МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра системного проектування

Розробка децентралізованої платформи з використанням алгоритмів штучного інтелекту

Курсову роботу виконав:
студент групи ФеІм – 12
спеціальності 122 – Комп'ютерні науки

Брень А.А

Науковий керівник:
доц. Шувар Р.Я.
«____» _____2023 р.

Львів 2023

АНОТАЦІЯ

Курсова робота присвячена аналізу та розробці моделі класифікації машинного навчання для передбачення успіху краудфандингових компаній на децентралізованій платформі. Метою дослідження є використання алгоритмів штучного інтелекту та технологій обробки даних для покращення точності прогнозування результатів проєктів на платформі для краудфандингу.

У розділі "Огляд технологій та алгоритмів використаних в курсовій роботі" розглянуто стек технологій та алгоритми машинного навчання для класифікації, включаючи логістичну регресію, KNN, дерева рішень, Naive Bayes Gaussian та Naive Bayes Bernoulli класифікатори, а також XGBoost. Розглянуті алгоритми надають можливість проводити класифікацію та передбачати результати проєктів.

У розділі "Аналіз та обробка даних для передбачення успіху краудфандингових компаній" описано процес збору та підготовки даних, а також попередню обробку, яка включає очищення даних, відбір ознак, нормалізацію та стандартизацію. Зазначено основні параметри для аналізу даних та перетворення категоріальних змінних.

У розділі "Розробка моделі класифікації" проведено визначення початкових значень тестового набору, порівняння моделей класифікації, оцінку та результати моделі XGBoost. Представлені результати роботи моделі та їх обговорення.

Ключові слова: класифікація, машинне навчання, краудфандинг, штучний інтелект, алгоритми.

ANNOTATION

This research paper focuses on the analysis and development of a machine learning classification model for predicting the success of crowdfunding campaigns. The objective of this study is to utilize artificial intelligence algorithms and data processing technologies to improve the accuracy of project outcome prediction on crowdfunding platforms.

The "Overview of Technologies and Algorithms Used in the Research" section examines the technology stack, including logistic regression, KNN, decision trees, Naive Bayes Gaussian and Naive Bayes Bernoulli classifiers, as well as XGBoost. These algorithms enable classification and prediction of project outcomes.

The "Data Analysis and Preprocessing for Predicting the Success of Crowdfunding Campaigns" section describes the process of data collection and preparation, as well as preprocessing steps such as data cleansing, feature selection, normalization, and standardization. Key parameters for data analysis and transformation of categorical variables are identified.

In the "Development of the Classification Model" section, the initial values of the test dataset are defined, classification models are compared and the XGBoost model is evaluated and its results are presented. The findings and discussions on the model's performance are provided.

Keywords: classification, machine learning, crowdfunding, artificial intelligence, algorithms.

Зміст

Вступ *5*

Актуальність теми дослідження **5**

Мета та об'єкти дослідження **6**

Завдання дослідження **6**

Розділ 1. Огляд та обґрунтування технологій і алгоритмів використаних в курсовій роботі *8*

1.1. Стек технологій. **8**

1.2. Логістична регресія **9**

1.3. KNN **9**

1.4. Дерева рішень. Random Forest **10**

1.5. Naive Bayes Gaussian та Naive Bayes Bernoulli класифікатори **11**

1.6. XGBoost **12**

Розділ 2. Аналіз та обробка даних для передбачення успіху краудфандингових компаній 14

2.1. Збір та підготовка даних про краудфандингові проєкти **14**

2.1.1 Опис основних параметрів з датасету 14

2.2. Попередня обробка даних **16**

2.2.1. Очищення даних 16

2.2.2. Відбір ознак 16

2.2.3. Нормалізація та стандартизація 17

2.2.4. Поділ даних на тренувальний та тестовий набори 18

2.3. Процес виконання обробки даних **19**

2.3.1. Основні параметри для аналізу 20

2.3.2. Перетворення категоріальних змінних в дамі-змінні 26

Розділ 3. Розробка моделі класифікації *28*

3.1. Визначення початкових значень(тестового набору) **28**

3.2. Порівняння моделей класифікації **32**

3.3. Оцінка та результати моделі XGBoost **34**

3.4. Результати роботи моделі **38**

Висновок *40*

Список використаної літератури *42*

Додатки *43*

Посилання на GitHub репозиторій проєкту. **45**

Вступ

Актуальність теми дослідження.

На сьогоднішній день, краудфандинг є популярним механізмом фінансування для невеликих підприємств, стартапів та творчих осіб, які шукають ресурси для реалізації своїх ідей. Однак існуючі краудфандингові платформи, такі як Kickstarter, зазвичай мають централізовану структуру, де процеси витрачання коштів та прийняття рішень залежать від платформи або центральних організацій. Це може призводити до недостатньої прозорості, обмежених можливостей контролю та потенційних ризиків для учасників краудфандингу.

Децентралізована платформа на основі технології блокчейн та алгоритмів штучного інтелекту надає можливість вирішити ці проблеми. Використання блокчейн забезпечує недоступність даних та незмінність транзакцій, алгоритми штучного інтелекту дозволяють аналізувати та передбачати успішність краудфандингових проєктів, використовуючи різноманітні дані, такі як характеристики проєкту, історія успіху та відгуки користувачів. Це створює нові можливості для інвесторів приймати обґрунтовані рішення стосовно фінансування, контролювати процес витрачання коштів та забезпечує більшу прозорість та довіру в краудфандинговому середовищі.

Отже, актуальність даної теми полягає в потребі децентралізованої платформи з використанням алгоритмів штучного інтелекту, яка дозволить ефективно використовувати потенціал краудфандингу, забезпечувати прозорість та контроль для учасників, та сприяти успішності краудфандингових проєктів.

Мета та об'єкти дослідження

Основною метою даної курсової роботи є розробка моделі класифікації машинного навчання для децентралізованої платформи, яка допоможе передбачати успішність краудфандингових проєктів, що сприятиме залученню додаткового фінансування для творчих ініціатив.

Об'єкти дослідження:

1. Децентралізована платформа: Об'єктом дослідження є децентралізована платформа, яка базується на технології блокчейн та використовує смарт-контракти для забезпечення прозорості та контролю у процесі краудфандингу. Розробка такої платформи вимагає дослідження різних аспектів блокчейн технології, створення смарт-контрактів та впровадження необхідних алгоритмів для ефективного функціонування платформи.
2. Алгоритми штучного інтелекту: Дослідження охоплює алгоритми штучного інтелекту, які використовуються для аналізу та передбачення успішності краудфандингових проєктів. Об'єктом дослідження є розробка моделі класифікації машинного навчання, яка враховує різноманітні фактори, що впливають на успішність проєкту. Дослідження включає в себе вибір та підготовку відповідних даних, використання популярних алгоритмів машинного навчання, їх тренування та оцінку ефективності.

Завдання дослідження

1. Зібрати та підготувати відповідні дані: Збір необхідних даних про краудфандингові проєкти, включаючи характеристики проєктів, фінансові показники, відгуки та інші релевантні фактори, що можуть впливати на успішність проєкту.
2. Аналіз даних та виявлення кореляцій: Використовуючи методи аналізу даних, провести дослідження та виявити кореляції між різними

факторами та успішністю краудфандингових проєктів. Це може включати статистичний аналіз, використання алгоритмів машинного навчання та інші методи.

3. Розробка моделі класифікації: На основі результатів аналізу даних, розробити модель класифікації машинного навчання, яка зможе передбачати успішність краудфандингових проєктів. Вибрати та застосувати відповідні алгоритми машинного навчання, побудувати та навчити модель на підготовлених даних.
4. Оцінка ефективності моделі: Провести оцінку ефективності розробленої моделі класифікації, використовуючи відповідні метрики та тестові дані. Визначити точність, чутливість, специфічність та інші показники ефективності моделі.
5. Сформулювати висновки на основі результатів дослідження та оцінки ефективності моделі. Запропонувати рекомендації щодо використання розробленої моделі для передбачення успішності краудфандингових компаній та можливих напрямків подальших досліджень.

Розділ 1. Огляд технологій та алгоритмів використаних в курсовій роботі

1.1. Стек технологій.

- Python – високорівнева мова програмування з багатьма бібліотеками та фреймворками для аналізу даних та машинного навчання, що дозволить ефективно реалізувати та оцінити модель.
- Бібліотека Scikit-learn – є однією з найпопулярніших бібліотек для машинного навчання в середовищі Python. Вона містить реалізацію багатьох алгоритмів класифікації, включаючи методи рішень, ансамблі, нейронні мережі та інші. Scikit-learn надасть зручний інтерфейс для побудови моделі класифікації та оцінки її ефективності.
- Pandas та NumPy – це бібліотеки для обробки та аналізу даних в Python. Pandas надає структури даних, такі як DataFrame, для зручного маніпулювання та фільтрації даних. NumPy забезпечує підтримку для роботи з багатовимірними масивами та матрицями. Вони будуть використовуватись для підготовки та обробки даних перед подальшою обробкою та моделюванням.
- Jupyter Notebook – це інтерактивне середовище для виконання коду Python та створення документів, що поєднують код, тексти та візуалізації. Воно дозволяє ітеративно виконувати код, відображати результати та додавати пояснювальні коментарі. Jupyter Notebook буде використовуватись для реалізації та документування кроків дослідження.
- Matplotlib та Seaborn – це бібліотеки для візуалізації даних в Python. Вони надають широкі можливості для побудови графіків, діаграм та інших візуалізаційних засобів. Вони будуть використовуватись для візуалізації результатів дослідження та аналізу.

1.2. Логістична регресія

Логістична регресія є одним з найпоширеніших алгоритмів машинного навчання, які використовуються для бінарної класифікації. Вона базується на логістичній функції, яка моделює ймовірність належності об'єкта до певного класу.

Принцип роботи логістичної регресії полягає в побудові лінійної регресійної моделі, яка прогнозує логістичну функцію від залежних змінних. Логістична функція набуває значень від 0 до 1, що можна інтерпретувати як ймовірність належності до певного класу. Для класифікації використовується порогове значення, яке визначає, до якого класу належить об'єкт.

Логістична регресія є ефективним алгоритмом, який добре справляється з лінійно роздільними даними. Вона також може бути розширена до нелінійних залежностей за допомогою методу поліноміальних ознак або використанням інших трансформацій.

