КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Економічний факультет
Кафедра економічної кібернетики

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

«Прогнозування ймовірності проведення оплати користувачів мобільного додатку»

студента 4 курсу
спеціальності 051 «Економіка»
ОПП «Економічна кібернетика»
денної форми навчання
Чумака Богдана Володимировича

Науковий керівник:
Кандидат фізико-математичних наук, доцент
Кравець Тетяна Вікторівна

Засвідчую, що у цій дипломній роботі немає запозичень із праць інших авторів без відповідних посилань
Студент_____
(підпис)

Роботу допущено до захисту перед ЕК
рішенням кафедри економічної кібернетики
від 12 червня 2023 р., протокол № 17
Завідувач кафедри:
доктор економічних наук, професор
Ляшенко Олена Ігорівна

_____
(підпис)

КИЇВ – 2023

РЕФЕРАТ

Кваліфікаційна робота бакалавра містить: 60 ст., 6 рис., 8 табл., 22 джерела, додатки

Ключові слова: мобільні додатки, юніт-економіка, додатки для знайомств, CatBoost, XGBoost, логістична регресія, ймовірність оплати, сегментація користувачів.

Об’єкт дослідження: мобільний додаток “Taimi” та його користувачі

Мета дослідження: визначити найефективнішу модель прогнозування ймовірності проведення оплати за допомогою методів машинного навчання серед таких, як XGBoost, логістична регресія, CatBoost.

Методи дослідження: моделі машинного навчання XGBoost, CatBoost. Статистична модель – логістична регресія.

Наукова новизна, теоретична значимість дослідження: дістало подальший розвиток застосування моделей машинного навчання. Випробувана нова сфера їх застосування – мобільні додатки.

Практична цінність: створена модель прогнозування ймовірності оплати для додатку “Taimi”. Її впровадження у продукт дозволить значно покращити користувацький досвід та монетизаційні метрики додатку.

RESUME

Taras Shevchenko National University of Kyiv,
Faculty of Economics, Department of Economic Cybernetics

Key words: mobile applications, unit economics, dating applications, CatBoost, XGBoost, logistic regression, likelihood of payment, user’s segmentation.

The graduation research of student Bohdan Chumak

deals with different machine learning models, such as XGBoost, CatBoost, Logistic regression, in order to find the best one to forecast the likelihood of payment by mobile application users.

The work is interesting for creating opportunities for mobile application developers to segmentate their audience using machine learning models.

Pages 60, tables 8, bibliog. 22, append. 4

Зміст

Вступ

Актуальність теми дослідження. Сучасний світ розвивається з шаленою швидкістю, потужності обчислювальних машин подвоюються кожні кілька років. Уже зараз кожному жителю Землі безкоштовно доступні хмарні потужності таких технологічних гігантів, як Google та Microsoft. Тому з кожним роком машинне навчання та нейронні мережі набувають все більшого поширення. Вони всебічно проникають життя людей, тому потрібно навчитися використовувати їх потенціал у повній мірі.

Одна з галузей, де можна ефективно використовувати алгоритми машинного навчання – мобільні додатки. Вони мають велику кількість користувачів та збирають по кожному з них певний набір інформації, що дозволяє якісно навчати моделі для відповідних задач. Серед найпопулярніших застосувань – пошук порушників, кластеризація користувачів, прогнозування цінності, яку принесе користувач за час життя на продукті та прогнозування ймовірності здійснення певної дії.

Мобільні додатки відносно новий вид програмного забезпечення, вчені ще не встигли в повній мірі оцінити потенціал даного функціоналу для власних досліджень. Однак у останні 10 років починають з’являтися статті, у яких розглядають різноманітні аспекти їхньої роботи. Так у праці Yiting Deng , Anja Lambrecht та Yongdong Liu[1] розглядається робота різних бізнес-моделей у мобільних додатках з описом переваг та недоліків моделі Freemium. Sandeep Arora, Frenkel ter Hofstede та Vijay Mahajan порівняли вплив платної та безкоштовної версії додатку на його ріст[2]. При цьому використання моделей машинного навчання набирає популярності та починає знаходити нові застосування, які описує в своїй роботі Yunbin Deng[3]. Детальний розбір використання машинного навчання та аналітики великих даних подає Robert S.H. Istepanian[4], який застосовує ці методи у сфері mobile health. Petr Hajek, Mohammad Zoynul Abedin і Uthayasankar Sivarajah використовують модель

XGBoost для прогнозування і виявлення зловмисників при проведенні платежів через мобільні додатки.

Об'єкт дослідження: мобільний додаток "Taimi" та його користувачі.

Предмет дослідження: прогнозування ймовірності здійснення оплати користувачем мобільного додатку

Мета дослідження: визначити найефективнішу модель прогнозування ймовірності проведення оплати за допомогою методів машинного навчання серед таких, як XGBoost, логістична регресія, CatBoost.

Завдання дослідження:

1. Провести аналіз сучасної економічної наукової літератури та узагальнити теоретичні дані з питання економіки мобільних додатків.
2. Дослідити будову та особливості моделей XGBoost, логістична регресія, CatBoost.
3. Визначити показники ефективності прогнозування ймовірності проведення платежу в мобільному додатку "Taimi".

Методи дослідження: моделі машинного навчання XGBoost, CatBoost. Статистична модель – логістична регресія.

Наукова та практична новизна роботи: дістало подальший розвиток застосування моделей машинного навчання. Випробувана нова сфера їх застосування – мобільні додатки. Створена модель прогнозування ймовірності оплати для додатку "Taimi". Її впровадження у продукт дозволить значно покращити користувацький досвід та монетизаційні метрики додатку.

Розділ 1. Особливості роботи мобільних додатків. Оцінка їхньої ефективності.

1.1. Юніт-економіка та бізнес-моделі мобільних додатків

Мобільні додатки - це технологія, який дозволяє розробникам з усіх куточків світу створювати продукти, які будуть працювати на будь-якому смартфоні та виконувати різноманітні функції - від замовлення їжі до консультації з лікарем.

У 2022 році обсяг світового ринку мобільних додатків оцінювався в 206,85 млрд доларів США і, як очікується, збільшуватиметься з середньорічним темпом зростання 13,8% у період з 2023 по 2030 рік (Рис. 1).

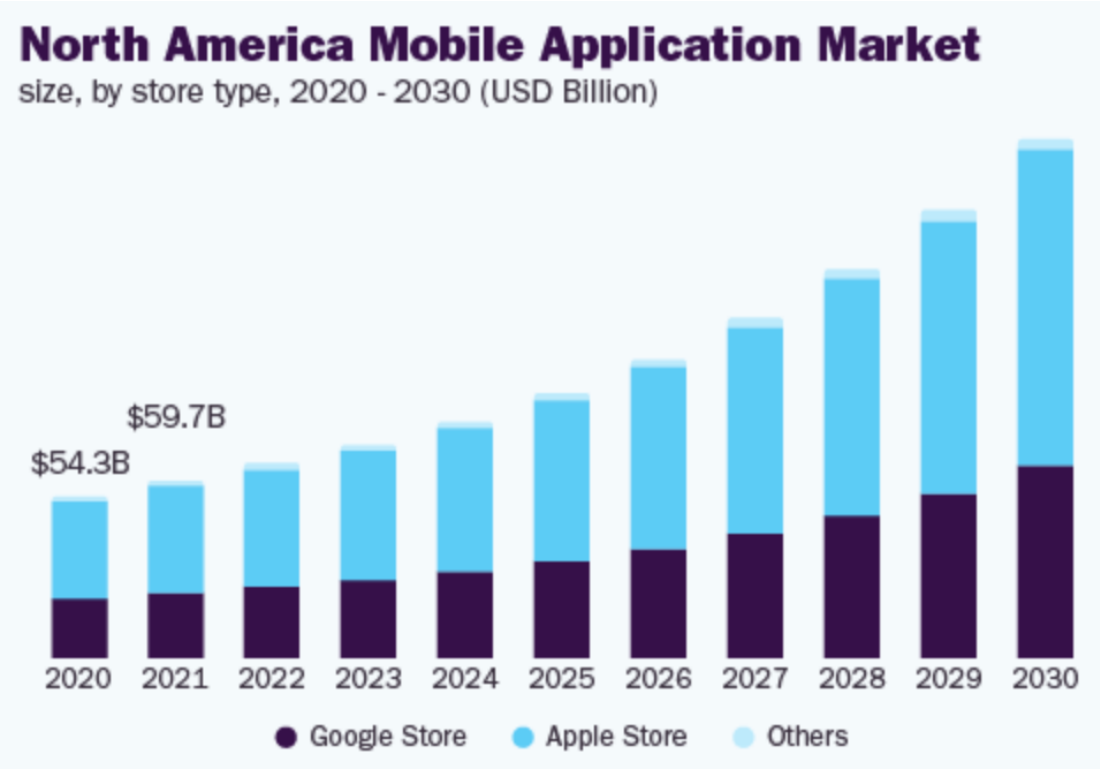

Рис. 1. Динаміка об'єму ринку мобільних додатків

Джерело: [6]

Більшість мобільних додатків - це бізнес, головна ціль якого заробити гроші. При цьому економічна ефективність такого бізнесу вимірюється специфічними метриками. Базовим підходом до визначення ефективності роботи мобільного додатку є юніт-економіка.

Юніт-економіка - це аналіз та оцінка фінансових показників бізнесу в розрізі кожної одиниці продукції, послуги або клієнта. Вона зосереджена на розумінні прибутковості та життєздатності таких окремих одиниць і допомагає визначити економічну цінність, що генерується кожною такою одиницею.

У контексті мобільного додатку юніт-економіка може включати аналіз фінансових показників, пов'язаних з кожним користувачем або клієнтом. Таких як вартість придбання, отриманий дохід і пов'язані з ним витрати на надання послуги. Вивчаючи економіку на рівні користувача, компанії можуть оцінити фінансову стійкість своєї діяльності, визначити сфери для вдосконалення та прийняти обґрунтовані рішення щодо ціноутворення, маркетингових стратегій та розподілу ресурсів.

Юніт-економіка використовується для наступних задач:

- *Створення стратегії ціноутворення та монетизації.* Юніт-економіка дає уявлення про цінність, яку додаток надає користувачам, і допомагає визначити відповідні стратегії ціноутворення та монетизації. Розуміючи потенціал доходу і структуру витрат, пов'язаних з кожним користувачем, з’являється можливість приймати обґрунтовані рішення щодо встановлення цін, пропонування преміум-функцій, впровадження покупок у додатку тощо.
- *Залучення та утримання клієнтів*. Юніт-економіка допомагає оцінити економічну ефективність залучення нових клієнтів для мобільного додатку. Розрахувавши вартість залучення клієнтів(CAC) і порівнявши її з життєвою цінністю клієнта(LTV), ви зможете оцінити, чи приносять ваші маркетингові кампанії позитивний дохід. Крім того, розуміння економічної ефективності утримання та відтоку користувачів дозволяє визначити стратегії для підвищення лояльності клієнтів, зниження рівня відтоку та збільшення їх життєвої цінності.
- *Розподіл ресурсів*. Аналіз юніт-економіки дозволяє ефективно розподіляти ресурси. Визначивши одиниці (користувачів або сегменти

клієнтів), які приносять найбільший дохід і мають найвищу рентабельність, ви можете визначити пріоритети у маркетингових зусиллях, розробці продуктів та ініціативах з підтримки клієнтів. Це гарантує, що ви розподілите свій час і бюджет на ті сфери, які мають найбільший вплив на прибуток.

За роки існування мобільних додатків з'явився цілий ряд бізнес-моделей. Популярність цих моделей може варіюватися залежно від конкретного додатку, цільової аудиторії та динаміки ринку. Ось деякі з найпопулярніших бізнес-моделей в індустрії мобільних додатків:

- *Безкоштовна з рекламою*. У цій моделі додаток пропонується користувачам безкоштовно, але в ньому відображається реклама. Рекламодавці платять власнику додатку за покази реклами або кліки, забезпечуючи дохід. Ця модель дозволяє користувачам отримати доступ до основних функцій додатку без будь-яких попередніх платежів. Приклад: "TikTok" - це додаток соціальна мережа, який пропонує користувачам безкоштовну платформу для створення та обміну короткими відео. Він генерує дохід за рахунок реклами в додатку, показуючи різні типи оголошень між відеоконтентом.
- *Freemium*. Це поєднання слів "free" і "premium". Додаток пропонується безкоштовно з обмеженою функціональністю або можливостями, а користувачі мають можливість перейти на преміум-версію, сплативши певну суму. Преміум-версія зазвичай розблоковує додаткові функції, видаляє рекламу або пропонує покращений користувацький досвід. Моделі Freemium мають на меті перетворити безкоштовних користувачів на платних клієнтів, демонструючи цінність преміум-пропозиції[7].

  Приклад: "Spotify" - це додаток для потокової передачі музики, який пропонує безкоштовну версію з обмеженими можливостями, включаючи періодичну рекламу та відтворення у випадковому порядку. Користувачі можуть перейти на преміум-версію "Spotify Premium", яка

забезпечує прослуховування без реклами, відтворення в режимі офлайн та інші преміум-функції.

