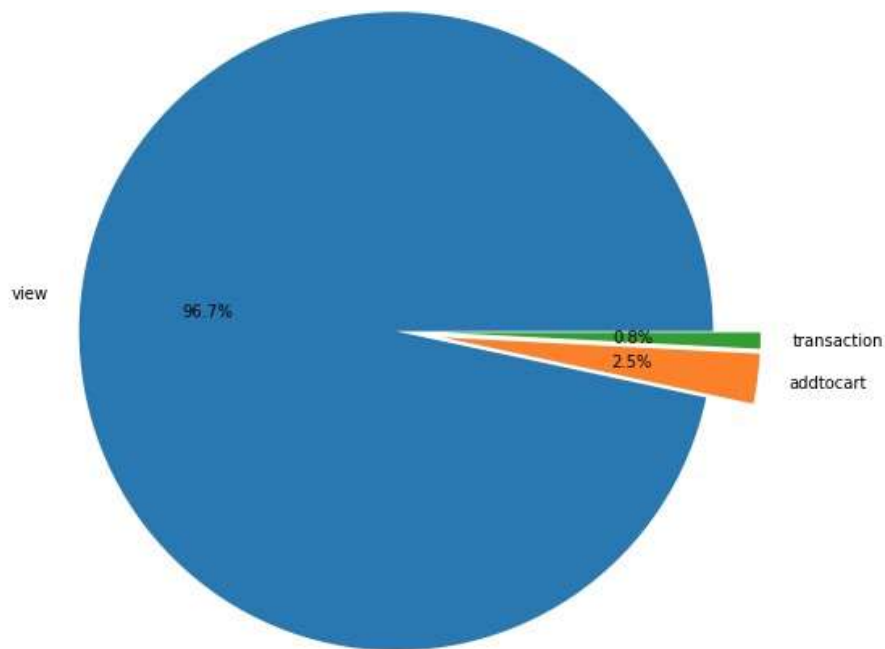
```
view          2664312
addtocart      69332
transaction    22457
Name: event, dtype: int64
```

```
0] <matplotlib.axes._subplots.AxesSubplot at 0x7fdedac95e48>
```



```
1: data = events.event.value_counts()
   labels = data.index
   sizes = data.values
   explode = (0, 0.15, 0.15) # explode 1st slice
   plt.subplots(figsize=(8,8))
   # Plot
   plt.pie(sizes, explode=explode, labels=labels, autopct='%1.1f%%', shadow=False, startangle=0)

   plt.axis('equal')
   plt.show()
```


[+ Code](#)
[+ Markdown](#)

Дерево категорій

```
: category_tree = pd.read_csv('../input/category_tree.csv')
```

```
: category_tree.head(10)
```

```
3]:
```

	categoryid	parentid
0	1016	213.0
1	809	169.0
2	570	9.0
3	1691	885.0
4	536	1691.0
5	231	NaN
6	542	378.0
7	1146	542.0
8	1140	542.0
9	1479	1537.0

Category IDs показує зв'язок різних продуктів між собою, наприклад Category ID 1016 дочірній продукт ID 213.

Кількість продуктів з ID 1016

```
: items.loc[(items.property=='categoryid')&(items.value == '1016')].sort_values('timestamp').head()
```

```
4]:
```

	timestamp	itemid	property	value
7236969	2015-05-10 03:00:00	6777	categoryid	1016
8597591	2015-05-10 03:00:00	161686	categoryid	1016
9496408	2015-05-10 03:00:00	276491	categoryid	1016
6880131	2015-05-10 03:00:00	443058	categoryid	1016
7202531	2015-05-10 03:00:00	462004	categoryid	1016

Кількість унікальних відвідувачів та сума загалом

```
:
# all unique visitors
all_customers = events['visitorid'].unique()
print("Unique visitors:", all_customers.size)

# all visitors
print('Total visitors:', events['visitorid'].size)
```

```
Unique visitors: 1407580
Total visitors: 2756101
```

+ Code

+ Markdown

Кількість користувачів, які щось придбали

```
: customer_purchased = events[events.transactionid.notnull()]['visitorid'].unique()
customer_purchased.size
```

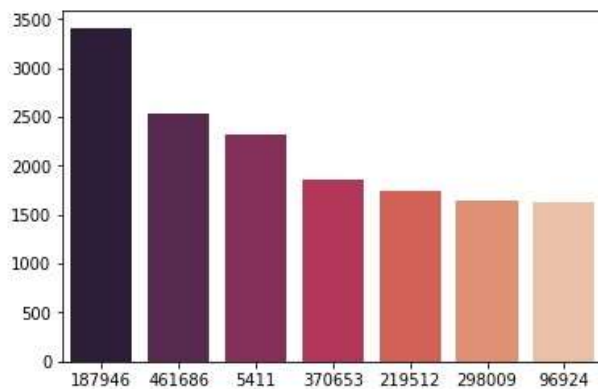
```
6] 11719
```

Існує 1407580 унікальних відвідувачів, 11719 відвідувачів зробили хоча б одну покупку

Позиції, які мають найбільше переглядів

```
import operator
grouped = events.groupby('event')['itemid'].apply(list)
views = grouped['view']
# creating dictionary for key value pair
count_view = {}
#since views is a list, we will convert it into numpy array for further manipulations
views = np.array(views[:])
#counting uniques values of views of this numpy views array
unique, counts = np.unique(views, return_counts=True)
# converting unique and counts as a dictionary with key as unique and value as counts
count_view = dict(zip(unique, counts))
#sorting the dictionary
sort_count_view = sorted(count_view.items(), key = operator.itemgetter(1), reverse = True)
# keeping number of unique views on X-axis
x = [i[0] for i in sort_count_view[:7]]
# keeping count number of views on Y-axis
y = [i[1] for i in sort_count_view[:7]]
sns.barplot(x, y, order=x, palette="rocket")
```