У контексті дослідження, логістична регресія може бути використана для класифікації краудфандингових проєктів на основі різноманітних ознак, таких як тип проєкту, сума зібраних коштів, тривалість компанії та інші. Вона може допомогти передбачити успішність проєкту шляхом встановлення зв'язку між характеристиками та його вірогідністю залучення достатньої кількості фінансування.

1.3. KNN

K-Nearest Neighbors (KNN) (k-найближчих сусідів) є одним з найпростіших та найпопулярніших алгоритмів машинного навчання для класифікації та регресії. Він використовується для вирішення задач, де об'єкти класифікуються на основі близькості до інших об'єктів в наборі даних.

Принцип роботи KNN полягає у визначенні класу нового об'єкта на основі ближніх до нього об'єктів з набору даних. Алгоритм знаходить k найближчих сусідів до нового об'єкта за допомогою певної метрики відстані, такої як Евклідова відстань або Манхетенська відстань. Потім він визначає клас нового об'єкта, шляхом голосування більшості з його k найближчих сусідів.

KNN є непараметричним алгоритмом, оскільки він не залежить від попередніх припущень щодо розподілу даних. Він простий у реалізації та використанні і може бути ефективним для класифікації невеликих наборів даних. Однак, він може вимагати значних обчислювальних витрат для великих наборів даних, оскільки потребує обчислення відстаней до всіх об'єктів в наборі.

У контексті дослідження, KNN може бути використаний для класифікації краудфандингових проєктів, використовуючи різноманітні характеристики проєктів, такі як категорія проєкту, фінансові дані, кількість спонсорів тощо. За допомогою KNN можна передбачити успіх або невдачу проєкту на основі схожості з іншими вже відомими проєктами з набору даних.

1.4. Дерева рішень. Random Forest

Дерева рішень є популярними алгоритмами машинного навчання, які використовуються для класифікації та регресії. Вони представляють собою ієрархічну структуру, де кожен вузол відповідає рішенню на основі певної ознаки. Дерева рішень добре інтерпретовані, легкі у розумінні та можуть використовуватись для передбачення класу об'єктів на основі їх характеристик.

Однак, дерева рішень мають певні обмеження, такі як схильність до перенавчання (overfitting), особливо коли вони мають глибоку структуру. Це може призвести до поганої узагальнюючої здатності моделі на нових

даних. Тому, для покращення ефективності та зменшення перенавантаження, можна використовувати ансамблеві методи, такі як Random Forest.

Random Forest (випадковий ліс) є ансамблевим методом машинного навчання для класифікації, який поєднує кілька дерев рішень, щоб отримати більш точні прогнози. Він працює шляхом створення випадкових підвибірок даних та випадкового вибору ознак для побудови кожного дерева. Кожне дерево рішень в Random Forest незалежно приймає рішення, а потім прогнози об'єднуються за допомогою голосування або середнього значення для класифікації або регресії відповідно.

Random Forest має кілька переваг. Він зменшує ефект перенавчання, оскільки працює зі зменшеним набором ознак та даних. Він також може виявляти важливі ознаки та проводити оцінку їх впливу на модель. Крім того, Random Forest може працювати з великими наборами даних і добре справлятись з пропущеними значеннями та викидами даних.

У контексті дослідження, дерева рішень та Random Forest можуть використовуватись для розробки моделей класифікації, які будуть прогнозувати успішність краудфандингових компаній. Ці алгоритми дозволять використовувати різні характеристики проєктів та побудувати модель, яка здатна передбачати ймовірність успіху проєкту на основі цих ознак.

1.5. Naive Bayes Gaussian та Naive Bayes Bernoulli класифікатори

Naive Bayes Gaussian та Naive Bayes Bernoulli(Наївний Баєс-Гаус та Наївний Баєс-Бернулі) класифікатори є двома варіантами Наївного баєсового алгоритму, які використовуються для класифікації в машинному навчанні. Наївний баєсовий алгоритм є ймовірнісним алгоритмом класифікації, який ґрунтується на принципі Баєсової теореми та припущенні про незалежність ознак. Основна ідея полягає в тому, що

вважається, що кожна ознака вносить свій внесок у вірогідність належності об'єкта до певного класу.

Naive Bayes Gaussian використовується, коли ознаки мають нормальний (гаусівський) розподіл. Цей варіант підходить для неперервних числових ознак. Він моделює кожен клас як гаусівський розподіл і використовує його параметри (середнє значення та стандартне відхилення) для обчислення ймовірності належності об'єкта до класу.

Naive Bayes Bernoulli використовується, коли ознаки є бінарними (0 або 1), наприклад, показують наявність або відсутність певної характеристики. Цей варіант підходить для використання з бінарними даними. Він моделює кожен клас як бернуллієвий розподіл та використовує його параметри (ймовірність успіху та ймовірність невдачі) для обчислення ймовірності належності об'єкта до класу.

Ці два варіанти Naive Bayes є простими та швидкими алгоритмами класифікації. Вони добре працюють у випадках, коли припущення про незалежність ознак близьке до реальності. Застосування Naive Bayes Gaussian або Naive Bayes Bernoulli залежить від типу даних, з якими ми маємо справу, і характеристик самої задачі класифікації.

1.6. XGBoost

XGBoost (Extreme Gradient Boosting) є алгоритмом машинного навчання, який використовує градієнтний бустинг для покращення прогнозів моделі. Він є реалізацією градієнтного бустингу для різних задач, включаючи класифікацію, регресію та ранжування. XGBoost відзначається своєю високою швидкодією та ефективністю. Він може працювати з великими наборами даних та використовувати паралельну обробку для прискорення обчислень. Алгоритм використовує дерев'яний ансамбль моделей, де кожне дерево побудоване на основі помилок попередніх дерев та максимізує задану функцію втрати. XGBoost має кілька важливих

особливостей, таких як регуляція для запобігання перенавчанню, автоматичний підбір оптимальної кількості дерев та гнучкість в налаштуванні параметрів моделі. Він також підтримує вагові набори даних, що дає можливість надати більшу вагу певним об'єктам або класам. У контексті дослідження, XGBoost може бути використаний для побудови моделей класифікації, які будуть передбачати успішність краудфандингових компаній. Він може працювати з багатьма характеристиками проєктів та автоматично вивчати їх вплив на цільову змінну. Цей алгоритм дозволить отримати точні та надійні прогнози щодо успішності проєктів та допоможе зробити кращі рішення у краудфандинговій сфері.

Розділ 2. Аналіз та обробка даних для передбачення успіху краудфандингових компаній

2.1. Збір та підготовка даних про краудфандингові проєкти

Kickstarter — це популярна платформа краудфандингу, яка допомогла багатьом підприємцям та просто мрійникам втілити в життя свої інноваційні ідеї. Однак не всі проєкти Kickstarter є успішними, і розуміння факторів, які сприяють успіху чи провалу, може бути цінним як для творців, так і для інвесторів.

Для дослідження даних про краудфандингові компанії було використано датасети з вебсайту Web Robots. У цьому наборі даних зібрано інформацію про велику кількість проєктів Kickstarter і про те, чи врешті-решт вони досягли своїх цілей щодо фінансування. Цей набір даних включає широкий спектр типів проєктів, зокрема технологічні стартапи, творчі ініціативи в галузі мистецтва та багато іншого.

Аналізуючи цей набір даних, дослідники та аналітики можуть отримати уявлення про характеристики успішних і неуспішних проєктів Kickstarter, такі як цілі фінансування, категорії і джерела фінансування.

Загалом, цей набір даних дає комплексний погляд на екосистему Kickstarter і є цінним ресурсом для розуміння динаміки краудфандингу та факторів, які сприяють успіху чи невдачі проєкту.

Вхідний датасет складається з 112 наборів даних у форматі csv за період з 2022 по травень 2023 року. Об’єднані дані місять в собі 591566 рядків та 39 колонок.

2.1.1 Опис основних параметрів з датасету

- ***backers_count***: Кількість осіб, які підтримали проєкт із фінансової точки зору.
- ***blurb***: Короткий опис проекту, який надається на Kickstarter.
- ***category***: Категорія, до якої належить проект.

- ***converted_pledged_amount***: Сума, зібрана проектом в конвертованій валюті.
- ***country*:** Країна, з якої походить проект.
- ***created_at****:* Дата та час створення проекту.
- ***currency:*** Валюта, в якій запрошується фінансування.
- ***deadline***: Дата та час закінчення збору коштів для проекту.
- ***fx_rate***: Обмінний курс для конвертації валюти проекту.
- ***goal:*** Цільова сума, яку проєкт хоче зібрати.
- ***id:*** Унікальний ідентифікатор проекту.
- ***launched_at:*** Дата та час запуску проекту.
- ***location:*** Місцезнаходження проекту.
- ***name:*** Назва проекту.
- ***pledged:*** Сума, зібрана проектом в оригінальній валюті.
- ***slug***: Характерний ідентифікатор проекту для URL.
- ***spotlight:*** Показує, чи проєкт був відзначений на Kickstarter як особливий.
- ***staff_pick:*** Показує, чи був проєкт обраний командою Kickstarter.
- ***state:*** Поточний статус проекту (успішний, невдалий, активний тощо).
- ***state_changed_at***: Дата та час останньої зміни статусу проекту.
- ***static_usd_rate:*** Фіксований обмінний курс для конвертації валюти проекту в долари США.
- ***usd_pledged:*** Сума, зібрана проектом в доларах США.
- ***usd_type***: Тип обробки валюти (залежить від джерела даних).
- ***main_category***: Основна категорія, до якої належить проект.
- ***sub_category***: Підкатегорія, до якої належить проект.
- ***goal_usd***: Цільова сума в доларах США.

2.2. Попередня обробка даних

2.2.1. Очищення даних

Процес очищення даних є необхідним перед аналізом та розробкою моделі класифікації для передбачення успіху краудфандингових компаній. Очищення даних включає в себе виявлення та виправлення помилок, вирішення пропущених значень та видалення непотрібних атрибутів, що можуть негативно впливати на якість та достовірність аналізу.

Під час процесу виявлення та виправлення помилок, дані аналізуються з метою виявлення можливих помилок або неточностей, таких як помилки введення, вимірювання або обробки даних. Знайдені помилки будуть виправлені шляхом корекції або заміни неправильних значень на правильні, що допоможе забезпечити точність та достовірність даних.

Пропущені значення, які можуть бути присутніми в даних, вирішуються за допомогою різних методів, таких як заповнення середніми значеннями, інтерполяція або виключення відповідних записів з даних. Це дозволяє зберегти повноту даних та уникнути вплив пропущених значень на аналіз та модель.