- *Покупки в додатку*. Ця модель передбачає надання безкоштовного додатку, але з додатковим контентом, віртуальними товарами або преміум-функціями в додатку, які користувачі можуть придбати. Це дозволяє користувачам взаємодіяти з додатком і здійснювати додаткові покупки, щоб покращити свій досвід або прогрес у додатку. Поширеними прикладами є розблокування рівнів у грі, купівля віртуальної валюти або доступ до ексклюзивного контенту[8]. Приклад: "Candy Crush Saga" - це популярна мобільна гра, яка пропонує покупки в додатку бустерів, додаткових життів і додаткових рівнів для покращення ігрового досвіду. Гравці можуть робити додаткові покупки, щоб долати складні рівні чи пришвидшувати свій прогрес у грі.
- *Підписки*. Підписка передбачає стягнення з користувачів регулярної плати (щомісячної, щоквартальної або щорічної) за доступ до контенту або послуг додатку. Ця модель зазвичай використовується для додатків, які надають постійну цінність або доступ до преміум-контенту, таких як потокові медіа-платформи, фітнес-програми з персоналізованими планами тренувань або інструменти для підвищення продуктивності з розширеними функціями. Передплата забезпечує стабільний потік доходів і сприяє формуванню довгострокових відносин з клієнтами. Приклад: "Netflix" - це додаток для потокового відео, який вимагає підписки для доступу до великої бібліотеки фільмів і телешоу. Користувачі платять щомісячну плату, щоб насолоджуватися переглядом відео без реклами.
- *Оплата за завантаження*. У цій моделі користувачі платять одноразову плату за завантаження та доступ до додатку. Зазвичай додаток є повністю функціональним без будь-яких додаткових покупок або підписок. Оплата за завантаження є простою моделлю отримання доходу і часто використовується для додатків з певною нішею або

високою цінністю, таких як професійні додатки, спеціалізовані утиліти або освітній контент преміум-класу.
Приклад: "Minecraft" - це відеогра-пісочниця, яку користувачі купують в магазинах додатків за одноразову плату. Після покупки користувачі отримують повний доступ до функцій гри і можуть грати без будь-яких додаткових платежів.

- *Спонсорство або партнерство*. Деякі мобільні додатки співпрацюють з брендами, компаніями або спонсорами, щоб монетизувати свою базу користувачів. Це може включати відображення спонсорського контенту, надання брендованого досвіду або пропонування ексклюзивних знижок чи винагород користувачам додатків. Моделі спонсорства або партнерства вимагають налагодження відносин з відповідними партнерами та узгодження їхніх пропозицій з цільовою аудиторією додатку.
  Приклад: "Nike Training Club" - це фітнес-додаток, який пропонує плани тренувань та поради щодо харчування. Додаток співпрацює з брендом Nike, щоб пропонувати спонсорські тренування з використанням фірмового контенту.
- *Монетизація даних*. У деяких випадках додатки збирають дані користувачів (за згодою та з дотриманням правил конфіденційності) і монетизують їх, надаючи третім особам інформацію для дослідження ринку або можливості таргетованої реклами. Ця модель використовує дані користувачів для опосередкованого отримання доходу, пропонуючи при цьому безкоштовний доступ до додатку.
  Приклад: "Карти Google" - це навігаційний додаток, який збирає дані користувача, включаючи інформацію про місцезнаходження, щоб надавати персоналізовані рекомендації та оновлювати інформацію про трафік. Google використовує ці дані для покращення можливостей таргетування реклами, опосередковано монетизуючи додаток.

- *Партнерський маркетинг*(affiliate marketing). Партнерський маркетинг передбачає просування продуктів або послуг у додатку та отримання комісії за кожного успішного реферала або конверсію. Власники додатків можуть співпрацювати з партнерськими мережами або конкретними брендами, щоб просувати відповідні пропозиції для своєї користувацької бази. Партнерський маркетинг може бути реалізований за допомогою посилань, банерів або спеціальних реферальних кодів.
- Ліцензування або біле маркування(white labeling). Деякі мобільні додатки генерують дохід, ліцензуючи свою технологію. Це дозволяє іншим компаніям використовувати функції або інфраструктуру цього додатку у своїх власних продуктах.

Важливо зазначити, що різні моделі монетизації можуть працювати краще для певних категорій додатків або демографічних груп користувачів. При виборі відповідної моделі монетизації враховуються такі фактори, як призначення додатку, цільова аудиторія, конкуренція та користувацький досвід. Розробники та компанії часто експериментують з різними бізнес-моделями. Вони можуть змінювати модель навіть протягом короткого часу щоб оптимізувати отримання прибутку, зберігаючи при цьому позитивний користувацький досвід.

1.2. Основні метрики мобільних додатків.

Якісне управління мобільним додатком потребує постійного моніторингу усіх аспектів продукту. Для цього потрібно знайти точки, які визначають ефективність роботи додатку та визначити, які метрики дозволять розуміти поточний стан кожної з цих точок. В залежності від специфіки продукту, його бізнес-моделі та етапу розвитку метрики можуть різнитися. При цьому є набір метрик, за якими слідкують практично всі мобільні додатки:

- *Метрики придбання(Acquisition Metrics):*
  - Кількість завантажень додатку.

- Кількість разів, коли додаток з'являється в результатах пошуку або в основних розділах магазину додатків.
- Коефіцієнт конверсії в App Store. Відсоток відвідувачів магазину, які завантажили додаток.

Показники придбання дозволяють оцінити успішність маркетингових кампаній і стратегій. Відстежуючи такі показники, як кількість завантажень і коефіцієнт конверсії в App Store, можна зрозуміти, наскільки добре додаток резонує з цільовою аудиторією і чи сприяють маркетингові зусилля залученню користувачів.

Показники придбання допомагають оцінити ефективність оптимізації App Store. Моніторинг таких показників, як кількість показів у App Store, коефіцієнт кліків (CTR) і коефіцієнт конверсії в App Store, дає уявлення про видимість і привабливість додатку.

Показники придбання дозволяють порівняти ефективність різних каналів залучення користувачів. Відстежуючи показники залучення для кожного каналу (наприклад, органічна, платна реклама, реферали або соціальні мережі), можна визначити, які канали приносять найбільше завантажень, конверсій і високоякісних користувачів. Ці знання допоможуть ефективно розподіляти маркетинговий бюджет і ресурси.

- *Метрики залученості (Engagement Metrics):*
  - Кількість користувачів, які активно взаємодіяли з додатком протягом певного періоду часу.
  - Тривалість сесії. Середня тривалість часу, який користувачі проводять в межах однієї сесії.

Показники залученості допомагають оцінити утримання користувачів, що є ключовим показником успіху додатку. Відстежуючи такі показники, як активні користувачі, тривалість сесії та час між сесіями, можна зрозуміти, як часто користувачі повертаються до додатку і наскільки вони зацікавлені в ньому з часом. Вищі показники залученості та утримання вказують на те, що

користувачі знаходять цінність у додатку і з більшою ймовірністю продовжуватимуть ним користуватися.

Крім того метрики залученості дають уявлення про задоволеність користувачів і про те, наскільки добре додаток відповідає їхнім потребам. Ви можете визначити області додатку, на які користувачі витрачають найбільше часу, а також функції чи контент, які вони вважають найбільш цінними. Ці знання допоможуть визначити пріоритети вдосконалень і поліпшень, які відповідають вподобанням і очікуванням користувачів. Розуміючи, як користувачі взаємодіють з додатком, можна виявити потенційні сфери для впровадження вбудованих покупок, реклами або преміум-функцій, які відповідають поведінці та вподобанням користувачів. Це може допомогти оптимізувати стратегії монетизації та збільшити дохід[10].

- *Метрики утримання(Retention Metrics):*
  - Коефіцієнт утримання. Відсоток користувачів, які продовжують користуватися додатком протягом певного періоду.
  - Коефіцієнт відтоку. Відсоток користувачів, які припиняють користуватися додатком або скасовують підписку протягом певного періоду.
  - Час між сесіями. Середній часовий проміжок між сеансами користувача, що вказує на частоту та залученість користувачів.

Показники утримання користувачів, такі як коефіцієнт утримання, вимірюють відсоток користувачів, які продовжують користуватися додатком протягом тривалого часу. Вищий показник утримання вказує на те, що користувачі лояльні до додатку, знаходять у ньому цінність і з більшою ймовірністю стануть довгостроковими користувачами. Зосередженість на утриманні користувачів допомагає створити базу лояльних користувачів, яка може сприяти сталому зростанню та збільшенню життєвої цінності(LTV) кожного користувача[9].

Також показники утримання можуть слугувати індикаторами задоволеності користувачів. Якщо користувачі продовжують взаємодіяти з

додатком протягом тривалого періоду, це свідчить про те, що вони задоволені досвідом і знаходять цінність у використанні додатку. І навпаки, високий показник відтоку користувачів може свідчити про проблеми з користувацьким досвідом, продуктивністю роботи додатку або контентом, які необхідно вирішити.

Показники утримання тісно пов'язані з генеруванням доходу. Користувачі, які постійно повертаються, з більшою ймовірністю здійснюють покупки в додатку, підписуються на преміум-функції або контент і генерують постійний дохід. Зосереджуючись на утриманні користувачів, збільшуються шанси перетворити їх на платоспроможних клієнтів і максимізувати потенціал монетизації додатку. До того ж, утримання існуючих користувачів часто є більш економічно ефективним, ніж залучення нових. Залучення користувачів може бути дорогим і вимагати маркетингових зусиль, рекламних кампаній та інших каналів залучення. Покращуючи показники утримання користувачів, можна зменшити потребу в агресивних стратегіях залучення користувачів і ефективніше розподіляти ресурси.

- *Метрики монетизації:*
  - Коефіцієнт конверсії у покупку. Відсоток користувачів, які виконують бажану дію, наприклад, здійснюють покупку або підписуються на послугу.
  - Середній дохід на користувача (ARPU): Середній дохід, що генерується на одного користувача протягом певного періоду часу.
  - Життєва цінність користувача (LTV). Загальна цінність, яку генерує користувач протягом усього свого життєвого циклу як користувач додатку. Вимірюється у грошах.
  - Вартість встановлення (CPI). Маркетингові витрати на одне встановлення додатку.
  - Рентабельність інвестицій (ROI): Фінансовий прибуток, отриманий від маркетингу та залучення користувачів. Вираховується, як відношення LTV до CPI.

Метрики монетизації допомагають зрозуміти, наскільки ефективно додаток генерує дохід. Відстежуючи такі показники, як середній дохід на користувача (ARPU), коефіцієнт конверсії в покупку та загальний дохід, можна оцінити фінансовий стан додатку. Ця інформація має вирішальне значення для підтримки та розвитку бізнесу, покриття витрат на розробку та обслуговування.

Показники монетизації дозволяють оцінити ефективність різних стратегій монетизації у додатку. Відстежуючи показники, пов'язані з покупками в додатку, підписками, доходами від реклами або іншими джерелами доходів, ви можете оцінити, які стратегії приносять найбільший дохід, і визначити області для оптимізації або експериментів.

Показники монетизації можуть впливати на користувацький досвід та утримання користувачів. Дуже важливо знайти баланс між монетизацією та задоволеністю користувачів, щоб уникнути негативного впливу на рівень залучення та утримання користувачів. Відстежуючи показники монетизації разом з показниками залучення та утримання користувачів, можна виявити взаємозв'язок між ними та оптимізувати стратегії монетизації без шкоди для користувацького досвіду.

- *Метрики користувацького досвіду(UX Metrics):*
  - Кількість збоїв або проблем зі стабільністю роботи додатку, з якими стикаються користувачі.
  - Час, необхідний для того, щоб додаток завантажився і став придатним для використання.
  - Оцінки та відгуки користувачів у магазинах додатків(App Stores), що вказують на їхню задоволеність та зворотній зв'язок.

Метрики користувацького досвіду допомагають оцінити рівень задоволеності користувачів додатком. Такі показники, як користувацькі рейтинги, огляди та відгуки, дають цінну інформацію про те, як користувачі сприймають застосунок.

Метрики користувацького досвіду також допомагають оцінити продуктивність додатку. Такі показники, як кількість збоїв, час завантаження та швидкість відгуку, надають інформацію про стабільність та надійність додатку. Якщо користувачі мають позитивний і приємний досвід роботи з додатком, вони з більшою ймовірністю продовжать користуватися ним з часом. Зосередившись на покращенні користувацького досвіду, можна підвищити рівень утримання користувачів.

- *Соціальні метрики:*
  - Кількість разів, коли користувачі ділилися контентом з додатку в соціальних мережах.
  - Коефіцієнт рефералів. Відсоток нових користувачів, залучених за допомогою рефералів від існуючих користувачів.

Соціальні показники допомагають виміряти охоплення на платформах соціальних мереж. Такі показники, як кількість підписників, вподобань, поширень та згадок у соціальних мережах, вказують на рівень впізнаваності та видимості додатку в соціальних спільнотах. Відстежуючи ці показники, можна оцінити ефективність стратегій у соціальних мережах.

Крім того, такі метрики можуть запропонувати цінний аналіз відгуків та настроїв користувачів. Відстежуючи згадки, коментарі та прямі повідомлення в соціальних мережах, ви можете отримати уявлення про думки користувачів, рівень їхньої задоволеності та потенційні проблеми чи занепокоєння[12].

Постійне спостереження за соціальними метриками може допомогти вам у підтримці клієнтів та ініціативах з розбудови спільноти. Відстежуючи запити, відгуки та скарги користувачів у соціальних мережах, можна оперативно реагувати на них та надавати ефективну підтримку.

Відстежуючи всі ці показники, можна отримати цінну інформацію про поведінку користувачів, продуктивність додатку, отримання прибутку та задоволеність користувачів. Це дозволить приймати рішення на основі даних, оптимізувати користувацький досвід і підвищити рентабельність мобільного додатку.