8] <matplotlib.axes._subplots.AxesSubplot at 0x7fdf15c460b8>



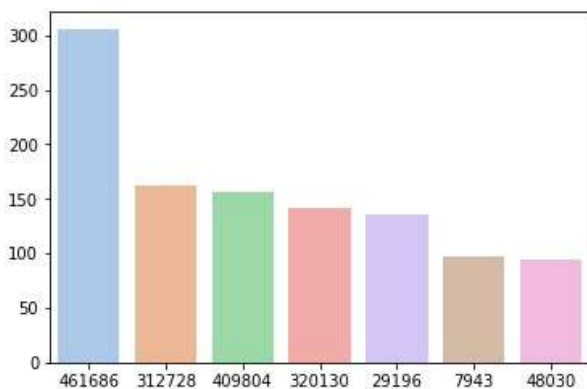
Позиції, які були найчастіше додані до кошика

```

addtocart = grouped['addtocart']
# creating dictionary for key value pair
count_addtocart = {}
#since addtocart is a list, we will convert it into numpy array for further manipulations
addtocart = np.array(addtocart[:])
#counting unique values of addtocart items of this numpy addtocart array
unique, counts = np.unique(addtocart, return_counts=True)
# converting unique and counts as a dictionary with key as unique and value as counts
count_addtocart = dict(zip(unique, counts))
#sorting the dictionary
sort_count_addtocart = sorted(count_addtocart.items(), key = operator.itemgetter(1), reverse = True)
# keeping number of unique views on X-axis
x = [i[0] for i in sort_count_addtocart[:7]]
# keeping count number of views on Y-axis
y = [i[1] for i in sort_count_addtocart[:7]]
sns.barplot(x, y, order=x, palette="pastel")

```

9] <matplotlib.axes._subplots.AxesSubplot at 0x7fdb733dcf8>



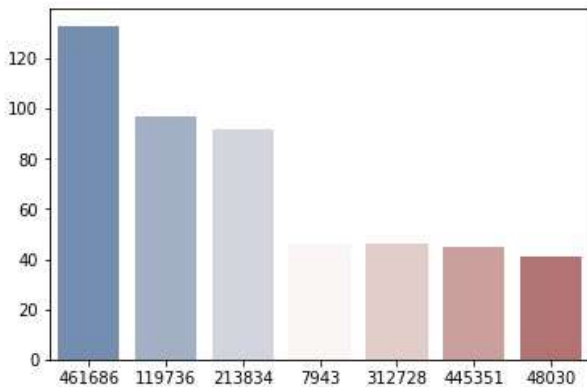
+ Code

+ Markdown

Позиції, які були найчастіше куплені

```
transaction = grouped['transaction']
# creating dictionary for key value pair
count_transaction = {}
#since addtocart is a list, we will convert it into numpy array for further manipulations
transaction = np.array(transaction[:])
#counting unique values of addtocart items of this numpy addtocart array
unique, counts = np.unique(transaction, return_counts=True)
# converting unique and counts as a dictionary with key as unique and value as counts
count_transaction = dict(zip(unique, counts))
#sorting the dictionary
sort_count_transaction = sorted(count_transaction.items(), key = operator.itemgetter(1), reverse = True)
# keeping number of unique views on X-axis
x = [i[0] for i in sort_count_transaction[:7]]
# keeping count number of views on Y-axis
y = [i[1] for i in sort_count_transaction[:7]]
sns.barplot(x, y, order=x, palette="vlag")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fdf1572b588>
```



+ Code

+ Markdown

Як один із пунктів гібридної рекомендації, можемо запропонувати користувачу список товарів, які інші користувачі придбали разом з продуктом, який він наразі дивиться

```

: # список користувачів, що зробили покупку
customer_purchased = events[events.transactionid.notnull()].visitorid.unique()

#список покупок усіх користувачів
purchased_items = []

for customer in customer_purchased:
    purchased_items.append(list(events.loc[(events.visitorid == customer) & (events.transactionid.notnull())].transactionid.unique()))

```

[+ Code](#)
[+ Markdown](#)

```

: purchased_items[:7]

```

```

:2] [[356475],
    [15335,
     380775,
     237753,
     317178,
     12836,
     400969,
     105792,
     25353,
     200793,
     80582,
     302422],
    [81345],
    [150318, 49521],
    [310791, 299044],
    [54058,
     284871,

```

Визначимо функцію, що покаже продукти, які були придбані разом один и тим самим покупцем

```
:
def recommend_items(item_id, purchased_items):
    recommendation_list = []
    for x in purchased_items:
        if item_id in x:
            recommendation_list += x

    # remove the pass item from the list and merge the above created list
    recommendation_list = list(set(recommendation_list) - set([item_id]))
    return recommendation_list
```

Зробимо рекомендацію базуючись на даних попереднього покупця, коли користувач переглядає позицію з item_id =

```
:
recommend_items(200793, purchased_items)
```

```
[4] [105792, 12836, 80582, 380775, 15335, 400969, 25353, 302422, 237753, 317178]
```

```
events.head(5)
```

+ Code

+ Markdown