Непотрібні атрибути, які не мають значення для аналізу та розробки моделі класифікації, видаляються. Це можуть бути атрибути з багатьма пропущеними значеннями, атрибути без варіації або атрибути, які не мають впливу на цільову змінну. Видалення непотрібних атрибутів спрощує аналіз даних та зменшує обсяг обробки, покращуючи ефективність розробки моделі класифікації.

2.2.2. Відбір ознак

Відбір ознак є важливим етапом у розробці моделі класифікації для передбачення успіху краудфандингових компаній. Відбір ознак включає в себе ідентифікацію найбільш інформативних та релевантних ознак з наявних даних, що допоможе покращити якість та ефективність моделі.

Під час відбору ознак використовуються різні методи та алгоритми, такі як аналіз кореляції, алгоритми зменшення розмірності, важливість ознак на основі моделей машинного навчання тощо. Ці методи дозволяють визначити ті ознаки, які мають найбільший вплив на цільову змінну та найкраще відображають залежності в даних.

Аналіз кореляції дозволяє встановити статистичну залежність між ознаками та цільовою змінною. Ознаки, які мають високу кореляцію з цільовою змінною, є більш інформативними та мають великий вплив на результати моделі. Такі ознаки включаються до моделі для подальшого аналізу.

Алгоритми зменшення розмірності, такі як Principal Component Analysis (PCA) або L1-регуляризація, дозволяють зменшити кількість ознак у даних, зберігаючи при цьому найбільшу частину інформації. Це допомагає зменшити перевантаження моделі та уникнути проблеми перенавчання.

Важливість ознак на основі моделей машинного навчання визначається шляхом оцінки вагомості кожної ознаки у процесі навчання моделі. Ознаки, які мають високий коефіцієнт важливості, вважаються значущими для моделі та використовуються у подальшому аналізі та розробці.

Після відбору ознак формується новий набір даних, що складається лише з найінформативніших ознак. Це дозволяє зосередитись на найважливіших аспектах дослідження та покращує ефективність розробки моделі класифікації для передбачення успіху краудфандингових компаній.

2.2.3. Нормалізація та стандартизація

При нормалізації та стандартизації проводиться попередня обробка даних з метою приведення їх до однакового масштабу та розподілу. Цей

процес важливий для забезпечення стабільності та надійності роботи моделі класифікації.

Нормалізація є процесом приведення числових значень ознак до відповідного діапазону, зазвичай від 0 до 1. Це здійснюється шляхом перетворення значень кожної ознаки відповідно до їх максимального та мінімального значень у наборі даних. Нормалізація дозволяє уникнути проблеми з різними шкалами значень та покращити стабільність роботи моделі.

Стандартизація є процесом приведення значень ознак до стандартного нормального розподілу з середнім значенням 0 та стандартним відхиленням 1. Це здійснюється шляхом віднімання середнього значення ознаки та поділу на стандартне відхилення усіх значень в наборі даних. Стандартизація дозволяє забезпечити нульове середнє значення та однакове масштабування ознак, сприяючи збереженню важливої інформації та зниженню впливу викидів.

Використання нормалізації та стандартизації допомагає забезпечити однакові умови для обробки та аналізу даних у моделі класифікації. Це покращує стабільність та швидкодію моделі, а також забезпечує правильну інтерпретацію ваги ознак та їх впливу на результат класифікації.

2.2.4. Поділ даних на тренувальний та тестовий набори

Невід’ємним етапом є поділ набору даних на дві відокремлені частини: тренувальний набір і тестовий набір. Цей процес є важливим кроком у підготовці даних для моделі класифікації і дозволяє оцінити ефективність моделі на незалежних даних.

Тренувальний набір використовується для навчання моделі, щоб вона могла "вивчити" зв'язки та закономірності в даних. Цей набір містить вхідні ознаки та відповідні мітки класів, які використовуються для навчання моделі на основі різних алгоритмів машинного навчання.

Тестовий набір використовується для оцінки ефективності моделі, яка була навчена на тренувальному наборі. Цей набір також містить вхідні ознаки, але без відповідних міток класів. Модель використовується для прогнозування класів для зразків тестового набору, і її результати порівнюються зі справжніми мітками для оцінки точності та ефективності моделі.

Розбиття даних на тренувальний та тестовий набори є важливим для оцінки здатності моделі класифікації до узагальнення на нові, раніше невідомі дані. Воно допомагає виявити можливі перевантаження або недостатню здатність моделі до узагальнення, а також дозволяє налаштувати параметри моделі для досягнення найкращої продуктивності.

2.3. Процес виконання обробки даних

У даному розділі описується процес аналізу даних з використанням датасету про платформу Kickstarter за період 2022-2023 років. Всього датасет містить понад 591 тисячу записів, що представляє значний обсяг інформації для аналізу та моделювання.

(591566, 39)

Out[29]:

	backers_count	blurb	category	converted_pledged_amount	country	country_displayable_name	created_at	
0	8	A new, for-profit community theatre group base...	{"id":285,"name":"Plays","analytics_name":"Pla...	291.0	US	the United States	1541529891	
1	1	A 25 minute play with 5 scenes, including an o...	{"id":285,"name":"Plays","analytics_name":"Pla...	1.0	US	the United States	1536253805	{"id":
2	5	Help us recovering the Opera tradition in one ...	{"id":285,"name":"Plays","analytics_name":"Pla...	695.0	IT	Italy	1541602785	
3	6	Using the Power of Visual and Performing Arts ...	{"id":285,"name":"Plays","analytics_name":"Pla...	367.0	US	the United States	1540763981	{"id"
4	2	Ché Walker's AMERICAN CARNAGE, a staged readin...	{"id":285,"name":"Plays","analytics_name":"Pla...	2.0	US	the United States	1532828061	

5 rows × 39 columns

Рис. 2.3 – Датасет перед обробкою даних

2.3.1. Основні параметри для аналізу

У цьому підрозділі описуються колонки, які залишились після обробки даних та які були обрані для аналізу.

Основні параметри для аналізу та формування моделі включають:

- Сума зібраних коштів (pledges): Цей параметр вказує на загальну суму коштів, які були зібрані для проекту. Хоча ця інформація не була використана через її футуристичний характер, вона може бути корисною для додаткового аналізу та порівняння результатів.
- Цільова сума проекту в доларах США: Цей параметр вказує на суму коштів, яку проєкт має намір зібрати. Включення цього параметра в аналіз є важливим, оскільки досягнення цільової суми може бути одним з ключових факторів успішності проекту.
- Тривалість компанії (від дати запуску до кінцевого терміну): Цей параметр вказує на тривалість компанії, тобто період часу, протягом якого проєкт збирав кошти. Включення цього параметра дозволяє врахувати часовий аспект та його вплив на успішність проекту.
- Тривалість підготовки (від дати створення до дати запуску): Цей параметр вказує на період часу, який був витрачений на підготовку проекту до його запуску. Включення цього параметра може допомогти оцінити важливість передпроектної підготовки та її вплив на успішність проекту.
- Місце розташування компанії (в США або за її межами): Цей параметр вказує на місце проведення компанії – чи вона проводиться в США чи в інших країнах. Включення цього

параметра дозволяє врахувати вплив географічного розташування на успішність проекту.

- Довжина опису компанії: Цей параметр вказує на кількість символів у текстовому описі компанії. Включення цього параметра дозволяє оцінити вплив обсягу інформації, наданої проектом, на його успішність.
- Категорія компанії (представлена у вигляді даммі-змінних): Цей параметр вказує на категорію, до якої належить проект. Включення цього параметра дозволяє врахувати вплив категорії на успішність проекту.
- Отримання особливої підтримки від Kickstarter (так/ні) (дамі-змінна): Цей параметр вказує, чи отримав проєкт особливе підтримання від команди Kickstarter. Хоча це значення відноситься до майбутнього, його було залишено в аналізі, щоб користувачі могли перевірити потенційну компанію, а творці могли оцінити свій можливий результат з чи без підтримки.

Вибір цих параметрів був зумовлений їх вагомістю та потенційним впливом на успішність краудфандингових проектів на платформі Kickstarter. Вони надають необхідну інформацію для аналізу та розробки моделі класифікації для передбачення успіху проектів.

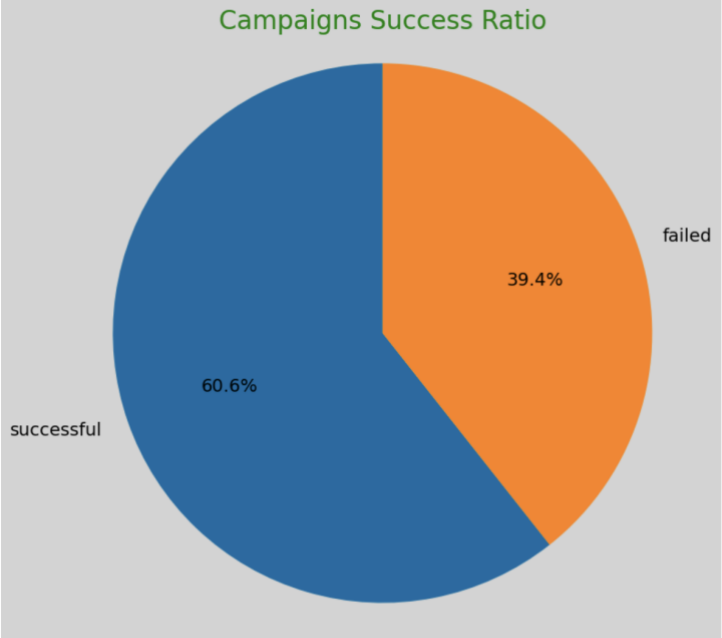
Out[29]:

	id	goal_usd	country	created_at	deadline	launched_at	state_changed_at	spotlight	staff_pick	main_category	sub_category	blurb	state
	1883729963	1000.00000	US	2018-11-06 18:44:51	2018-12-27 18:00:43	2018-11-27 18:00:43	2018-12-27 18:00:44	0	0	Theater	Plays	A new, for-profit community theatre group base...	failed
	110053898	10000.00000	US	2018-09-06 17:10:05	2019-01-25 19:38:00	2018-11-27 15:22:37	2019-01-25 19:45:22	0	0	Theater	Plays	A 25 minute play with 5 scenes, including an o...	failed
	102406680	55451.96832	IT	2018-11-07 14:59:45	2018-12-13 13:27:12	2018-11-13 13:27:12	2018-12-13 13:27:12	0	0	Theater	Plays	Help us recovering the Opera tradition in one ...	failed
	1821399542	25000.00000	US	2018-10-28 21:59:41	2018-12-01 01:00:00	2018-10-30 16:00:55	2018-12-01 01:00:00	0	0	Theater	Plays	Using the Power of Visual and Performing Arts ...	failed
	825863534	1500.00000	US	2018-07-29 01:34:21	2018-11-08 04:48:43	2018-10-09 03:48:43	2018-11-08 04:48:44	0	0	Theater	Plays	Ché Walker's AMERICAN CARNAGE, a staged readin...	failed