1.3. Додаток для знайомств Taimi. Його будова та особливості використання.

Прогноз ймовірності проведення оплати буде будуватися на даних, які надані додатком для знайомств Taimi. Тому важливо детально розглянути основний функціонал цього продукту, а також зазначити особливості поведінки користувачів та метрики, які вважаються ключовими для оцінки ефективності роботи додатку.

Taimi - це додаток для знайомств і соціальна мережа, розроблена для ЛГБТК+ людей. Він має на меті забезпечити безпечну та інклюзивну платформу для людей, щоб вони могли спілкуватися, заводити друзів та досліджувати стосунки[11].

Шлях користувача на продукті починається зі скачування додатку та переходу до етапу реєстрації (Рис. 2):

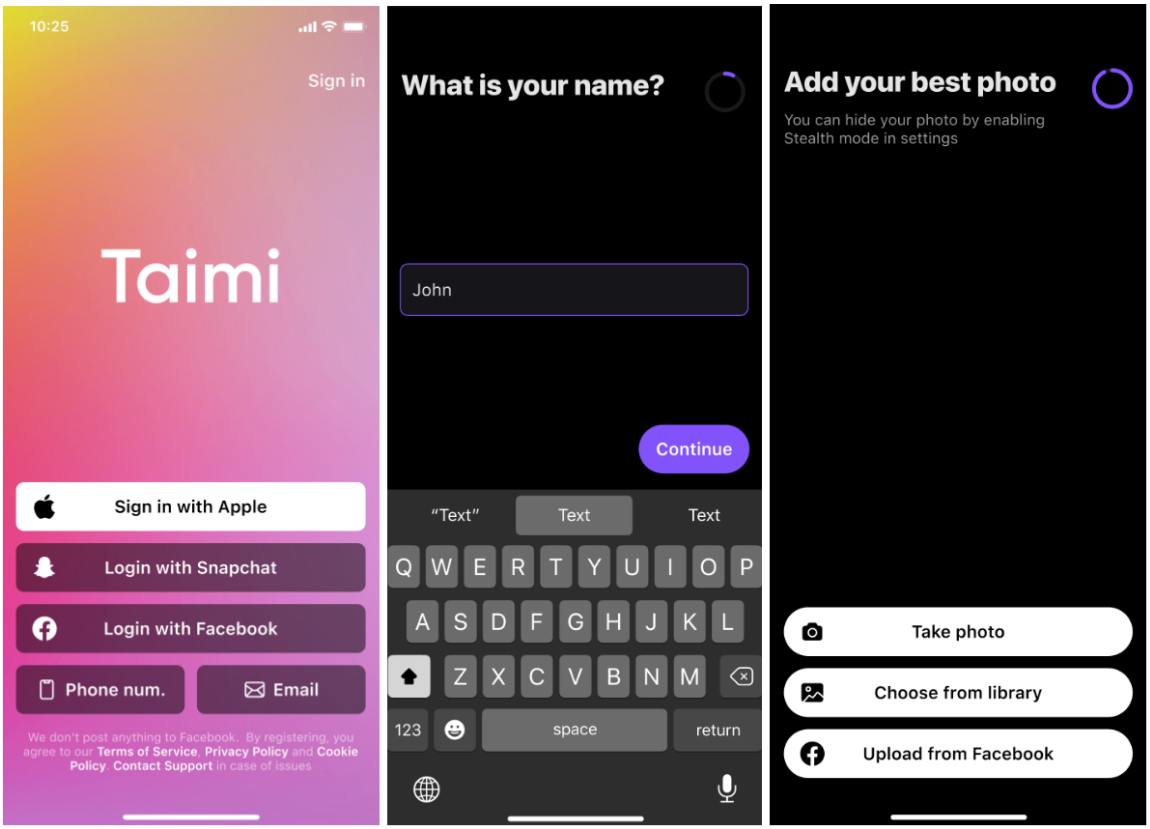

Рис. 2. Інтерфейс реєстрації

Джерело: [11]

На етапі реєстрації користувачів вказують інформацію про себе. В неї входять базові питання щодо імені, дати народження, гендеру, роботи,

інтересів. Також користувач отримує можливість додати власні фото та заповнити свій профіль. Кінцевий етап реєстрації - вибір уподобань щодо потенційного партнера. Користувачі обирають вік, гендер та дистанцію до партнера.

Наступний крок у користувацькому досвіді - перехід у розділ додатку під назвою Finder (Рис. 3). Це місце, де з'являються потенційні партнери. Користувач має змогу переглянути текстові та фото картки іншої людини і прийняти рішення щодо неї. У нього є вибір або відхилити цю людину, або ж вподобати її і натиснути на кнопку лайку. Також користувач має змогу відправити повідомлення цій людині або відправити їй "Rainbow like" - знак особливої уваги та преміум-функціонал.

Рис. 3. Інтерфейс розділу Finder

Джерело: [11]

Розділ додатку у якому користувачі мають змогу спілкуватися - Message Tab (Рис. 4). Тут зберігаються усі чати за участю даного користувача. Він може

розпочати чи продовжити діалог з тими, хто його вподобав. Важливо додати, що діалог між користувачами можливий лише за умови взаємного уподобання.

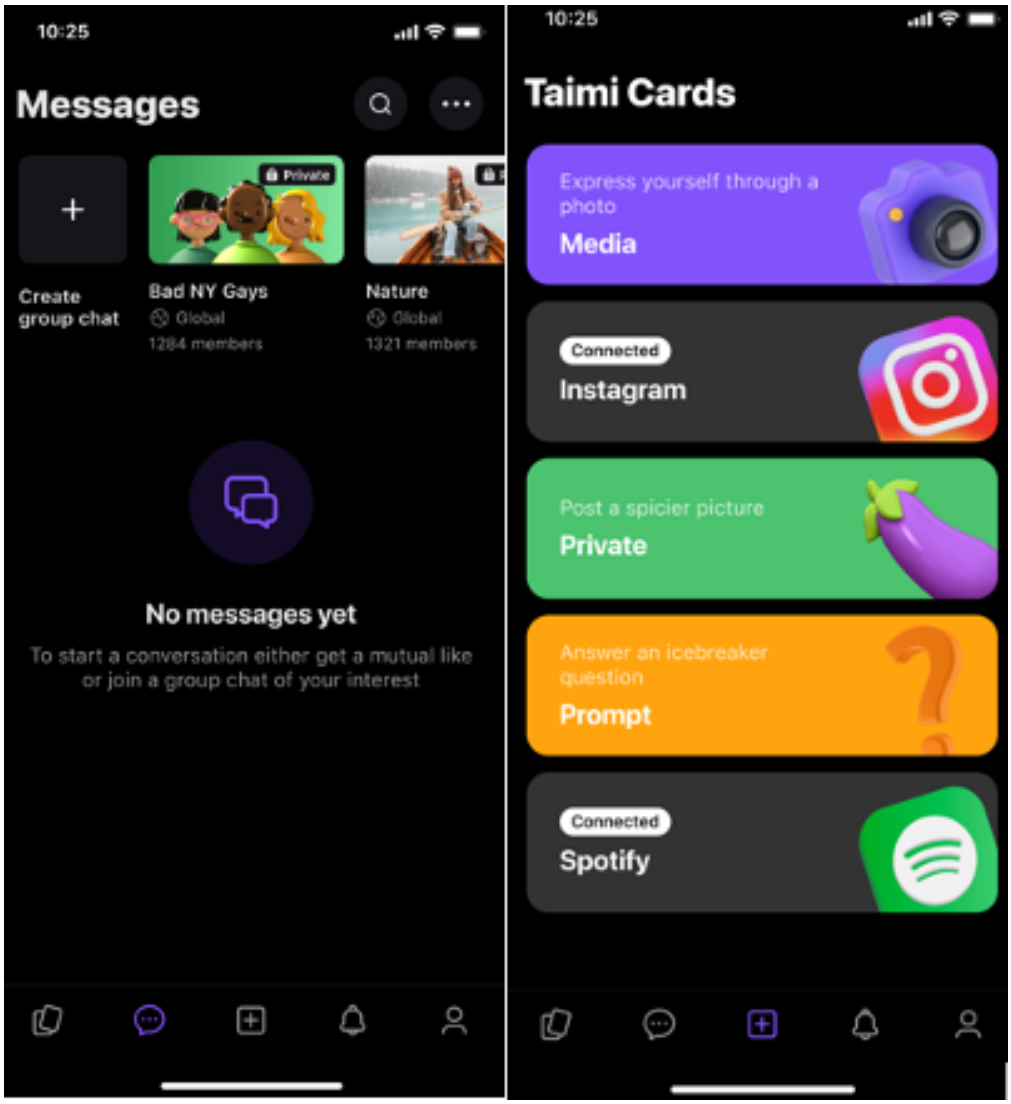

Рис. 4. Message Tab і Creator

Джерело: [11]

Розділ для створення контенту – Creator (Рис. 4). У ньому користувачі можуть поділитися власними фотографіями, додати Instagram профіль, додати текстові картки або профіль з додатку Spotify.

Один з найважливіших розділів додатку - Notification Center (Рис. 5). У ньому користувачі мають можливість переглядати всю вхідну активність. Тобто усі лайки, взаємні вподобання, потенційні діалоги та візитери їхнього профілю. Важливо відмітити те, що користувач може бачити не усіх людей, які його вподобали. Деякі з них будуть мати нечітку фотографію і без купівлі

преміум-підписки неможливо буде переглянути їхні профілі, щоб вподобати у відповідь.

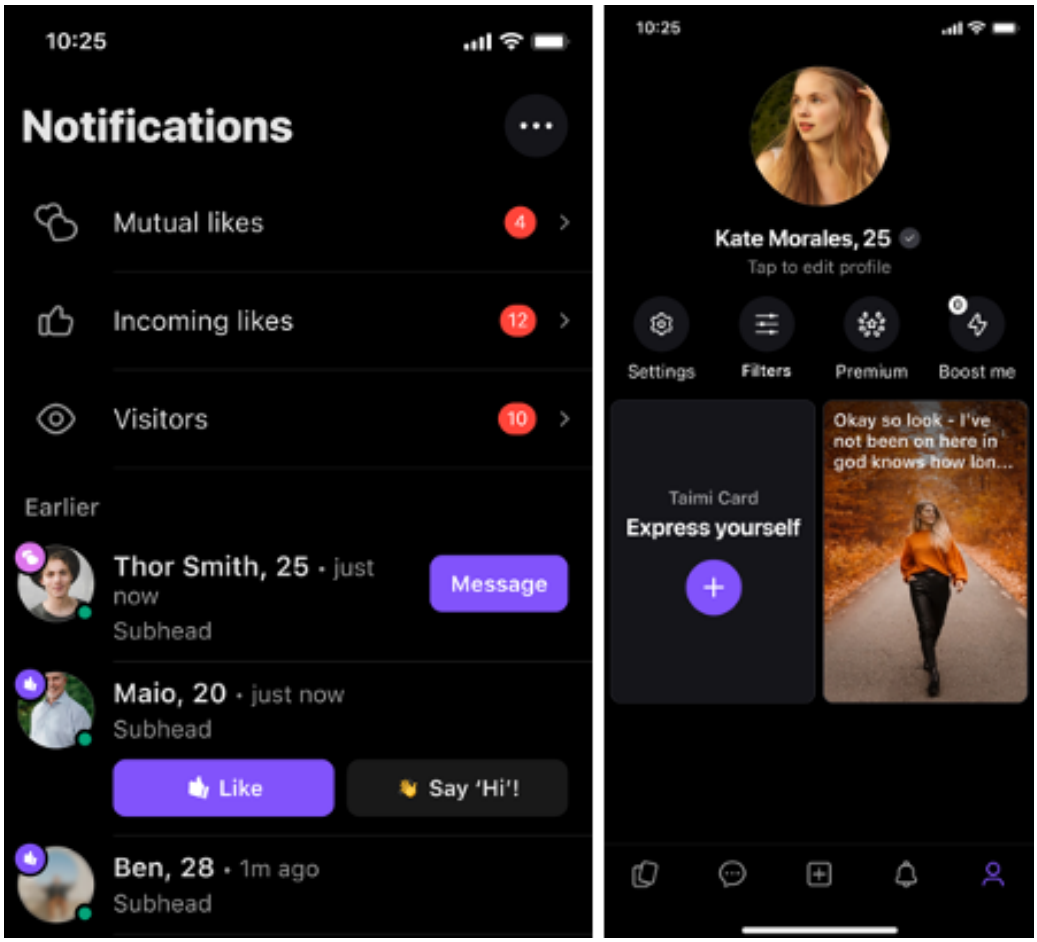

Рис. 5. Notification Center і Profile

Джерело: [11]

Останній розділ додатку – Profile (Рис. 5). Його мета полягає в тому, щоб користувач міг подивитися який вигляд має його профіль для інших людей.

Також в контексті прогнозування оплати варто розглянути монетизаційну систему даного додатку. Бізнес-модель додатку - Freemium з підпискою. Тобто основний функціонал надається безкоштовно, але є також можливості, які доступні лише для преміум-користувачів. Для їх розблокування необхідно придбати підписку.

Серед таких особливих можливостей варто виділити наступні:

- Можливість бачити людей, які тебе вподобали. У базовій версії додатку ця функція недоступна, що змушує людей переходити до розділу Finder

та оцінювати людей там. На це йде досить багато часу і є ймовірність, що людина не знайде жодного взаємного уподобання.

- Обмеження по кількості переглянутих профілів. Якщо людина переходить до розділу Finder, то кількість профілів, які вона може побачити обмежена. Щоб збільшити ліміт по кількості переглянутих профілів необхідно придбати підписку.
- Перегляд візитерів твого профілю. У базовій версії додатку недоступна можливість переглянути тих, хто зробив візит твого профілю.
- Rainbow Like. Можливість відправити особливе уподобання і виділитися серед інших.
- Boost. Функція, яка збільшує частоту показу твого профілю у розділі Finder на певний час. Тобто користувач отримує значно більше вхідної активності, такої як лайки, перегляди профілю, взаємні вподобання.
- Rollback. Функція, яка дозволяє повернутися до людини, яку ти випадково відхилив, і прийняти інше рішення.

При спробі користувача використати будь-яку з наведених функцій виникатиме екран, який пропонуватиме оформити підписку і отримати бажаний функціонал (Рис. 6).

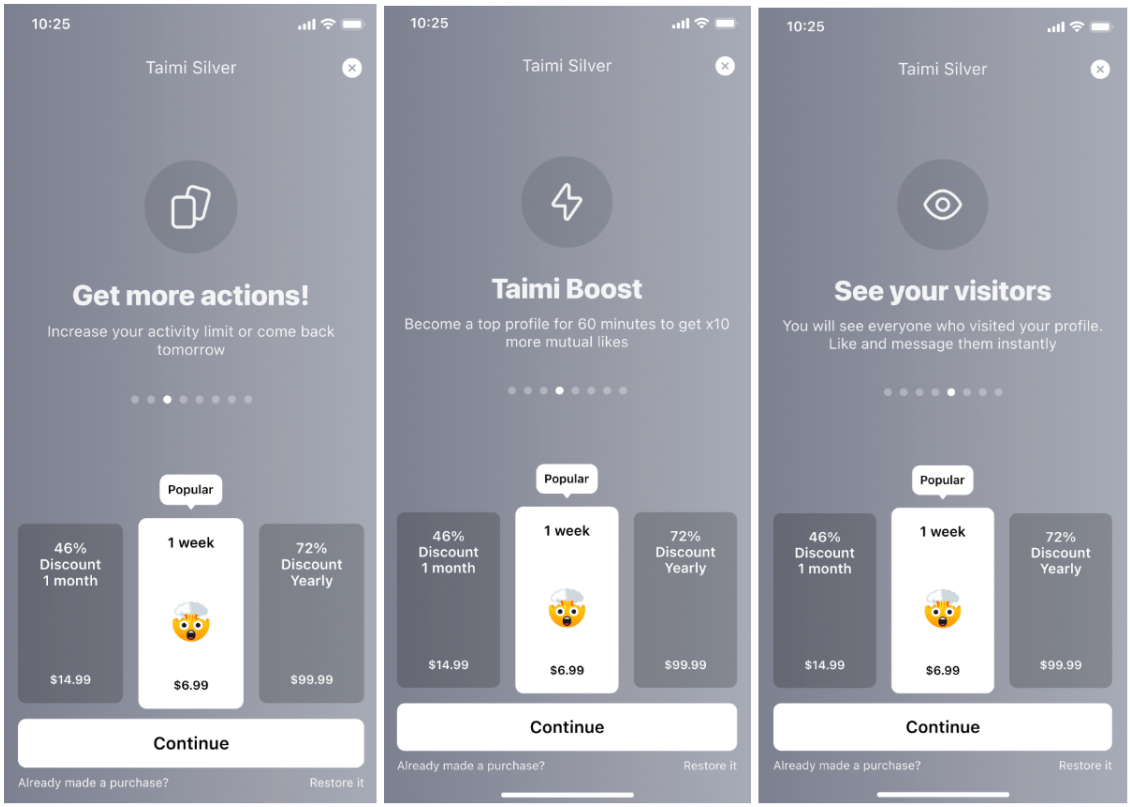

Рис. 6. Монетизаційні екрани

Джерело: [11]

Taimi - це додаток для знайомств, тому найважливішою частиною додатку є люди, які ним користуються. Саме аудиторія створює цінність користування, оскільки основна мета людей, які скачують додаток - знайти собі пару. Звідси виникає протиріччя у тому, яким чином повинен розвиватися бізнес. З одного боку основна ціль бізнесу - отримання і максимізація прибутку. Для цього логічним кроком здається максимальне обмеження дій користувача, бо в наслідок цього він з більшою ймовірністю проведе оплату. Однак, якщо занадто сильно обмежити людину, то вона просто перестане користуватися додатком, а значить інший користувач втратить потенційну пару, відповідно його цінність від користування також зменшиться.

Тому для подальшого розвитку продукту необхідно створити таку модель, яка дозволить чітко сегментувати тих, хто купить підписку в майбутньому і тих, хто не зробить цього. Якщо дана модель запрацює, то неплатники отримають більше свободи, будуть генерувати більше активності та тим самим будуть спонукати потенційних платників до здійснення покупки. Таким чином

можна буде досягнути позитивного впливу як на утримання та залученості, а й не зачепити монетизаційні метрики, які є ключовими для роботи продукту, як бізнесу.

Розділ 2. Огляд моделей прогнозування ймовірності оплати у мобільному додатку

2.1. Модель Catboost

CatBoost - це фреймворк градієнтного бустінгу, спеціально розроблений для обробки категорійних ознак і забезпечення високоефективних прогнозів. CatBoost розшифровується як "категоріальний бустінг"[14].

Ключові особливості моделі CatBoost:

Структура градієнтного бустінгу. CatBoost базується на градієнтному бустінгу, який є ансамблевим методом, що поєднує декілька слабких моделей прогнозування (як правило, дерева рішень) для створення сильної моделі прогнозування. Він працює шляхом ітеративного додавання нових моделей, які фокусуються на виправленні помилок, зроблених попередніми моделями.

Обробка категоріальних ознак. Однією з відмінних рис CatBoost є його здатність обробляти категоріальні ознаки безпосередньо, не вимагаючи попередньої обробки або “one hot encoding”. Він використовує інноваційний алгоритм під назвою ‘“Ordered Boosting” для ефективної обробки категоріальних ознак. Це особливо корисно при роботі з наборами даних, які містять суміш числових і категоріальних змінних.

Вбудована обробка пропущених значень. CatBoost має вбудовані механізми для обробки відсутніх значень у вхідних даних. Він може навчатися на відсутніх значеннях категоріальних ознак, розглядаючи їх як окрему категорію, а також обчислює відсутні значення числових ознак, використовуючи статистику, отриману з навчального набору даних.

Підтримка графічних процесорів. CatBoost забезпечує підтримку GPU, що дозволяє прискорити навчання та виведення на апаратному забезпеченні GPU. Це особливо корисно при роботі з великими наборами даних і складними моделями.

Автоматичне масштабування функцій. CatBoost автоматично виконує масштабування елементів.

Високо оптимізована реалізація. CatBoost реалізовано на C++ для ефективних обчислень та продуктивності. Він оптимізований для роботи з великими наборами даних і може використовувати паралельну обробку для прискорення завдань навчання і прогнозування.

Стійкість до зашумлених даних. CatBoost розроблений для того, щоб бути стійким до зашумлених даних та викидів. Він може обробляти зашумлені або неправильні мітки в навчальних даних без значного негативного впливу на продуктивність.

Налаштування гіперпараметрів. CatBoost надає ряд гіперпараметрів, які можна налаштувати для оптимізації роботи моделі. До них відносяться параметри, пов'язані зі швидкістю навчання, глибиною дерева, регуляризацією тощо. Ефективне налаштування гіперпараметрів може додатково покращити точність та узагальнення моделі.

CatBoost надає метрики важливості ознак, які вказують на внесок кожної ознаки в прогнозі моделі. Ця інформація може бути використана для вибору ознак, розуміння процесу прийняття рішень моделлю та отримання уявлення про дані.

CatBoost набув популярності в різних сферах, включаючи класифікацію, регресію та завдання ранжування. Здатність працювати з категоріальними ознаками, відновлювати відсутні значення та надавати високоякісні прогнози робить його потужним інструментом в арсеналі машинного навчання.

Однак, як і будь-яка модель, CatBoost має певні недоліки:

Інтенсивні обчислення. Робота CatBoost з категоріальними ознаками та іншими розширеними функціями може зробити його більш трудомістким у порівнянні з деякими іншими алгоритмами машинного навчання. Навчання та прогнозування за допомогою CatBoost може зайняти більше часу та обчислювальних ресурсів, особливо на великих наборах даних.

Більший обсяг пам'яті. Розширені функції CatBoost та робота з категоріальними змінними можуть призвести до збільшення обсягу пам'яті

порівняно з простішими алгоритмами. Це слід враховувати при роботі в середовищах з обмеженою пам'яттю[15].

Обмежена документація. Хоча CatBoost має документацію, вона вважається менш вичерпною порівняно з іншими добре відомими бібліотеками машинного навчання.

2.2. Логістична регресія

Логістична регресія - це метод статистичного моделювання, який використовується для прогнозування ймовірності бінарного результату на основі однієї або декількох незалежних змінних. Це тип регресійного аналізу, де залежна змінна є категоричною або бінарною (наприклад, так/ні, 0/1)[16].

Ключові особливості логістичної регресії:

Бінарний результат. Логістична регресія використовується, коли залежна змінна є бінарною, тобто може набувати лише двох можливих значень (наприклад, успіх/неуспіх, істина/хибність). Мета полягає в тому, щоб змоделювати ймовірність того, що результат буде в одній категорії на основі значень незалежних змінних.

Логістична функція. Логістична регресія застосовує логістичну або сигмоїдну функцію для перетворення результату лінійної регресії у значення ймовірності між 0 і 1.

Логістичні шанси. Логістична регресія моделює лог-шанси або логіт ймовірності настання події. Логістичні шанси являють собою логарифм відношення шансів, тобто відношення ймовірності настання події до ймовірності того, що вона не відбудеться.

Припущення лінійності. Логістична регресія припускає, що зв'язок між незалежними змінними та лог-шансом результату є лінійним. Вона також припускає відсутність мультиколінеарності (високої кореляції) між незалежними змінними і те, що спостереження є незалежними одне від одного.

Оцінка максимальної правдоподібності. Логістична регресія оцінює коефіцієнти незалежних змінних за допомогою методу найменших квадратів (МНК).

Коефіцієнти та відношення шансів. Коефіцієнти, отримані в результаті логістичної регресії, відображають зміну лог-шансів результату при зміні відповідної незалежної змінної на одиницю. Відношення шансів, яке є експонентою коефіцієнтів, показує, як змінюються шанси результату при збільшенні незалежної змінної на одиницю.

Багатовимірна логістична регресія. Логістична регресія може працювати з декількома незалежними змінними, що дозволяє моделювати більш складні взаємозв'язки між предикторами і бінарним результатом. У багатовимірній логістичній регресії коефіцієнти відображають зміну лог-шансів результату при зміні відповідної незалежної змінної, а інші змінні залишаються сталими.

Широке прикладне застосування. Логістична регресія широко використовується в різних галузях, включаючи медицину, соціальні науки, маркетинг, фінанси тощо. Вона зазвичай використовується для таких завдань, як прогнозування наявності хвороб, відтік клієнтів, ймовірності неповернення кредитів та виявлення шахрайства.

Також варто розглянути недоліки даної моделі:

Припущення про лінійність. Логістична регресія передбачає лінійний зв'язок між незалежними змінними та лог-шансом результату. Якщо зв'язок нелінійний, логістична регресія може не відобразити його точно. У таких випадках більш доречними можуть бути більш складні методи моделювання.

Обмежена бінарним результатом. Логістична регресія призначена для бінарних результатів і може не підходити для моделювання результатів з декількома категоріями. Вона не може обробляти залежні змінні з більш ніж двома рівнями без модифікацій або розширень.

Чутливість до пропусків. Логістична регресія може бути чутливою до пропусків у даних. Пропуски, особливо значущі, можуть сильно впливати на оцінені коефіцієнти і погіршувати результати роботи моделі. Виявлення

пропусків та відповідні методи їх обробки мають вирішальне значення для пом'якшення цієї проблеми.

Припущення про незалежність. Логістична регресія припускає, що спостереження в наборі даних є незалежними одне від одного. Порушення цього припущення, наприклад, наявність автокореляції або кластеризації даних, може призвести до упереджених оцінок коефіцієнтів і ненадійних висновків. Для роботи з такими структурами даних можуть знадобитися вдосконалені методи[17].

Відсутність автоматичного вибору ознак. Логістична регресія не виконує автоматичного відбору ознак і не обробляє надлишкові ознаки. Для визначення найбільш релевантних предикторів і зменшення ризику надмірної підгонки необхідно застосовувати методи відбору ознак, такі як покрокова регресія або методи регуляризації, такі як LASSO або гребенева регресія.

2.3. XGBoost

XGBoost (eXtreme Gradient Boosting) - популярний алгоритм машинного навчання, відомий своєю високою продуктивністю та ефективністю. Він є реалізацією фреймворку градієнтного бустінгу і широко використовується як для задач регресії, так і для задач класифікації. XGBoost будує потужну ансамблеву модель шляхом послідовного об'єднання декількох слабких моделей прогнозування, як правило, дерев рішень[18].