In [30]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 197197 entries, 0 to 197196
Data columns (total 13 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   id                197197 non-null  int64
 1   goal_usd          197197 non-null  float64
 2   country           197197 non-null  object
 3   created_at        197197 non-null  object
 4   deadline          197197 non-null  object
 5   launched_at       197197 non-null  object
 6   state_changed_at  197197 non-null  object
 7   spotlight         197197 non-null  int64
 8   staff_pick        197197 non-null  int64
 9   main_category     197197 non-null  object
 10  sub_category      197197 non-null  object
 11  blurb             197194 non-null  object
 12  state             197197 non-null  object
dtypes: float64(1), int64(3), object(9)
memory usage: 19.6+ MB
```

Рис. 2.3.1 – Датасет після очищення даних




```
successful    119577
failed         77620
Name: state, dtype: int64
successful    0.606383
failed        0.393617
Name: state, dtype: float64
```

Рис. 2.3.2. – Відсоткове співвідношення успішних компаній до невдалих

У наборі даних є злегка незбалансована класифікація для змінної "success". Приблизно 61% спостережень мають класифікацію "1", тоді як 39% - "0". Це означає, що кількість успішних кампаній трохи перевищує кількість невдалих.

Збалансованість класів є важливим аспектом при побудові моделей класифікації. У даному випадку, незначна незбалансованість класів може не мати значного впливу на аналіз.

Для подальшого аналізу та моделювання даних можна використовувати змінну "success" як цільову змінну для класифікації та використовувати решту ознак для прогнозування успіху кампаній на платформі.

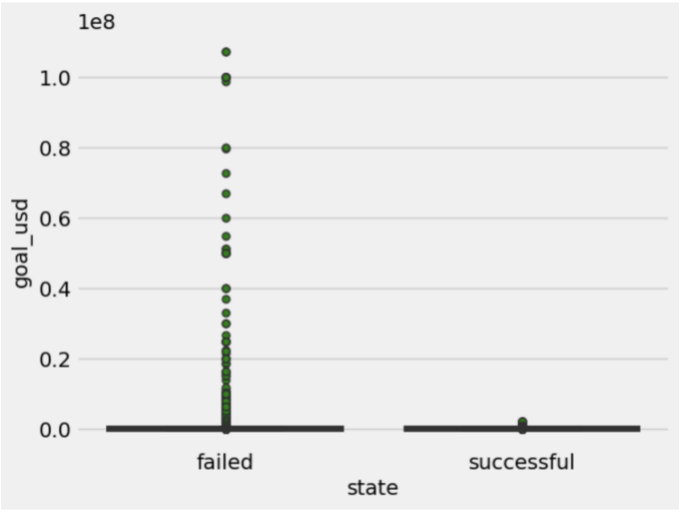


Рис. 2.3.3. – Розподіл цільових сум залежно від статусу(успішний/невдалий) компанії

На основі наданих результатів можна зробити наступні висновки щодо параметра "Goal in USD":

- Середнє значення (Mean value): 34750.45. Це вказує на середнє значення цільової суми, яку ставлять перед собою кампанії на Kickstarter.

- Медіана (Median value): 4575.0. Медіанне значення є менш чутливим до викидів і може відображати більш типову цільову суму.
- Мінімальне значення (Minimum value): 0.01. Це найменша цільова сума, яку можна встановити для кампанії на Kickstarter.
- Максимальне значення (Maximum value): 107294986.0. Це найбільша цільова сума, яку було встановлено в датасеті.
- Стандартне відхилення (Standard deviation): 942782.65. Це вказує на розкид цільових сум кампаній, що може відображати різноманітність проектів на Kickstarter.
- Відсоток викидів (Outlier percentage): 11.61%. Це вказує на відсоток кампаній, які мають цільову суму, що виходить за межі визначених вище порогів (нижнього та верхнього), що може свідчити про особливі або виняткові проекти.

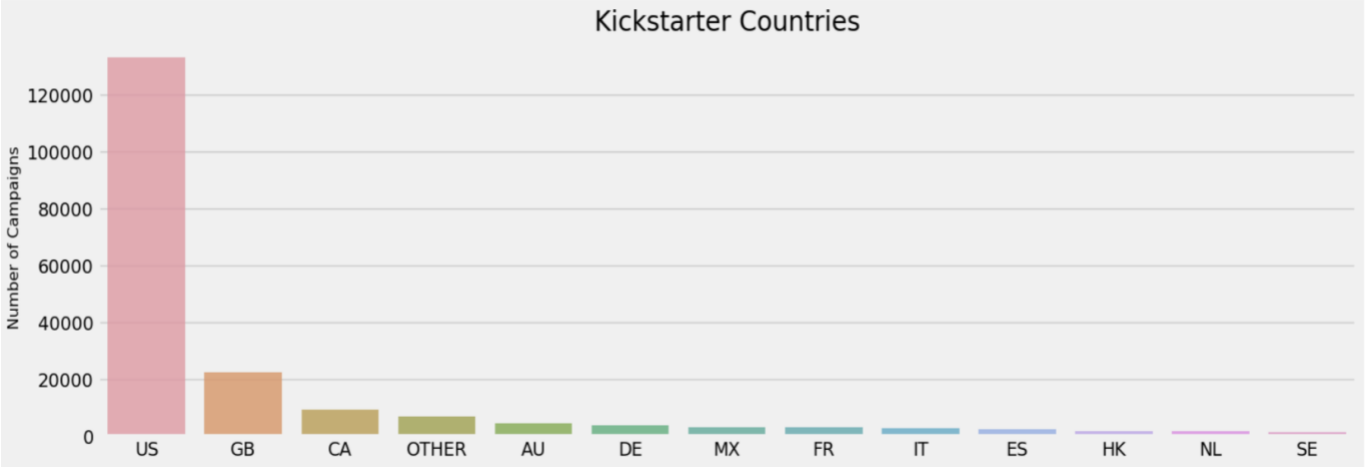


Рис. 2.3.4. – Розподіл компаній за країнами на платформі Kickstarter

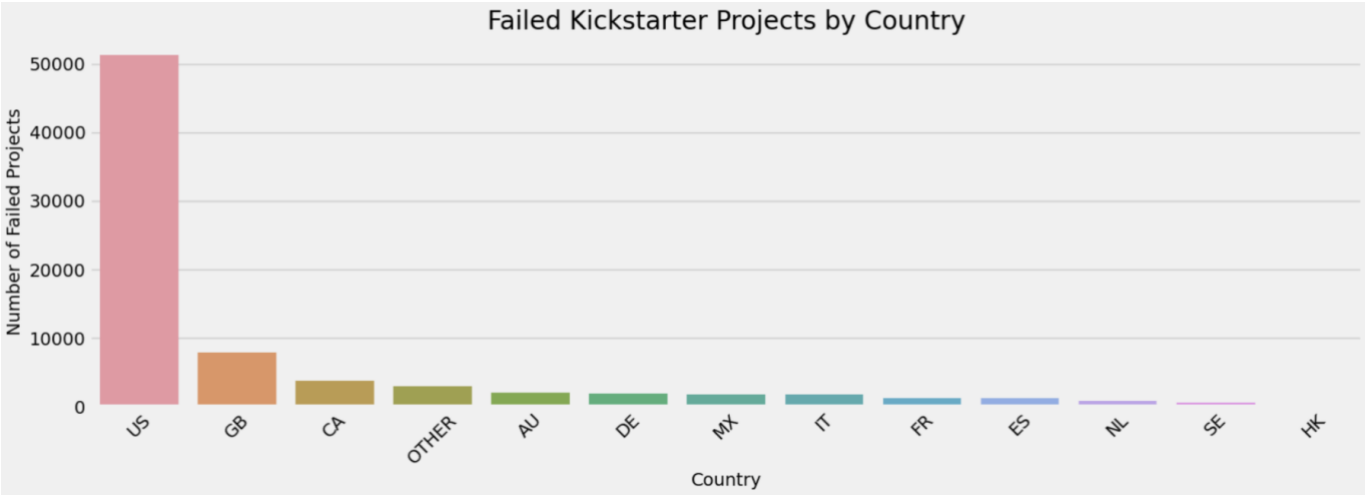


Рис. 2.3.5. – Розподіл неуспішних проектів на платформі Kickstarter за країнами

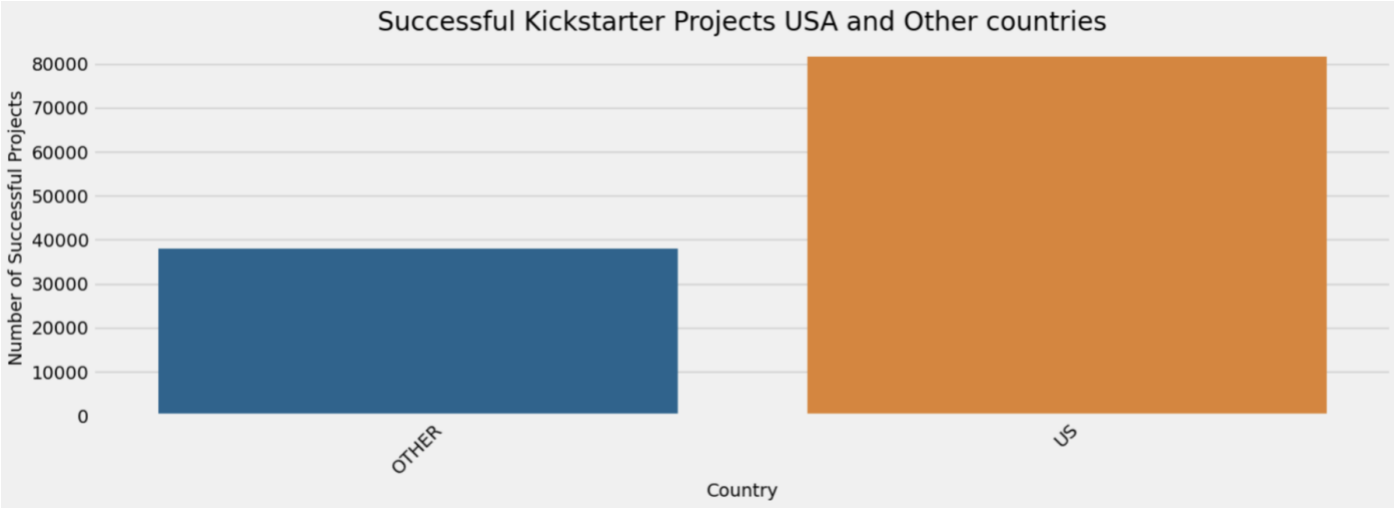


Рис. 2.3.6. – Розподіл успішних проектів на платформі Kickstarter за країнами (США та інші)

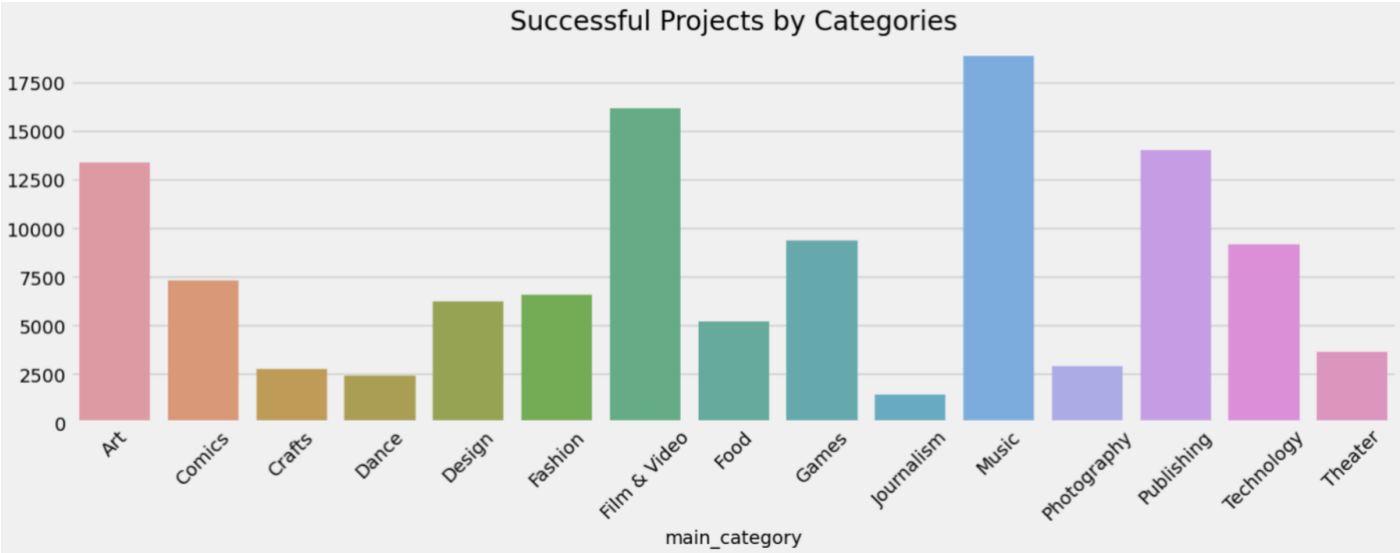


Рис.2.3.7. – Графік успішних проектів за категоріями

На основі показників можна зробити наступні висновки щодо успіху компаній в залежності від вибору категорії:

- Деякі категорії мають значно більшу кількість успішних проектів, наприклад, Music, Film & Video та Publishing. Це може вказувати на те, що компанії, що працюють у цих галузях, мають більшу ймовірність досягти успіху на Kickstarter.
- Деякі категорії можуть бути менш успішними, якщо порівнювати їх з іншими, наприклад, Crafts, Dance, Photography та Theater. Це може свідчити про меншу популярність проектів у цих галузях або про інші фактори, які впливають на їх успішність.