Ключові особливості моделі XGBoost:

Градієнтний бустинг. XGBoost працює за принципом градієнтного бустингу, коли кожна наступна модель навчається виправляти помилки, допущені попередніми моделями. Цей ітеративний процес призводить до створення сильної прогнозної моделі, яка поєднує прогнози декількох слабких моделей.

Налаштовувані функції втрат. XGBoost дозволяє користувачам визначати власні функції втрат, які можуть бути особливо корисними при

роботі зі спеціалізованими задачами або унікальними проблемними областями. Це забезпечує гнучкість у вирішенні різних типів проблем та оптимізації для конкретних цілей.

Обробка відсутніх значень. XGBoost має вбудовані механізми для обробки відсутніх значень у вхідних даних. Під час навчання він автоматично вивчає найкращий напрямок для обробки відсутніх значень на основі наявної інформації в наборі даних.

Важливість ознаки. XGBoost надає метрики важливості ознак, які кількісно оцінюють внесок кожної ознаки в прогнози моделі. Ця інформація може бути використана для вибору функцій, визначення важливих змінних і отримання уявлення про дані.

Підтримка паралельних і графічних процесорів. XGBoost підтримує паралельну обробку і може використовувати кілька ядер процесора для прискорення завдань навчання і прогнозування. Крім того, він пропонує прискорення на графічному процесорі, що забезпечує ще більш швидкі обчислення і масштабованість для великих наборів даних.

Деревоподібна обрізка. XGBoost застосовує техніку під назвою "tree pruning", щоб зменшити перенавчання та покращити здатність моделі до узагальнення. Обрізка видаляє непотрібні гілки з дерев рішень, щоб спростити модель, зберігаючи при цьому її прогностичну силу.

Рання зупинка. XGBoost підтримує ранню зупинку, техніку, яка дозволяє зупинити навчання на ранній стадії, коли продуктивність моделі на валідаційному наборі даних перестає покращуватися. Це допомагає запобігти надмірному пристосуванню та економить обчислювальні ресурси.

Налаштування гіперпараметрів. XGBoost має декілька гіперпараметрів, які можна налаштовувати для оптимізації його роботи під конкретну задачу. Правильне налаштування гіперпараметрів має вирішальне значення для досягнення найкращої продуктивності XGBoost.

XGBoost здобув значну популярність і успішно застосовується в різних сферах і змаганнях завдяки своїй високій точності прогнозування,

масштабованості та гнучкості. Це потужний та універсальний інструмент як для дослідницьких, так і для виробничих завдань машинного навчання.

Також варто знати і розуміти потенційні слабкі місця даної моделі:

Інтенсивні обчислення. XGBoost може вимагати значних обчислень, особливо при роботі з великими наборами даних або складними моделями. Навчання моделі XGBoost може зайняти багато часу, а алгоритм може вимагати значних обчислювальних ресурсів, особливо якщо ви використовуєте велику кількість дерев або виконуєте налаштування гіперпараметрів.

Налаштування гіперпараметрів. XGBoost має декілька гіперпараметрів, які необхідно налаштувати для досягнення оптимальної продуктивності. Пошук правильної комбінації гіперпараметрів може зайняти багато часу і вимагати ретельних експериментів та перехресної перевірки. Неправильне налаштування гіперпараметрів може призвести до неоптимальних результатів або навіть до надмірного налаштування.

Потреба в достатній кількості навчальних даних. XGBoost зазвичай вимагає достатньої кількості навчальних даних для хорошої роботи. Коли набір даних малий або незбалансований, існує вищий ризик перенавчання, і продуктивність моделі може бути не такою високою, як з більшими та збалансованішими наборами даних.

Відсутність вбудованої підтримки текстових даних. XGBoost в першу чергу розроблений для числових і категоріальних ознак і не має вбудованої підтримки для обробки та роботи з текстовими даними. Якщо ваш набір даних містить значну кількість текстової інформації, вам може знадобитися попередня обробка та перетворення даних у числове або категоріальне представлення перед використанням XGBoost.

Використання пам'яті. Споживання пам'яті XGBoost може бути відносно високим, особливо при роботі з великими наборами даних або складними моделями. Вимоги до пам'яті можуть обмежити його використання в

середовищах з обмеженою пам'яттю або при роботі з дуже великими наборами даних.

Чутливість до викидів. Як і інші деревоподібні алгоритми, XGBoost чутливий до викидів у даних. Викиди можуть мати значний вплив на структуру дерев рішень і можуть призвести до неоптимальної роботи моделі. Тому перед застосуванням XGBoost важливо попередньо обробити дані та належним чином обробити викиди[19].

Незважаючи на ці обмеження, XGBoost залишається потужним і універсальним алгоритмом, який може забезпечити відмінні результати прогнозування в широкому діапазоні сценаріїв. Розуміючи його обмеження і розглядаючи їх в контексті вашої конкретної проблеми, ви можете приймати обґрунтовані рішення про те, коли і як ефективно використовувати XGBoost.

Розділ 3. Прогнозування ймовірності оплати у мобільному додатку

3.1. Практичне застосування моделі для прогнозування ймовірності оплати

Прогнозування ймовірності оплати в мобільному додатку для знайомств може запропонувати багато переваг для бізнесу.

Області застосування прогнозу ймовірності оплати:

Прогнозування доходів. Прогнозуючи ймовірність оплати, можна оцінити потенційний дохід, який генерує мобільний додаток для знайомств. Ця інформація є цінною для фінансового планування, бюджетування та встановлення цільових показників доходу. Вона дозволяє оцінити фінансову життєздатність додатку і приймати обґрунтовані рішення щодо ціноутворення, моделей підписки або стратегій монетизації.

Сегментація користувачів. Прогнозування ймовірності опдати може допомогти сегментувати користувачів. Така сегментація дозволяє адаптувати маркетингові стратегії, рекламні пропозиції та користувацький досвід на основі різних сегментів користувачів. Наприклад, можна зосередити свої зусилля на користувачах з високою ймовірністю оплати, щоб заохотити їх оновити підписку або здійснити покупку в додатку.

Персоналізовані маркетингові кампанії. Прогнозуючи ймовірність оплати, можна розробляти цільові маркетингові кампанії для оптимізації залучення та утримання користувачів. Можна визначити користувачів, які з більшою ймовірністю перетворюються на платоспроможних клієнтів, і розробити персоналізовані повідомлення або стимули, щоб заохотити їх здійснити платіж. Такий підхід може підвищити ефективність маркетингу і збільшити коефіцієнт конверсії.

Оптимізація функціоналу додатку. Прогнозування ймовірності оплати може дати уявлення про функціонал або аспекти додатку, які сприяють залученню користувачів і конверсії. Аналізуючи взаємозв'язок між

функціоналом та ймовірністю оплати, можна визначити ключові фактори, що впливають на платіжну поведінку. Ця інформація може спрямовувати зусилля з розробки продукту, визначати пріоритети покращення функцій та оптимізувати користувацький досвід, щоб збільшити ймовірність оплати.

Прогнозування та утримання відтоку. Прогнозування ймовірності оплати також може допомогти у прогнозуванні відтоку та утриманні клієнтів. Відстежуючи ймовірність оплати в часі, можна виявити користувачів, які схильні до ризику відтоку або припинення підписки. Це дозволить вжити проактивних заходів, таких як персоналізовані пропозиції, оновлення функцій або втручання служби підтримки, щоб утримати цих користувачів і підвищити ймовірність їхніх платежів.

Оцінка ефективності бізнесу. Ймовірність оплати є важливим показником ефективності мобільного додатку для знайомств. Відстежуючи ймовірність оплати користувацької бази, можна оцінити ефективність стратегій монетизації, маркетингових кампаній та вдосконалення продукту. Це допоможе виміряти успіх бізнес-ініціатив і прийняти рішення на основі даних для оптимізації отримання прибутку.

Оцінка життєвої цінності клієнта (Customer Lifetime Value, CLV): Оцінка життєвої цінності користувача має вирішальне значення для розуміння довгострокової прибутковості додатку. Прогнозування ймовірності оплати дозволяє оцінити потенційний дохід, який користувач може згенерувати протягом усього свого життєвого циклу. Ця інформація допомагає оцінити рентабельність інвестицій (ROI) у залучення та утримання користувачів, що дає змогу ефективно розподіляти ресурси та зосередитися на найцінніших користувачах.

Загалом, прогнозування ймовірності оплати в мобільному додатку для знайомств дає практичну інформацію для прогнозування доходів, сегментації користувачів, персоналізованого маркетингу, оптимізації функцій, прогнозування відтоку, оцінки ефективності бізнесу та оцінки CLV. Це

покращує розуміння поведінки користувачів і полегшує прийняття стратегічних рішень, що сприяють зростанню та прибутковості бізнесу.

Найбільш оптимальним застосуванням моделі для мого продукту буде відключення монетизаційних обмежень для когорти користувачів, які мають низьку ймовірність оплати.

Бізнес-модель продукту “Taimi” побудована на тому, що користувачі оформляють підписку для отримання додаткових функцій та можливостей у додатку. Найбільш бажаною і популярною функцією є перегляд тих користувачів, які уже вподобали тебе. Це значно економить час пошуку партнера і дає точне розуміння хто тебе може лайкнути.

З іншого боку користувачі без підписки є, свого роду, генераторами активності, які забезпечують підписників стимулами до повторного заходу в додаток і подальшого поновлення підписки.

Продукт постійно балансує між тим, щоб дати користувачам більше свободи, тим самим підвищити активність та зменшити відтік користувачів, та тим, щоб збільшити середній заробіток з одного користувача, що ймовірно призведе до зменшення утримання користувачів на продукті та точно зменшить активність.

Застосування моделі, яка прогнозує ймовірність оплати, дозволяє контролювати цей баланс на продукті і якісно надавати достатньо активності(вподобань) для кожного з користувачів, без втрати у монетизаційних метриках.

Для практичної реалізаціх пропонується наступна послідовність дій - кожного ранку модель буде проганятися на наборі користувачів, які були у додатку хоча б раз в останні 7 днів. Таким чином ми зможемо гарантовано охопити більше 90% користувачів(за інформацією “Taimi” щоденно лише 7% користувачів повертаються після відсутності у додатку більше ніж 7 днів), які зайдуть у додаток поточного дня.

Для кожного з користувачів буде спрогнозована ймовірність зробити покупку у наступні 7 днів. Буде встановлений трешхолд, який буде

сегментувати користувачів. Якщо ймовірність нижча ніж цей трешхолд, то користувач потрапляє до когорти тих, кого ми вважаємо користувачами з низькою ймовірністю оплати.

Наступний крок - зняття обмежень з користувачів з низькою ймовірністю оплати. Ці люди протягом дня будуть мати можливість переглядати тих, хто лайкнув. Таким чином користувачі будуть продукувати більше лайків у відповідь, отримувати більше взаємних симпатій, частіше повертатися у додаток та провокувати на покупку інших користувачів, у яких ще немає підписки.

3.2. Кластеризація користувачів додатку

Першим кроком у створенні будь-яких моделей є збір та обробка даних, на основі яких буде робитися прогноз. У нашому випадку такими даними будуть конкретні метрики та атрибути кожного окремого користувача. Задля максимізації точності прогнозу необхідно всебічно розглянути дії та характеристики користувачів. Тому у дослідженні будуть розглянуті наступні дані:

- *Демографічні дані користувача.* Розглянемо демографічну інформацію, таку як вік, стать, місцезнаходження, професія та рівень освіти. Ці змінні можуть допомогти виявити відмінності в платіжній поведінці різних сегментів користувачів.
- *Показники взаємодії з додатком*. Використаємо функції, які кількісно оцінюють взаємодію користувача з додатком, такі як кількість надісланих/отриманих повідомлень, частота входів у додаток, тривалість сесій у додатку та кількість знайдених збігів. Вищий рівень залученості часто вказує на вищу ймовірність оплати[21].
- *Повнота профілю*. Виміряємо повноту профілів користувачів, включаючи наявність фотографій профілю, детальний опис особистості та інтересів. Користувачі з більш повними профілями можуть

демонструвати більшу прихильність до додатку, що збільшує ймовірність оплати.

- *Історія підписок*. Включимо дані про історію підписки користувача, такі як тип підписки (наприклад, безкоштовна, пробна, преміум), тривалість підписки, а також будь-які минулі скасування або поновлення. Ця інформація може допомогти оцінити схильність користувача до перетворення на платоспроможного клієнта.
- *Покупки в додатку.* Розглянемо змінні, пов'язані з покупками в додатку, такі як взаємодія з платіжними екранами та використання преміум функціоналу. Користувачі, які раніше використовували преміум функціонал, з більшою ймовірністю захочуть платити в майбутньому, щоб використати його знову.
- *Соціальна взаємодія*. Оцінимо метрики, пов'язані з соціальною взаємодією в додатку, такі як кількість вподобань, які користувач отримує у своєму профілі. Позитивні соціальні відгуки можуть позитивно впливати на платіжну поведінку.
- *Поведінкова сегментація*. Сегментуємо користувачів на основі їхніх поведінкових моделей, таких як частота використання, інтенсивність взаємодії. Побудова окремих моделей для кожного сегмента або включення специфічних особливостей сегмента може підвищити точність прогнозування[22].

Отже, база даних була зібрана за 20 календарних днів. Об’єм вибірки – 1852131 записів[13]. У неї входять характеристики користувачів та метрики їх взаємодії з додатком. Зібрані дані згруповані по даті та user_id(унікальний ідентифікатор користувача). Дані містять лише тих користувачів, які були активні в день збору інформації. Тобто ті, які хоча б раз за день зайшли у додаток.

Після ретельного аналізу поведінки користувачів, та їх взаємодії з додатком були виведені та відібрані наступні метрики:

Цільова метрика:

is_payer - ідентифікатор того, чи здійснив користувач оплату у наступні 7 днів після збору інформації про нього. Набуває значень 1 або 0. Важливо, що метою моделі - є прогнозування ймовірності здійснення оплати у наступні 7 днів, тому що таке обмеження спрощує задачу прогнозування і дозволяє більш точно використати результати роботи моделі. Отже, ця метрика є цільовою.

Демографічні метрики:

job - область роботи користувача. Обирає самостійно, на етапі реєстрації в додатку.

geo - регіон проживання даного користувача. Може набувати одного з наступних значень: LATAM, SEA, EUR, MENA, ENG, CHINA, CIS, OTHER, JAPAN, AFRICA, USA.

country_group - групування користувачів за ознакою країни проживання.

Визначається за наступним правилом:

```
multiIf(country_code = 'US', 'US',
        country_code = 'UK', 'UK',
        country_code in ('FR', 'AU', 'DE'), 'EU', 'Other')
```

age - вік користувача, надається ним же, на етапі реєстрації.

gender - гендер користувача. Обирається на етапі реєстрації, не може бути змінений пізніше.

l_for - комбінація гендерів, у яких зацікавлений користувач. Кожен користувач при реєстрації робить вибір, які гендери підходять йому. Створюється з’єднанням стрічок, де кожен гендер отримує у відповідність 0 або 1:

```
concat('m', toString(looking_for1),
    '_f', toString(looking_for2),
    '_tm', toString(looking_for3),
    '_tf', toString(looking_for4),
    '_i', toString(looking_for5),
    '_o', toString(looking_for6)
    )
```

Наприклад, якщо користувач шукає лише гендер “жінка”, то отримає наступну комбінацію - ‘m0_f1_tm0_tf0_i0_o0’

Сталі метрики або ті, які визначаються на етапі реєстрації:

lt(lifetime) - тривалість життя користувача на продукті. Визначається, як різниця у кількості днів між поточною датою та датою реєстрації. Може набувати значень у проміжку від 0 до 1800. Верхня межа - дата створення продукту.

ln_lt - натуральний логарифм від тривалості життя користувача на продукті. Використовується для зменшення впливу великих значень даної метрики. Оскільки життя користувача у більш пізні лайфтайми відрізняється між собою значно менше ніж життя у перші дні.

group_lt - метрика, яка ділить користувачів на різні групи за ознакою тривалості їх життя на продукті. Відбуваться за наступним правилом:

multiIf(lt is null, 'not',
lt < 7, '1-7',
lt < 20, '8-19', '20+')

reg_source - спосіб реєстрації користувача. При скачуванні додатку користувачу надається вибір способу реєстрації. Можливі значення - apple(реєстрація через apple_id), snap(реєстрація через аккаунт Snapchat), fb(реєстрація через додаток Facebook), google(реєстрація через аккаунт Google), form(реєстрація через будь-яку пошту), phone(реєстрація за номером телефону).

platform - платформа, на якій встановлений додаток. Може набувати значень ‘Android’, ‘IOS’, ‘Other’ - тобто залежить від типу девайсу та операційної системи, яка встановлена на ньому.

lcnt_users_on_person - кількість користувачів, яких асоціюють з персоною даного користувача. Персона - це результат роботи моделі, яка визначає чи була певна людина зареєстрована у додатку раніше. Такі випадки можуть траплятися, якщо людина видалила свій профіль і зареєструвала новий або змінила девайс і втратила доступ до свого профілю. Для кращого розуміння

поведінки таких користувачів і виділення їх в окремий сегмент на нашому продукті використовується таке поняття, як персона. Воно збирає під себе усі аккаунти однієї людини. Тобто одна персона може мати кілька аккаунтів, але певний аккаунт має лише одну персону.

is_many_person - показник того чи на даній персоні є інші аккаунти, крім того, що розглядається.

gmail - показник того, чи пошта користувача містить в собі домен gmail. Пошта також може бути відсутня.

private_email - показник того, чи пошта користувача містить в собі домен privaterelay. Пошта також може бути відсутня.

not_gmail - показник того, чи пошта користувача не містить в собі домен gmail або домен privaterelay. Пошта також може бути відсутня.

good_name - показник того чи у користувача введене коректне ім’я. Тобто ім’я повинно містити великі літери та хоча б один пробіл.

isp - назва провайдера, який надає послуги інтернету даному користувачу. Може набувати наступних значень: movistar, metro, orange, t-mobile, verizon, vodafone, Other.

Метрики активності та залученості:

days_from_last_login - кількість днів, які пройшли від попереднього логінодня(день, в який юзер відкрив додаток хоча б один раз) до поточної дати. Може набувати значень від 0 до 180.

ln_days_from_last_login - натуральний логарифм кількісті днів, які пройшли від попереднього логінодня(день, в який юзер відкрив додаток хоча б один раз) до поточної дати.

count_logins - кількість логіноднів за попередні 180 днів.

count_logins_7d - кількість логіноднів за попередні 7 днів.

count_logins_30d - кількість логіноднів за попередні 30 днів.

ln_count_logins - натуральний логарифм від кількості логінів за останні 180 днів.

ln_likes_received - натуральний логарифм від кількості отриманих лайків у додатку.

is_likes_received - показник наявності хоча б одного отриманого лайку у певний логінодень.

ln_likes_sent - натуральний логарифм від кількості відправлених лайків у додатку.

is_likes_sent - показник наявності хоча б одного відправленого лайку у певний логінодень.

skips_sent - кількість переглянутих користувачів, які не були лайкнуті даним користувачем, тобто кількість скіпів.

is_skips_sent - показник наявності хоча б одного відправленого скіпа у певний логінодень.

skips_received - кількість отриманих скіпів за день.

is_skips_received - показник наявності хоча б одного отриманого скіпа у певний логінодень.

mutuals_sender - кількість відправлених взаємних симпатій за день. Тобто тих взаємних симпатій, де користувач лайкає іншу людину у відповідь.

is_mutuals_sender - показник наявності хоча б одної відправленої взаємної симпатії у певний логінодень.

mutuals_receiver - кількість отриманих взаємних симпатій за день. Тобто тих взаємних симпатій, де користувач перший лайкає іншу людину, а вона реагує лайком у відповідь.

is_mutuals_receiver - показник наявності хоча б одної отриманої взаємної симпатії у певний логінодень.

first_messages_sent - кількість діалогів, у яких користувач першим відправив повідомлення.

is_first_messages_sent - показник наявності хоча б одного діалогу, у якому користувач першим відправив повідомлення у певний логінодень.

first_messages_received - кількість діалогів, у яких користувач першим отримав повідомлення.

is_first_messages_received - показник наявності хоча б одного діалогу, у якому користувач першим отримав повідомлення у певний логінодень.

total_messages_sent - загальна кількість повідомлень, яку надіслав користувач за певний логінодень.

is_total_messages_sent - показник наявності хоча б одного відправленого повідомлення за день.

total_messages_received - загальна кількість повідомлень, яку отримав користувач за певний логінодень.

is_total_messages_received - показник наявності хоча б одного отриманого повідомлення за день.

me_section_views - кількість переглядів власного профілю за день.

is_me_section_views - показник хоча б одного перегляду профілю за певний день.

ln_notify_section_views - натуральний логарифм від кількості переглядів розділу нотифікацій у додатку за день.

is_notify_section_views - показник наявності хоча б одного заходу у розділ нотифікацій у додатку за день.

messages_section_views - кількість переглядів розділу повідомлень у додатку за день.

is_messages_section_views - показник наявності хоча б одного перегляду розділу повідомлень за день.

dialogs - кількість нових діалогів, у яких брав участь користувач. Тобто таких, які були створені у цей же день.

is_dialogs - показник наявності нових діалогів, у яких брав участь користувач у певний день.

dialogs_all - кількість усіх діалогів, у яких брав участь користувач. Тобто включені такі, які могли бути створені не у цей день.

is_dialogs_all - показник наявності будь-яких діалогів, у яких брав участь користувач у певний день.

was_in_low_pay - показник того, чи перебував колись користувач під дією функціоналу ‘low pay probability’. Тобто функціоналу, який визначає користувачів, які з малою ймовірністю проведуть оплату і тому відключає для них основні монетизаційні обмеження.

is_push_enabled - показник того чи ввімкнені у користувача пуш-нотифікації у певний день.

cards_count - кількість карток створених користувачем на момент збору даних. Може включати медіа картки, текстові картки та біо-картку, де користувач розказує про себе.

Метрики взаємодії з платіжними екранами та преміум функціоналом:

is_pw_blurs - показник хоча б одного перегляду сторінки, яка виникає після натискання на користувача, профіль якого можна переглянути лише з преміум-аккаунтом.

cnt_pw_blurs_7d - кількість переглядів сторінки, яка виникає після натискання на користувача, профіль якого можна переглянути лише з преміум-аккаунтом.

days_from_last_pw_blurs - кількість днів, які пройшли з останнього перегляду сторінки, яка виникає після натискання на користувача, профіль якого можна переглянути лише з преміум-аккаунтом.

is_pw_boost - показник хоча б одного перегляду сторінки, яка виникає після спроби використати функціонал збільшення активності.

cnt_pw_boost_7d - кількість переглядів сторінки, яка виникає після спроби використати функціонал збільшення активності.

days_from_last_pw_boost - кількість днів, які пройшли з останнього перегляду сторінки, яка виникає після спроби використати функціонал збільшення активності.

is_pw_spot_search - показник хоча б одного перегляду сторінки, яка виникає після спроби використати функціонал зміни локації.

cnt_pw_spot_search_7d - кількість переглядів сторінки, яка виникає після спроби використати функціонал зміни локації.

days_from_last_pw_spot_search - кількість днів, які пройшли з останнього перегляду сторінки, яка виникає після спроби використати функціонал зміни локації.

is_pw_visitors - показник хоча б одного перегляду сторінки, яка виникає після спроби використати функціонал перегляду тих профілів, яка зробили візит твого профілю.

cnt_pw_visitors_7d - кількість переглядів сторінки, яка виникає після спроби використати функціонал перегляду тих профілів, яка зробили візит твого профілю.

days_from_last_pw_visitors - кількість днів, які пройшли з останнього перегляду сторінки, яка виникає після спроби використати функціонал перегляду тих профілів, яка зробили візит твого профілю.

is_pw_sub_store - показник хоча б одного перегляду розділу ‘subscription store’.

cnt_pw_sub_store_7d - кількість переглядів розділу ‘subscription store’

days_from_last_pw_sub_store - кількість днів, які пройшли з останнього перегляду розділу ‘subscription store’.

is_pw_rollback - показник хоча б одного перегляду сторінки, яка виникає після спроби використати функціонал rollback.

cnt_pw_rollback_7d - кількість переглядів сторінки, яка виникає після спроби використати функціонал rollback.

days_from_last_pw_rollback - кількість днів, які пройшли з останнього перегляду сторінки, яка виникає після спроби використати функціонал rollback.

is_pw_superlike - показник хоча б одного перегляду сторінки, яка виникає після спроби використати функціонал superlike.

cnt_pw_superlike_7d - кількість переглядів сторінки, яка виникає після спроби використати функціонал superlike.

days_from_last_pw_superlike - кількість днів, які пройшли з останнього перегляду сторінки, яка виникає після спроби використати функціонал superlike.

is_pw_limit_swipe - показник хоча б одного перегляду сторінки, яка виникає після того, як користувач використав усі свої перегляди профілів.

cnt_pw_limit_swipe_7d - кількість переглядів сторінки, яка виникає після того, як користувач використав усі свої перегляди профілів.

days_from_last_pw_limit_swipe - кількість днів, які пройшли з останнього перегляду сторінки, яка виникає після того, як користувач використав усі свої перегляди профілів.

is_spent_boost - показник хоча б одного використання функціоналу, який збільшує вхідну активність.

cnt_spent_boost_7d - кількість використання функціоналу, який збільшує вхідну активність.

days_from_last_spent_boost - кількість днів, які пройшли з останнього використання функціоналу, який збільшує вхідну активність.

is_spent_super_like - показник хоча б одного використання функціоналу ‘super like’.

cnt_spent_super_like_7d - кількість використання функціоналу ‘super like’.

days_from_last_spent_super_like - кількість днів, які пройшли з останнього використання функціоналу ‘rollback’.

is_spent_rollback - показник хоча б одного використання функціоналу ‘rollback’.

cnt_spent_rollback_7d - кількість використання функціоналу ‘super like’.

days_from_last_spent_rollback - кількість днів, які пройшли з останнього використання функціоналу ‘rollback’.

is_click_buy_bloor - показник хоча б одного натискання на кнопку покупки на екрані ‘bloor’.

cnt_click_buy_bloor_7d - кількість натискань на кнопку покупки на екрані ‘bloor’.

days_from_last_click_buy_bloor - кількість днів, які пройшли з останнього натискання на кнопку покупки на екрані ‘bloor’.

is_click_buy_rollback - показник хоча б одного натискання на кнопку покупки на екрані rollback.

cnt_click_buy_rollback_7d - кількість натискань на кнопку покупки на екрані rollback.

days_from_last_click_buy_rollback - кількість днів, які пройшли з останнього натискання на кнопку покупки на екрані rollback.

is_click_buy_sub_store - показник хоча б одного натискання на кнопку покупки у розділі sub_store.

cnt_click_buy_sub_store_7d - кількість натискань на кнопку покупки у розділі sub_store.

days_from_last_click_buy_sub_store - кількість днів, які пройшли з останнього натискання на кнопку покупки у розділі sub_store.

is_click_buy_super_like - показник хоча б одного натискання на кнопку покупки на екрані buy_super_like.

cnt_click_buy_super_like_7d - кількість натискань на кнопку покупки на екрані buy_super_like.

days_from_last_click_buy_super_like - кількість днів, які пройшли з останнього натискання на кнопку покупки на екрані buy_super_like.

is_click_buy_visitor - показник хоча б одного натискання на кнопку покупки на екрані buy_visitor.

cnt_click_buy_visitor_7d - кількість натискань на кнопку покупки на екрані buy_visitor.

days_from_last_click_buy_visitor - кількість днів, які пройшли з останнього натискання на кнопку покупки на екрані buy_visitor.

is_pw_mixlim - показник хоча б одного перегляду екрана, який виникає після натикання на функціонал, який доступний лише з преміум підпискою.

cnt_pw_mixlim_7d - кількість натискань на екран, який виникає після натикання на функціонал, який доступний лише з преміум підпискою.

days_from_last_pw_mixlim - кількість днів, які пройшли з останнього натискання на екран, який виникає після натикання на функціонал, який доступний лише з преміум підпискою.

is_pw_mixlim - показник хоча б одного перегляду екрана розділу ‘shop’.

cnt_pw_mixlim_7d - кількість натискань на екран розділу ‘shop’

days_from_last_pw_mixlim - кількість днів, які пройшли з останнього натискання на екран розділу ‘shop’.