- Категорії, які мають помірну кількість успішних проектів, такі як Design, Fashion, Games та Technology, можуть бути привабливими для компаній, які шукають успіх на Kickstarter.

2.3.2. Перетворення категоріальних змінних в дамі-змінні

Дамі-змінні (dummy variables) – це бінарні змінні, які використовуються для представлення категоріальних даних у числовому вигляді. використовуються для представлення категоріальних параметрів,

таких як категорії проектів або наявність особливого підтримання від Kickstarter. Замість використання текстових або категоріальних значень, ми перетворюємо ці параметри на дамі-змінні, що дозволяє нам використовувати їх у моделі машинного навчання, яка працює з числовими даними. Ми перетворюємо main_category в дамі-змінну.

	success	goal_usd	country_US	cam_duration	prep_duration	description_length	staff_pick_True	main_category_Comics	main_category_Crafts	main_ca
0	0	1000.00000	1	30	20	25	0	0	0	
1	0	10000.00000	1	59	81	20	0	0	0	
2	0	55451.96832	0	30	5	15	0	0	0	
3	0	25000.00000	1	31	1	18	0	0	0	
4	0	1500.00000	1	30	72	19	0	0	0	

5 rows × 21 columns

Рис.2.3.8. – Фінальний набір даних після обробки та перетворень

Розділ 3. Розробка моделі класифікації

3.1. Визначення початкових значень(тестового набору)

Після тренування моделі класифікації на тренувальному наборі даних, потрібно перевірити її ефективність на нових, раніше невикористаних даних. Ці нові дані називаються тестовим набором даних і є ключовою складовою процесу оцінки моделі.

Визначення початкових значень для тестового набору даних включає підготовку даних, їх очищення, масштабування або нормалізацію, а також застосування інших перетворень, що були застосовані до тренувального набору даних.

Визначення початкових значень на тестовому наборі даних має кілька цілей. По-перше, воно допомагає забезпечити послідовність та об'єктивність оцінки моделі, оскільки ми використовуємо одні й ті ж методи підготовки даних для обох наборів. По-друге, воно дозволяє порівняти результати різних алгоритмів машинного навчання на одних і тих самих початкових умовах та даних.

В процесі визначення початкових значень тестового набору даних ми можемо застосовувати ті самі методи обробки даних, які були застосовані до тренувального набору, такі як видалення відсутніх значень, масштабування або нормалізацію функцій, а також застосування інших перетворень, необхідних для відповідності формату тестового набору даних.

Визначення початкових значень тестового набору даних допомагає забезпечити точну та надійну оцінку ефективності моделі на нових даних і є важливою складовою аналізу та моделювання даних.

Використаємо матрицю плутанини(confusion matrix) для для оцінки ефективності моделі класифікації на наборі даних. Вона показує кількість правильно та неправильно класифікованих екземплярів для кожного класу.

Матриця плутанини складається з чотирьох основних значень:

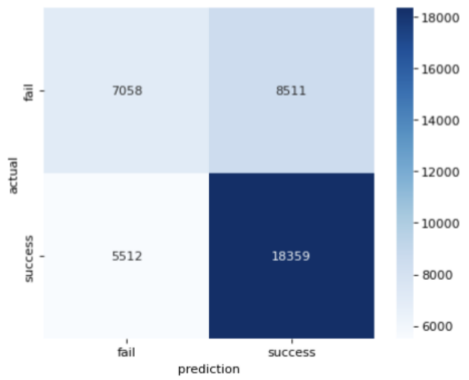
1. True Positive (TP): Кількість випадків, коли модель правильно визначила позитивний клас (успішний проект).
2. True Negative (TN): Кількість випадків, коли модель правильно визначила негативний клас (невдалий проект).
3. False Positive (FP): Кількість випадків, коли модель неправильно визначила позитивний клас (невдалий проєкт визнаний успішним).
4. False Negative (FN): Кількість випадків, коли модель неправильно визначила негативний клас (успішний проєкт визнаний невдалим).

Матриця плутанини дозволяє оцінити різні метрики ефективності моделі, такі як точність (precision), повнота (recall), F1-показник (F1-score) та інші. Ці метрики дають інформацію про здатність моделі правильно класифікувати дані для кожного класу.

Враховуючи матрицю плутанини, можна зробити аналіз результатів моделі та прийняти рішення щодо подальшого вдосконалення або використання інших підходів у класифікації даних.