Отже, фінальний набір даних для дослідження матиме наступний вигляд (Таблиця 1).

Таблиця 1

date	user_id	is_payer	l_for	…	…	days_from_last_pw_mixlim
2023-05-01	3984274	1	m0_f1_tm0_tf0_i0_o0	…	…	4

Тобто для кожної комбінації користувач - день логіну буде виведено всі метрики, які були описані вище.

3.3. Прогнозування ймовірності оплати

Подальша робота з готовими даними буде проводитися з допомогою програмного забезпечення Jupiter Notebook та мовою програмування Python.

Для реалізації моделей було використано наступні бібліотеки: pandas, numpy, sklearn, matplotlib та інші.

Перш за все дані були вивантажені з бази даних і перенесені у csv файли. Таке пристосування знадобилося, оскільки запит мовою SQL не зміг витягнути всі дані одночасно, довелося розділити його на 3 запити меншого об’єму, потім зберегти дані у 3 різних файла і згодом об’єднати їх у Jupiter Notebook у один набір даних.

Наступний крок - розподіл вибірки на тестову(1260408 записів – 70%) та тренувальну(540176 записів - 30%). Далі потрібно окремо виділити масив

номерів стовпчиків, у яких знаходяться категоріальні дані. Наступний крок - балансування вибірки. Серед усіх записів у зібраній базі лише 1.8% складають ті, які позначають успішну оплату. Тобто вибірка вкрай незбалансована і результати роботи моделі можуть бути погіршені. Через це потрібно зробити так, щоб модель навчалась на даних, які будуть мати однакову кількість записів. Для вирівнювання вибірки виділяємо записи які мають значення is_payer = 1. За допомогою рандомізатора вибираємо таку ж кількість записів із значенням is_payer = 0. Об'єднуємо ці два набори даних у одну вибірку, на основі якої проводимо навчання. Тренувальна вибірка складає 41692 записів.

Попередні кроки були спільні для усіх моделей. Наступні етапи дослідження описують застосування різних моделей з метою визначення найкращої для прогнозування ймовірності здійснення оплати у наступні 7 днів.

CatBoost

Перша використана модель для прогнозування - CatBoost. Алгоритм градієнтного бустингу з певними модифікаціями.

Для налаштування моделі були використані наступні параметри:

Loss_function = 'LogLoss',

Eval_metric = 'AUC',

Depth = 10,

12_leaf_reg = 1,

Iterations = 500,

Learning_rate = 0.1.

Зокрема важливим є метрика оптимізації – AUC. Площа під ROC-кривою (ROC AUC) - це метрика, яка кількісно оцінює загальну ефективність моделі бінарної класифікації. Вона відображає ймовірність того, що випадково вибраний позитивний приклад буде оцінений вище, ніж випадково вибраний негативний приклад. Іншими словами, показник ROC AUC вимірює здатність моделі розрізняти два класи.

Оцінка ROC AUC коливається від 0 до 1, де оцінка 1 вказує на ідеальний класифікатор, який може ідеально розділити позитивні та негативні приклади,

тоді як оцінка 0,5 вказує на класифікатор, який працює не краще, ніж випадковий вибір. Далі виводжу результати роботи моделі, зокрема точність та f1-score (Таблиця 2).

Таблиця 2

Класи	precision	recall	F1-score	Samples Count
0	1	0.88	0.93	545480
1	0.12	0.93	0.22	10160

Наступна метрика - AUC = 0.90.

Також для вірної інтерпретації результатів необхідно розуміти кількість true negative та false negative прогнозів (Таблиця 3).

Таблиця 3

результат	Відсоткове співвідношення класів, %	Кількість результатів у тестовій вибірці	Кількість результатів зважене на один день
False negative	0.05	289	48
False positive	20.85	115854	19309
True negative	77.30	429539	71589
True positive	1.80	9958	1659

Особливо важливо розуміти, як кількість false negative вплине на монетизацію продукту, а кількість true negative - на активність у додатку. Тому також виводжу ці метрики скориговані так, щоб можна було оцінити кількість користувачів у кожній когорті за один день. Для цього враховуємо, що тестові дані були зібрані за 20 днів та містять лише 30% від усіх записів.

Тобто монетизаційні обмеження будуть зняті для 71589 користувачів на день в середньому, а втрати у кількості підписників складатимуть 48 покупок.

Логістична регресія

Перш за все логістична регресія вимагає іншого поводження з категоріальними змінними. Необхідно створити комбінацію змінної та її значення та приєднати ці дані до набору даних замість стовбчиків з цими змінними. Використовую, так званий, “one hot encoding”. Також в процесі роботи з логістичною регресією важливо видалити рядки, які містять пропущені значення.

Таким чином ми зможемо передати у модель змінні, які не є числовими.

Наступний крок - застосування RFE. Тобто рекурсивне виключення ознак, яке базується на ідеї багаторазової побудови моделі та вибору найкращої або найгіршої ознаки, відкладання цієї ознаки в сторону, а потім повторення процесу з рештою ознак. Цей процес застосовується до тих пір, поки не будуть вичерпані всі ознаки в наборі даних. Метою RFE є вибір ознак шляхом рекурсивного розгляду все менших і менших наборів ознак.

Результат - зменшення кількості змінних, які не впливають на прогноз.

Наступним кроком проганяємо базову логістичну регресію та отримуємо оцінку p-value по кожній з незалежних змінних. Виключаємо з подальшої роботи ті, які більше ніж 0.05.

Після цього перепроганяємо ще раз і отримуємо фінальну модель(Додаток Г). На ній запускаємо заміряємо precision, recall, f1-score(Таблиця 4).

Таблиця 4

Класи	precision	recall	F1-score	Samples Count
0	0.98	1	0.99	528066
1	0.46	0.03	0.06	9111

Та AUC, який дорівнює 0.52. Крім того оцінюємо кількість, співвідношення та кількість на день false negative та true negative прогнозів(Таблиця 5).

Таблиця 5

результат	Відсоткове співвідношення класів, %	Кількість результатів у тестовій вибірці	Кількість результатів зважене на один день
False negative	1.64	8836	1472
False positive	0.06	320	53
True negative	98.24	527746	87957
True positive	0.06	275	45

XGBoost

Наступна модель так само, як і логістична регресія вимагає відповідної обробки категоріальних даних. Тому проводимо аналогічну підготовку.

Запускаємо модель на базових налаштуваннях. Отримуємо AUC, який дорівнює 0.54. Визначаємо кращі налаштування моделі з допомогою функції grid search.

Фінальні налаштування моделі:

```
xgb_cl = xgb.XGBClassifier(gamma = 1,
 learning_rate = 0.5,
 max_depth = 15,
 reg_lambda = 10,
 scale_pos_weight = 3)
```

Перезапускаємо модель і отримуємо наступні результати:

AUC = 0.68

Accuracy = 0.98

Заміряємо precision, recall, F1-score(Таблиця 6):

Таблиця 6

Класи	precision	recall	F1-score	Samples Count
0	0.99	1	0.99	528066
1	0.75	0.36	0.49	9111

Кількість, співвідношення та кількість на день false negative та true negative прогнозів(Таблиця 7).

Таблиця 7

результат	Відсоткове співвідношення класів, %	Кількість результатів у тестовій вибірці	Кількість результатів зважене на один день
False negative	1.06	5719	953
False positive	0.19	1062	177
True negative	98.13	527143	87857
True positive	0.6	3253	542

Підсумуємо результати роботи кожної з моделей (Таблиця 8).

Таблиця 8

Модель	Accuracy	AUC	True Negative	False Negative
CatBoost	0.88	0.90	71589	48
Logit	0.98	0.52	87957	1472
XGBoost	0.98	0.68	87856	953

Отже, отримані результати свідчать про перевагу моделей логістичної регресії та XGBoost у точності прогнозу, кількості True Negative прогнозів. При цьому AUC та кількість False Negative значно краще у моделі CatBoost.

Хоч CatBoost і програє у точності прогнозу, але інші моделі роблять дуже незбалансовані передбачення. Вони просто припускають, що майже всі користувачі не проведуть оплату. Це чітко видно, якщо звернути увагу на AUC, який дозволяє не лише оцінити точність прогнозу, але й враховує співвідношення класів цільової метрики.

Обов’язково потрібно також порівняти відсоток користувачів, які потраплять у когорту низької ймовірності оплати і не зроблять платіж. Тут ми бачимо перевагу уже Logit моделі.

Якщо ж говорити про практичне застосування моделей, то ключова метрика для оцінки ефективності роботи моделей - це кількість False Negative прогнозів. Тобто прогнозів, при яких користувач не зробить покупку, хоча мав би. Тут лідером знову є CatBoost, який дозволить втратити найменшу кількість грошей. Зокрема варто відзначити те, що середня кількість покупок на день складає близько 3200 штук. Використання Logit чи XGBoost значно зменшать доходи продукту і практично унеможливлять нормальну роботу компанії.

Отже, найкращим рішенням для використання на нашому продукті є модель CatBoost.

Висновки

Проведене дослідження дозволяє стверджувати, що алгоритми машинного навчання – ефективний інструмент для прогнозування ймовірності оплати у мобільному додатку. Вони дають змогу вагомо покращити продукт як у розрізі користувацького досвіду, так і з точки зору монетизації.

Модель CatBoost показала найкращі результати серед розглянутих моделей. Застосування даного методу дозволить отримати значну конкурентну перевагу на ринку додатків для знайомств. Користувачі отримають більше свободи дій у додатку та більш ймовірно проведуть більше часу, використовуючи продукт “Taimi”.

Варто відзначити, що модель XGBoost також показала себе непогано у задачі прогнозування ймовірності здійснення платежу. Вона гарно передбачила тих, хто справді не здійснить платежу, і тим самим могла збільшити вибірку тих, хто отримає розширений функціонал додатку без додаткової плати. Але за умови практичного використання вона призвела би до втрати рентабельності продукту, тому не може бути застосована.

Логістична регресія виявилась нездатною точно вирішити таку комплексну задачу. На її ефективність могли повпливати об’єм вибірки або можлива мультиколінеарність деяких змінних. Тому її застосування на продукті не принесе жодної користі.

Отже, моделі машинного навчання мають практичне застосування у сфері мобільних додатків. Завдяки ним можна сегментувати користувачів, прогнозувати ймовірність здійснення різних дій та на основі цього будувати продукт, який максимізує задоволеність користувачів та доходи розробників.

Список використаних джерел

1. 1. Deng Y. Spillover Effects and Freemium Strategy in the Mobile App Market [Електронний ресурс] / Y. Deng, A. Lambrecht, Y. Liu. – 2022 – Режим доступу до ресурсу: https://doi.org/10.1287/mnsc.2022.4619.

2. Arora S. The Implications of Offering Free Versions for the Performance of Paid Mobile Apps [Електронний ресурс] / S. Arora, F. ter Hofstede, V. Mahajan. – 2017. – Режим доступу до ресурсу: http://dx.doi.org/10.1509/jm.15.0205.

3. Deng Y. Deep Learning on Mobile Devices - A Review [Електронний ресурс] / Yunbin Deng. – 2019. – Режим доступу до ресурсу: https://doi.org/10.13140/RG.2.2.15012.12167.

4. m-Health 2.0: New perspectives on mobile health, Machine Learning and Big Data Analytics [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: https://doi.org/10.1016/j.ymeth.2018.05.015.

5. Hajek P. Fraud Detection in Mobile Payment Systems using an XGBoost-based Framework [Електронний ресурс] / Petr Hajek. – 2022. – Режим доступу до ресурсу: https://link.springer.com/article/10.1007/s10796-022-10346-6.

6. Mobile Application Market Size, Share, & Trends Analysis Report By Store Type (Google Store, Apple Store, Others), By Application, By Region, And Segment Forecasts, 2023 - 2030 [Електронний ресурс]. – 2022. – Режим доступу до ресурсу: https://www.grandviewresearch.com/industry-analysis/mobile-application-market.