```
Logistic Regression validation metrics with subset 1:
 Accuracy: 0.6627
 Precision: 0.6631
 Recall: 0.8997
 F1: 0.7635
 ROC: 0.6516
```

```
In [28]: make_confusion_matrix(lr)
```




```
Logistic Regression validation metrics with subset 2:
 Accuracy: 0.6621
 Precision: 0.6612
 Recall: 0.9059
 F1: 0.7644
 ROC: 0.6654
```

```
In [34]: make_confusion_matrix(lr2)
```

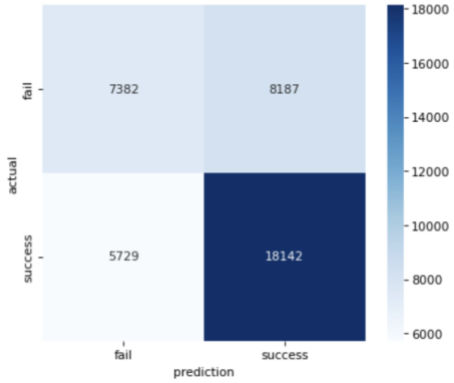


Рис. 3.1. – Графічне зображення матриці плутанини для наборів даних з певними параметрами

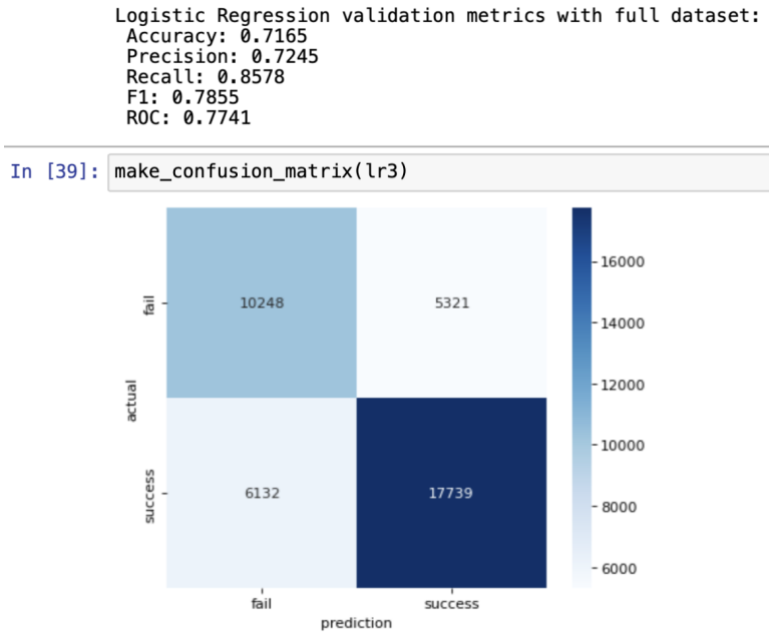


Рис. 3.2. – Графічне зображення матриці плутанини для набору даних з усіма параметрами

Модель логістичної регресії була використана для класифікації проектів на успішні та невдалі.

Підготовка даних: Перед побудовою моделі дані були розділені на набір ознак (X) та цільову змінну (y). Для нормалізації ознак був використаний стандартизатор (StandardScaler).

Результати моделі: Для моделі з параметром C=1000 (без регуляризації) отримано наступні метрики:

Точність (Precision): Для класу "0" - 0.70, для класу "1" - 0.72. Це вказує на те, що модель здатна вірно ідентифікувати успішні та невдалі проекти.

Повнота (Recall): Для класу "0" - 0.50, для класу "1" - 0.86. Це означає, що модель може ефективно виявляти більшість успішних проектів, але має недостатню здатність виявляти невдачі.

F1-показник (F1-Score): Для класу "0" - 0.58, для класу "1" - 0.79. Це комбінована метрика, яка враховує як точність, так і повноту. Високе значення F1-показника свідчить про хорошу збалансованість між точністю та повнотою моделі.

AUC-ROC(Area Under the Receiver Operating Characteristic Curve): Значення площі під кривою ROC становить 0.7741. Це показник ефективності моделі у відношенні до класифікації успішних і невдалих проектів.

Загалом, модель логістичної регресії показує добрі результати на повному наборі даних. Вона досягає точності 0.7165 і демонструє прийнятну здатність передбачати успішні та невдалі проекти. Однак, для покращення виявлення невдач можуть знадобитися додаткові оптимізації або використання інших моделей.

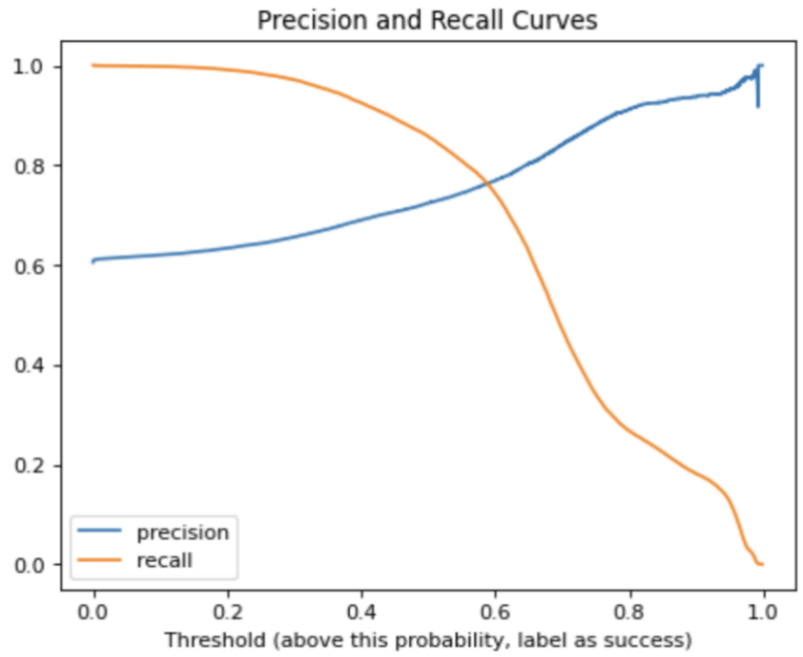


Рис. 3.1.1. – Графік компромісу між точністю та повною

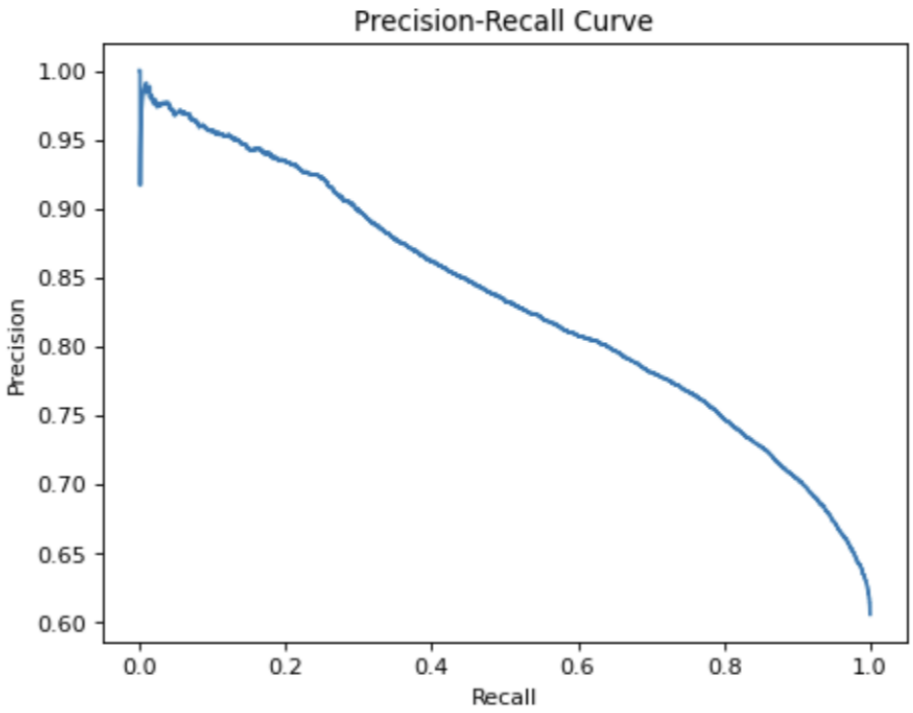


Рис. 3.1.2 – Крива точність-повнота

3.2. Порівняння моделей класифікації

У даному розділі проведено порівняння різних моделей класифікації для передбачення успіху краудфандингових проєктів на децентралізованій платформі. Для оцінки ефективності моделей були використані наступні метрики: accuracy, precision, recall, F1-score і ROC-AUC.

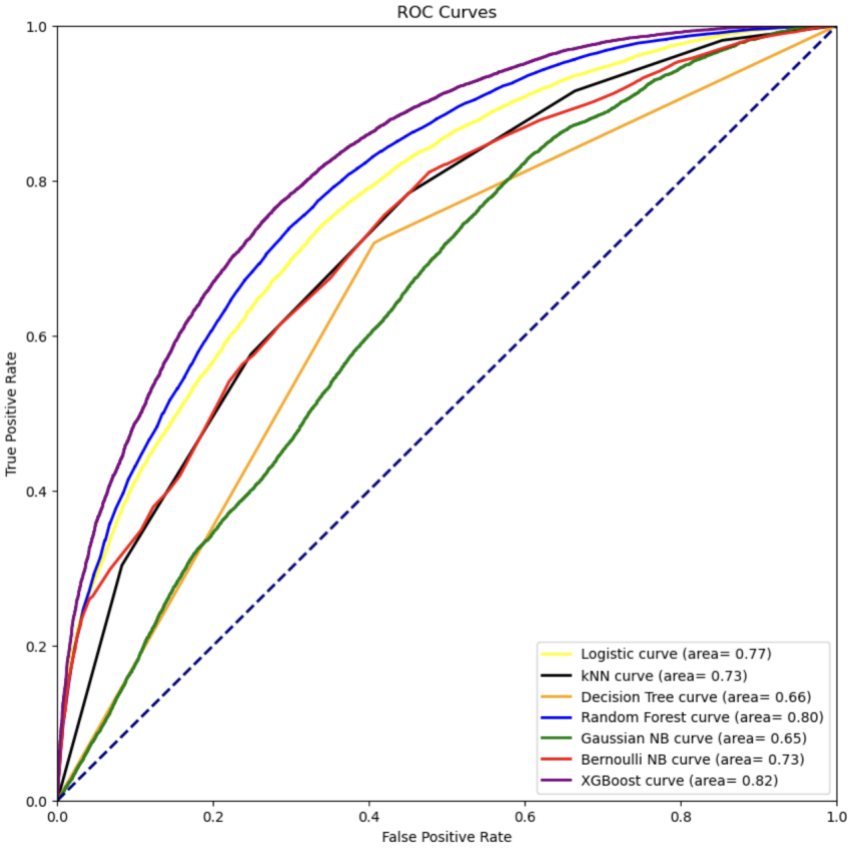


Рис. 3.2.1. – Порівняння ROC-кривих моделей класифікації

	Model_type	accuracy	precision	recall	f1	roc
0	KNN	0.695461	0.729276	0.790164	0.758500	0.735668
1	Logistic	0.715213	0.723881	0.855976	0.784406	0.773344
2	Decision Tree	0.664528	0.726711	0.714381	0.720493	0.651650
3	Random Forest	0.735827	0.761549	0.820410	0.789884	0.795849
4	Gaussian	0.629209	0.621540	0.990491	0.763794	0.644908
5	Bernoulli	0.692368	0.718666	0.808052	0.760742	0.730086
6	XGBoost	0.759888	0.770462	0.859285	0.812453	0.826736

Рис. 3.2.2. – Результати оцінки всіх моделей

Результати тестування показали, що модель XGBoost має найвищі показники точності (0.759888), precision (0.770462), recall (0.859285), F1-score (0.812453) і ROC-AUC (0.826736). Це свідчить про те, що модель

XGBoost здатна з високою точністю передбачати успішність краудфандингових проєктів на децентралізованій платформі.

Також помітно, що моделі Logistic Regression, Decision Tree і Random Forest також показали добрі результати, з високими показниками точності, precision і recall. Модель KNN та Bernoulli Naive Bayes показали менші значення цих метрик.

На основі проведеного порівняння моделей класифікації можна зробити висновок, що модель XGBoost є найбільш ефективною для передбачення успішності краудфандингових проєктів, проведемо детальніший аналіз моделі XGBoost.

3.3. Оцінка та результати моделі XGBoost

Після підбору параметрів та навчання моделі XGBoost на наборі даних, ми проводимо оцінку її ефективності та здатність передбачати успіх компаній. Для цього використовується тестовий набір даних, який не був використаний під час тренування моделі. Оцінка моделі включає різні метрики та відображає її здатність класифікувати дані коректно.

Проаналізуємо матрицю помилок яка надає уявлення про кількість правильно та неправильно класифікованих екземплярів кожного класу. У нашому випадку, матриця помилок виглядає таким чином:

$$\begin{matrix} 7634 & 4925 \\ 2704 & 16289 \end{matrix}$$