7. Deubener J. A Typology of Freemium Business Models for Mobile Applications [Електронний ресурс] / Johannes Deubener. – 2016. – Режим доступу до ресурсу: https://www.researchgate.net/publication/303176091_A_Typology_of_Freemium_Business_Models_for_Mobile_Applications.

8. Pujol N. Freemium: Attributes of an Emerging Business Model [Електронний ресурс] / Nicolas Pujol. – 2010. – Режим доступу до ресурсу: https://doi.org/10.2139/ssrn.1718663.

9. Tafradzhiyski N. Mobile App Retention [Електронний ресурс] / Nayden Tafradzhiyski. – 2023. – Режим доступу до ресурсу: https://www.businessofapps.com/guide/mobile-app-retention/.

10. Santos-Vijande M. Building user engagement to mhealth apps from a learning perspective: Relationships among functional, emotional and social drivers of user value [Електронний ресурс] / María Leticia Santos-Vijande. – 2022. – Режим доступу до ресурсу: https://doi.org/10.1016/j.jretconser.2022.102956.

11. https://taimi.com/

12. Pagano D. User Feedback in the AppStore: An Empirical Study [Електронний ресурс] / Dennis Pagano. – 2013. – Режим доступу до ресурсу: https://doi.org/10.1109/RE.2013.6636712.

13. Посилання на повну базу даних: https://drive.google.com/drive/folders/1FH0tajWWQ13xoRNs_HXg6oTAomzjScSk?usp=sharing

14. Prokhorenkova L. CatBoost: unbiased boosting with categorical features [Електронний ресурс] / Liudmila Prokhorenkova // 2019 – Режим доступу до ресурсу: https://doi.org/10.48550/arXiv.1706.09516.

15. Hancock J. CatBoost for big data: an interdisciplinary review [Електронний ресурс] / John T. Hancock // 2020 – Режим доступу до ресурсу: https://doi.org/10.1186/s40537-020-00369-8.

16. Hosmer Jr D. Applied logistic regression [Електронний ресурс] / D. Hosmer Jr – Режим доступу до ресурсу: https://www.researchgate.net/profile/Andrew-Cucchiara/publication/261659875_Applied_Logistic_Regression/links/542c7eff0cf277d58e8c811e/Applied-Logistic-Regression.pdf.

17. Steyerberg E. Assessing the performance of prediction models: a framework for some traditional and novel measures [Електронний ресурс] / Ewout W. Steyerberg – Режим доступу до ресурсу: https://doi.org/10.1097%2FEDE.0b013e3181c30fb2.

18. Chen T. XGBoost: A Scalable Tree Boosting System [Електронний ресурс] / Tianqi Chen. – 2016. – Режим доступу до ресурсу: https://doi.org/10.1145/2939672.2939785.

19. Al Daoud E. Comparison between XGBoost, LightGBM and CatBoost Using a Home Credit Dataset [Електронний ресурс] / Essam Al Daoud – Режим доступу до ресурсу: https://doi.org/10.5281/zenodo.3607805.

20. BERGLING O. Evaluation of machine learning methods to predict payment preferences [Електронний ресурс] / OSCAR BERGLING. – 2019. – Режим доступу до ресурсу: https://www.diva-portal.org/smash/get/diva2:1373851/FULLTEXT01.pdf.

21. Zhao X. Sales Prediction and Product Recommendation Model Through User Behavior Analytics [Електронний ресурс] / Xian Zhao – Режим доступу до ресурсу: 10.32604/cmc.2022.019750.

22. Wang S. Calculating dating goals: data gaming and algorithmic sociality on Blued, a Chinese gay dating app [Електронний ресурс] / Shuaishuai Wang – Режим доступу до ресурсу: https://doi.org/10.1080/1369118X.2018.1490796.

Додатки

Додаток А. Код CatBoost.

```
#підключення бібліотек
from sklearn import datasets
from sklearn import metrics
from sklearn.model_selection import train_test_split
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use("ggplot")
import catboost as ctb
import os
os.environ['KMP_DUPLICATE_LIB_OK']='True'
#зчитування даних
first = pd.read_csv('activity_no_payers.csv')
second = pd.read_csv('mixlims_no_payers.csv')
for column in second:
    if second[column].dtype == float:
        second[column] = second[column].astype('int')
third = pd.read_csv('combine.csv')
print(first.shape)
print(second.shape)
print(third.shape)
#поєднання даних у один датасет
left_merged_reversed = pd.merge(
...     first, second, how="left", on=["date", "user_id", "is_payer"]
... )
left_merged_reversed = pd.merge(
...     left_merged_reversed, third, how="left", on=["date", "user_id"]
... )
print(left_merged_reversed.shape)
for col in left_merged_reversed.columns:
    print(col)
    print(left_merged_reversed[col].dtype)
dataset = left_merged_reversed
dataset = dataset.query('lt > 1')
dataset.shape
#Поділ на тренувальну та тестову вибірки
X = dataset
y = dataset.is_payer
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30)
#Виділення категоріальних змінних
```

```
categorical_features_indices = np.where(X.dtypes != float)[0]
categorical_features_indices
#Балансування вибірок
train_df = pd.concat([X_train,y_train],axis=1)
payer = train_df[train_df['is_payer']==1]
not_payer = train_df[train_df['is_payer']==0]
not_payer = not_payer.sample(n=len(payer), random_state=101)
train_df = pd.concat([payer,not_payer],axis=0)
X_train = train_df.drop('is_payer',axis=1)
y_train = train_df['is_payer']
#Налаштування моделі
model = ctb.CatBoostClassifier(loss_function='Logloss', eval_metric='AUC',
  depth = 10,
 l2_leaf_reg = 1,
 iterations = 500,
 learning_rate = 0.1)
model.fit(X_train, y_train, cat_features=categorical_features_indices)
print(model)
```

Додаток Б. Код алгоритму Logit.

```
#Видалення пропущених значень
dataset = dataset.dropna(axis=0)
#Обробка категоріальних даних
cat_vars=['group_lt','reg_source','platform','l_for','job','geo','country_group','isp',
'is_many_person']
for var in cat_vars:
    cat_list='var'+'_'+var
    cat_list = pd.get_dummies(X[var], prefix=var)
    data_1=X.join(cat_list)
    X=data_1
cat_vars=['group_lt','reg_source','platform','l_for','job','geo','country_group','isp',
'is_many_person']
data_vars=X.columns.values.tolist()
to_keep=[i for i in data_vars if i not in cat_vars]

data_final=X[to_keep]
data_final.columns.values
X = data_final
#Реалізація моделі
import statsmodels.api as sm
logit_model=sm.Logit(y,X)
result=logit_model.fit()
print(result.summary2())
```

Додаток В. Код алгоритму XGBoost.

```
#Видалення пропущених значень
dataset = dataset.dropna(axis=0)
#Обробка категоріальних даних
cat_vars=['group_lt','reg_source','platform','l_for','job','geo','country_group','isp',
'is_many_person']
for var in cat_vars:
    cat_list='var'+'_'+var
    cat_list = pd.get_dummies(X[var], prefix=var)
    data_1=X.join(cat_list)
    X=data_1
cat_vars=['group_lt','reg_source','platform','l_for','job','geo','country_group','isp',
'is_many_person']
data_vars=X.columns.values.tolist()
to_keep=[i for i in data_vars if i not in cat_vars]

data_final=X[to_keep]
data_final.columns.values
X = data_final
#Grid Search
param_grid = {
    "max_depth": [3, 4, 5, 7],
    "learning_rate": [0.1, 0.01, 0.05],
    "gamma": [0, 0.25, 1],
    "reg_lambda": [0, 1, 10],
    "scale_pos_weight": [1, 3, 5],
    "subsample": [0.8],
    "colsample_bytree": [0.5],
}
from sklearn.model_selection import GridSearchCV
xgb_cl = xgb.XGBClassifier(objective="binary:logistic")
grid_cv = GridSearchCV(xgb_cl, param_grid, n_jobs=-1, cv=3,
scoring="roc_auc")

_ = grid_cv.fit(X, y)
grid_cv.best_score_
grid_cv.best_params_
#Реалізація моделі
from sklearn.metrics import accuracy_score
xgb_cl = xgb.XGBClassifier(gamma = 1,
 learning_rate = 0.5,
 max_depth = 15,
 reg_lambda = 10,
 scale_pos_weight = 3)
```

```
xgb_cl.fit(X_train, y_train)
preds = xgb_cl.predict(X_test)
accuracy_score(y_test, preds)
```

Додаток Г. Звіт логістичної регресії.

```
Optimization terminated successfully.
         Current function value: 0.075649
         Iterations 11
                            Results: Logit
================================================================================
Model:                  Logit               Pseudo R-squared:    0.106
Dependent Variable:     is_payer            AIC:                 270988.3096
Date:                   2023-06-07 11:45    BIC:                 271459.4357
No. Observations:       1790589             Log-Likelihood:      -1.3546e+05
Df Model:               37                  LL-Null:             -1.5144e+05
Df Residuals:           1790551             LLR p-value:         0.0000
Converged:              1.0000              Scale:               1.0000
No. Iterations:         11.0000
```

	Coef.	Std.Err.	z	P>|z|	[0.025	0.975]
is_pw_visitors	-0.7653	0.0110	-69.6187	0.0000	-0.7869	-0.7438
is_pw_sub_store	0.4941	0.0550	8.9883	0.0000	0.3863	0.6018
cnt_pw_sub_store_7d	0.3646	0.0423	8.6115	0.0000	0.2816	0.4475
is_spent_boost	1.9988	0.0308	64.8936	0.0000	1.9384	2.0592
was_in_low_pay	-2.6124	0.0159	-164.4163	0.0000	-2.6436	-2.5813
not_gmail	-0.2289	0.0231	-9.8947	0.0000	-0.2742	-0.1835
gmail	-0.1900	0.0205	-9.2554	0.0000	-0.2302	-0.1498
bad_name	-0.5108	0.0404	-12.6546	0.0000	-0.5899	-0.4317
group_lt_20+	-1.0281	0.0110	-93.5867	0.0000	-1.0496	-1.0066
reg_source_fb	-0.7429	0.0246	-30.1858	0.0000	-0.7912	-0.6947
reg_source_form	-0.5883	0.0322	-18.2528	0.0000	-0.6515	-0.5251
reg_source_google	-0.6851	0.0245	-27.9156	0.0000	-0.7332	-0.6370
reg_source_snap	-0.8302	0.0183	-45.3318	0.0000	-0.8661	-0.7943
l_for_-	-4.7954	0.3541	-13.5436	0.0000	-5.4893	-4.1014
l_for_m0_f0_tm0_tf0_i1_o0	-1.9517	0.1941	-10.0534	0.0000	-2.3322	-1.5712
l_for_m0_f0_tm1_tf0_i0_o0	-0.5746	0.0673	-8.5314	0.0000	-0.7066	-0.4426
l_for_m0_f1_tm0_tf0_i0_o0	-0.7294	0.0121	-60.1380	0.0000	-0.7532	-0.7057
l_for_m0_f1_tm0_tf0_i1_o1	-1.2772	0.2028	-6.2993	0.0000	-1.6746	-0.8798
l_for_m0_f1_tm0_tf1_i0_o0	-0.5292	0.0216	-24.5022	0.0000	-0.5715	-0.4869
l_for_m0_f1_tm1_tf1_i0_o1	-0.6426	0.0627	-10.2558	0.0000	-0.7654	-0.5198
l_for_m1_f0_tm1_tf0_i0_o0	-0.8866	0.0665	-13.3428	0.0000	-1.0169	-0.7564
l_for_m1_f0_tm1_tf0_i0_o1	-0.7128	0.0848	-8.4037	0.0000	-0.8790	-0.5465
l_for_m1_f1_tm0_tf0_i0_o0	-1.0735	0.0362	-29.6281	0.0000	-1.1445	-1.0025
l_for_m1_f1_tm1_tf0_i1_o1	-0.4845	0.1557	-3.1129	0.0019	-0.7896	-0.1795
l_for_m1_f1_tm1_tf1_i0_o1	-1.0868	0.0797	-13.6352	0.0000	-1.2431	-0.9306
l_for_m1_f1_tm1_tf1_i1_o1	-1.0078	0.0265	-37.9739	0.0000	-1.0598	-0.9558
job_0	-0.7048	0.0352	-20.0022	0.0000	-0.7739	-0.6358
job_1	-0.4988	0.0583	-8.5573	0.0000	-0.6131	-0.3846
job_12	-0.4299	0.0348	-12.3539	0.0000	-0.4981	-0.3617
job_14	-0.5656	0.0643	-8.7917	0.0000	-0.6917	-0.4395
job_18	-0.3229	0.0261	-12.3809	0.0000	-0.3740	-0.2718
job_20	-1.1336	0.0572	-19.8213	0.0000	-1.2457	-1.0215
job_23	-0.7093	0.0174	-40.7653	0.0000	-0.7434	-0.6752
job_9	-0.8308	0.0481	-17.2614	0.0000	-0.9252	-0.7365
job_none	-0.9951	0.0152	-65.2549	0.0000	-1.0250	-0.9652
country_group_Other	-1.3057	0.0157	-83.3230	0.0000	-1.3364	-1.2750
isp_t-mobile	-0.4774	0.0186	-25.6555	0.0000	-0.5138	-0.4409
is_many_person_Lot	-0.6633	0.0105	-63.0980	0.0000	-0.6839	-0.6427