Перший рядок матриці представляє клас "0" (неуспішні компанії), а другий рядок – клас "1" (успішні компанії). Перший стовпець матриці відповідає екземплярам, які були класифіковані як "0", а другий стовпець - екземплярам, класифікованим як "1". За результатами, ми можемо побачити, що модель виявила 7634 успішних компанії правильно, але

помилилася, віднісши 4925 неуспішних компаній до класу "1". Додатково, модель правильно визначила 16289 неуспішних компаній і 2704 успішних компаній.

	precision	recall	f1-score	support
0	0.74	0.61	0.67	12559
1	0.77	0.86	0.81	18993
accuracy			0.76	31552
macro avg	0.75	0.73	0.74	31552
weighted avg	0.76	0.76	0.75	31552

Рис. 3.3. – Результати класифікації

Precision(точність) визначає, наскільки точно модель класифікує дані. У нашому випадку, точність для класу "0" становить 0.74, а для класу "1" - 0.77. Чутливість вказує на здатність моделі правильно визначати позитивні випадки (успішні компанії). Ми бачимо, що чутливість(повнота) для класу "0" дорівнює 0.61, а для класу "1" - 0.86. F1-score комбінує точність та чутливість і надає міру ефективності моделі. У нашому випадку, F1-score становить 0.67 для класу "0" і 0.81 для класу "1".

Значення accuracy (точність) вказує на загальну правильність класифікації моделі і дорівнює 0.76. Макро-середнє значення (macro avg) використовується для обчислення середньої ваги метрик для кожного класу, незалежно від розміру класу. У нашому випадку, макро-середня точність та чутливість становлять 0.75 та 0.73 відповідно. Зважене середнє значення (weighted avg) обчислюється, враховуючи розмір класів, тому воно відображає вагу кожного класу. Зважене середнє значення точності та чутливості становить 0.76.

Отримане значення площі під кривою ROC для моделі XGBoost становить 0.8267. Це свідчить про те, що модель має добру здатність розрізняти між позитивними та негативними випадками.

Ми також використовуємо оптимальний поріг рішення для класифікації. Поріг рішення визначає, який вихід моделі буде вважатися позитивним, а який негативним. Найкращий поріг рішення(threshold), визначений за допомогою показників TPR (чутливість) та FPR (специфічність), становить 0.616956.

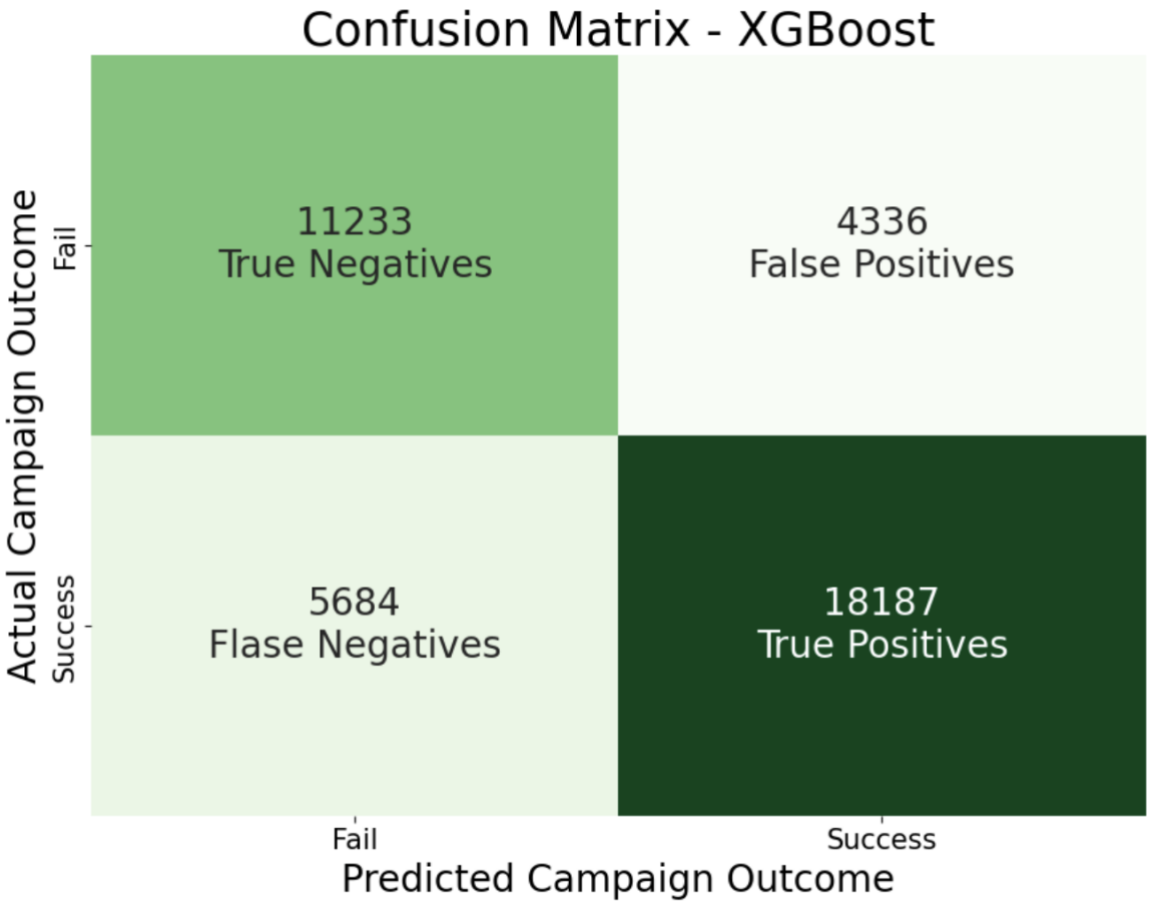


Рис. 3.3.1. – XGBoost – матриця плутанини при найкращому порозі рішення

Загалом, результати оцінки моделі XGBoost свідчать про її хорошу ефективність у передбаченні успіху компаній. Модель має задовільні значення точності, чутливості та F1-score для обох класів. Площа під кривою ROC підтверджує, що модель добре розрізняє між класами. Оптимальний поріг рішення допомагає досягнути кращого балансу між точністю та чутливістю при класифікації.

Можна зробити висновок що найефективніша модель є XGBoost з F1 0,8 і AUC 0,82. Порівняно з базовою моделлю (F1=0,75, AUC=0,65), показник був дещо покращений.

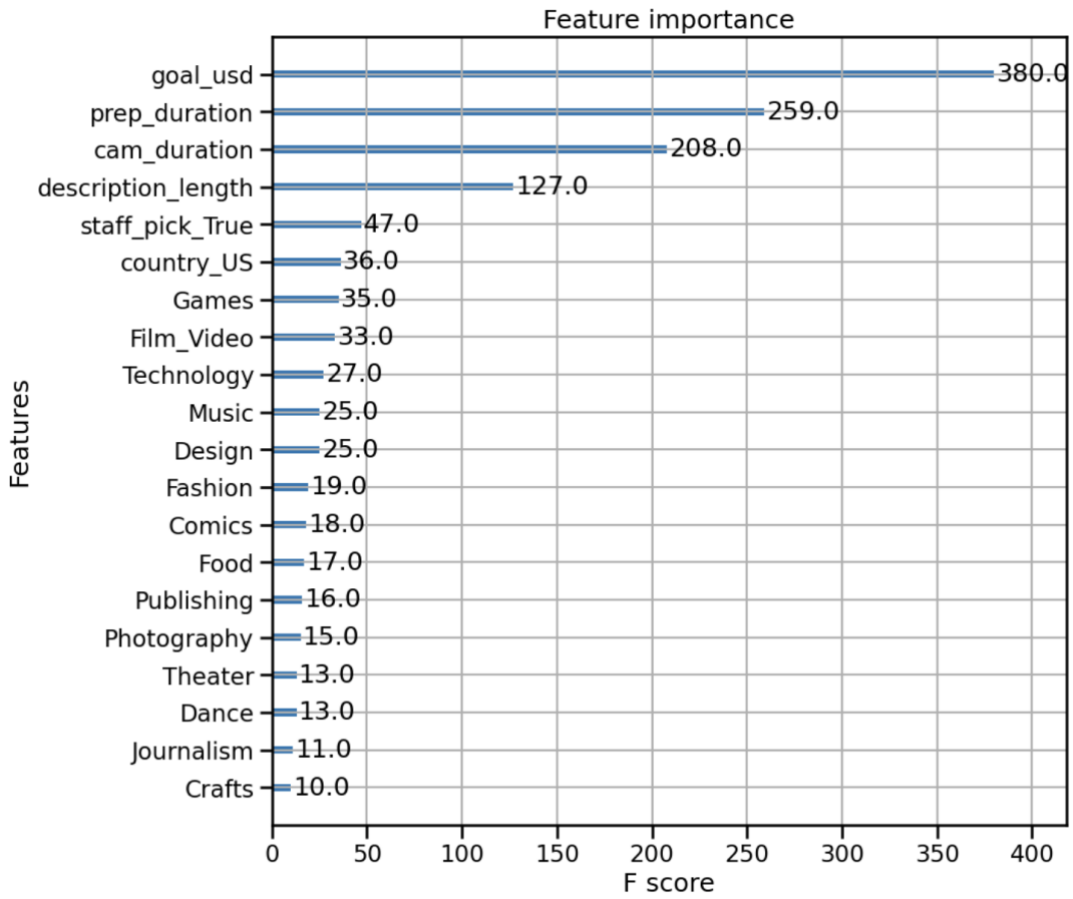


Рис. 3.3.2. – Важливість ознак в моделі XGBoost

Графік залежності між ознаками та їх F-score підтверджує важливість певних атрибутів в моделі класифікації успіху краудфандингових компаній. Згідно з наданими значеннями, кілька ознак виявилися особливо важливими для передбачення успішності проєктів.

Однією з ключових ознак є "goal_usd", яка відображає фінансову мету проєкту у доларах США. Ця ознака отримала високу оцінку і є важливим показником важливості для успіху проєкту. Крім того, ознаки, що

відносяться до тривалості кампанії ("cam_duration") та тривалості підготовки ("prep_duration"), також показали значимість у моделі.

Ознаки, пов'язані з категоріями проєктів, такі як "Comics", "Crafts", "Design", "Fashion", "Film_Video", "Games", "Journalism", "Music", "Photography", "Publishing", "Technology" та "Theater", також мають вплив на передбачення успішності. Ці ознаки репрезентують різні галузі творчості та дозволяють враховувати специфіку кожного сегмента.

Крім того, наявність ознаки "staff_pick_True" (проєкт, який отримав рекомендацію від команди платформи) та ознаки "country_US" (проєкт, розташований у Сполучених Штатах) також впливають на передбачення успішності краудфандингових проєктів.

Загалом, результати аналізу графіку підтверджують, що врахування цих ознак у моделі класифікації допоможе покращити її та забезпечити більш точні результати. Важливість цих ознак вказує на необхідність уваги до фінансових цілей, тривалості проєкту, категорій та рекомендацій команди платформи для досягнення успіху на децентралізованій платформі краудфандингу.

3.4. Результати роботи моделі

Testing of trained model

In [3]:
```python
import numpy as np
import pandas as pd
import xgboost as xgb
import pickle

# Завантаження навченої моделі
filename = "finalized_model.sav"
loaded_model = pickle.load(open(filename, 'rb'))

# Вхідні дані від користувача
goal = float(input("What is your goal (in USD)? "))
is_us = input("Is it based in the US? (True/False) ").lower() == 'true'
campaign_duration = int(input("What is your campaign duration (in days)? "))
preparation_duration = int(input("What is your preparation duration (in days)? "))
description = input("What is your campaign description? ")
is_staff_pick = input("Is it staff pick? (True/False) ").lower() == 'true'
category = input("Choose your category: ")

# Створення датафрейму з вхідними даними
test = pd.DataFrame([[goal, int(is_us), campaign_duration, preparation_duration,
                      len(description), int(is_staff_pick)]],
                    columns=['goal_usd', 'country_US', 'cam_duration', 'prep_duration',
                             'description_length', 'staff_pick_True'])

# Закодування категорії
categories = ['Comics', 'Crafts', 'Dance', 'Design', 'Fashion', 'Film_Video',
              'Food', 'Games', 'Journalism', 'Music', 'Photography',
              'Publishing', 'Technology', 'Theater']
category_encoded = [1 if cat == category else 0 for cat in categories]
test = pd.concat([test, pd.DataFrame([category_encoded], columns=categories)], axis=1)

# Прогнозування
prediction = loaded_model.predict(test)

# Виведення результату
if prediction[0] == 1:
    print("Your company is predicted to be successful.")
else:
    print("Your company is predicted to be unsuccessful.")
```

```
What is your goal (in USD)? 4000
Is it based in the US? (True/False) False
What is your campaign duration (in days)? 90
What is your preparation duration (in days)? 120
What is your campaign description? Компанія буде купувати котиків і продавати їх
Is it staff pick? (True/False) True
Choose your category: Dance
Your company is predicted to be successful.
```

Рис. 3.3.3 – Приклад тестування моделі в Jupyter №1

```
What is your goal (in USD)? 100000
Is it based in the US? (True/False) False
What is your campaign duration (in days)? 360
What is your preparation duration (in days)? 60
What is your campaign description? Scam for hamsters
Is it staff pick? (True/False) False
Choose your category: Theater
Your company is predicted to be unsuccessful.
```

Рис. 3.3.4 – Приклад тестування моделі в Jupyter №2

Висновок

В даній курсовій роботі було проведено детальне дослідження та розробка моделі класифікації машинного навчання для передбачення успіху краудфандингових компаній на децентралізованій платформі. За допомогою алгоритмів штучного інтелекту та аналізу різних факторів, таких як фінансові цілі, тривалість проєкту, категорії та рекомендації, було показано, що використання машинного навчання може допомогти у передбаченні ймовірності успіху проєкту.

Під час огляду літератури було проведено аналіз різних алгоритмів машинного навчання, зокрема K-найближчих сусідів (KNN) та дерев'яних ансамблів, які використовуються для класифікації даних. В результаті дослідження було вибрано відповідні алгоритми для розробки моделі класифікації успішності краудфандингових проєктів.

Попередня обробка даних включала нормалізацію, видалення відсутніх значень та кодування категоріальних змінних. Це дозволило підготувати дані для подальшого використання у моделі.

В результаті дослідження була розроблена та навчена модель класифікації, яка здатна передбачати успішність краудфандингових проєктів на основі вказаних факторів. Застосування машинного навчання у децентралізованій платформі краудфандингу дозволяє інвесторам та учасникам контролювати процес витрачання коштів та зробити більш обґрунтовані рішення.

Окрім того, проведено аналіз важливості ознак у моделі, де було виявлено, що фінансові цілі, тривалість проєкту, категорії та рекомендації команди платформи є ключовими факторами, які впливають на успішність краудфандингових проєктів.

Отже, розроблена модель класифікації на основі алгоритмів штучного інтелекту виявляється потенційно корисною для покращення процесу краудфандингу на децентралізованій платформі. Дане дослідження може

слугувати основою для подальших розвитку у сфері використання штучного інтелекту в краудфандингових платформах та забезпечити більш ефективну оцінку та управління проєктами.

Список використаної літератури

1. Kickstarter [Електронний ресурс] – https://www.kickstarter.com/
2. The Future of Equity Crowdfunding: Decentralized, Blockchain and AI [Електронний ресурс] – Посилання
3. Modelling Crowdfunding Ensemble Learning Prediction [Електронний ресурс] – Посилання
4. Prediction of Crowdfunding Project Success with Deep Learning [Електронний ресурс] –Посилання
5. Kaggle [Електронний ресурс] – https://www.kaggle.com
6. WebRobots [Електронний ресурс] – https://webrobots.io
7. Jupyter [Електронний ресурс] – https://jupyter.org
8. 5 Classification Algorithms for Machine Learning [Електронний ресурс] – Посилання
9. Interpretable Machine Learning with XGBoost [Електронний ресурс] – Посилання
10. Scikit-learn [Електронний ресурс] – https://scikit-learn.org/stable/
11. Seaborn [Електронний ресурс] – https://seaborn.pydata.org
12. XGBoost [Електронний ресурс] – Посилання
13. Top 6 Machine Learning Algorithms for Classification [Електронний ресурс] – Посилання

Додатки

Data_cleaning.ipynb

```
import pandas as pd
import numpy as np
from sqlalchemy import create_engine
from datetime import datetime, date
from ast import literal_eval

import warnings
warnings.filterwarnings('ignore')

engine = create_engine("sqlite:///Kickstarter.db")

query = "SELECT * FROM data"
df = pd.read_sql(query, engine)


print(df.shape)
df.head()

column_names = df.columns.tolist()
print(column_names)

# get rid of the "column-headers rows"
df = df[df.created_at != "created_at"]
df.shape

df = df.drop(columns=['creator', 'country_displayable_name', 'currency_symbol', 'currency_trailing_code', 'current_currency',
                      'disable_communication','friends', 'is_backing', 'is_starrable', 'is_starred',
                      'permissions', 'photo', 'profile', 'source_url', 'urls','usd_exchange_rate'])

# first looking at the ratio of successful and failed here
df.state.value_counts()

# Converting datetime objects
df['created_at'] = df['created_at'].astype(int)
df['created_at'] = pd.to_datetime(df['created_at'],unit='s')

df['deadline'] = df['deadline'].astype(int)
df['deadline'] = pd.to_datetime(df['deadline'], unit='s')

df['launched_at'] = df['launched_at'].astype(int)
df['launched_at'] = pd.to_datetime(df['launched_at'],unit='s')

df['state_changed_at'] = df['state_changed_at'].astype(int)
df['state_changed_at'] = pd.to_datetime(df['state_changed_at'],unit='s')
```

Baselining.ipynb

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression

from sklearn.metrics import classification_report
from sklearn.metrics import precision_score, recall_score, precision_recall_curve,f1_score, fbeta_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import roc_auc_score, roc_curve
from sklearn.metrics import roc_curve, auc

import warnings
warnings.filterwarnings('ignore')

df = pd.read_csv('../data/model/model.csv')
# check for class imbalance
```

```
print(df.success.value_counts())
sns.countplot(x="success", data=df, hue="success")

# Calculate the percentage of each campaign status
status_counts = df['success'].value_counts(normalize=True) * 100

# Plotting a pie chart
plt.figure(figsize=(8, 8), facecolor='lightgray')
plt.pie(status_counts, labels=status_counts.index, autopct='%1.1f%%', startangle=90, textprops={'color': 'black'})
plt.title('Campaigns Success Ratio', color='green')

# Equal aspect ratio ensures that pie is drawn as a circle
plt.axis('equal')

# Display the chart
plt.show()
print(df.success.value_counts())
target = df.success.value_counts(normalize=True)
print(target)

def make_confusion_matrix(model, threshold=0.6):
    # Predict class 1 if probability of being in class 1 is greater than threshold
    # (model.predict(X_test) does this automatically with a threshold of 0.5)
    y_predict = (model.predict_proba(X_test_sc)[:,1] >=threshold)
    success_confusion = confusion_matrix(y_test, y_predict)
    plt.figure(dpi=80)
    sns.heatmap(success_confusion, cmap=plt.cm.Blues, annot=True, square=True, fmt='d',
            xticklabels=['fail','success'],
            yticklabels=['fail','success']);
    plt.xlabel('prediction')
    plt.ylabel('actual')

# baselining with small subset of features
subset_1 = df.loc[:,['success', 'goal_usd', 'country_US', 'cam_duration']]
subset_1.head()
```

Final_model_test.ipynb

```
import numpy as np
import pandas as pd
import xgboost as xgb
import pickle

# Завантаження навченої моделі
filename = "finalized_model.sav"
loaded_model = pickle.load(open(filename, 'rb'))

# Вхідні дані від користувача
goal = float(input("What is your goal (in USD)? "))
is_us = input("Is it based in the US? (True/False) ").lower() == 'true'
campaign_duration = int(input("What is your campaign duration (in days)? "))
preparation_duration = int(input("What is your preparation duration (in days)? "))
description = input("What is your campaign description? ")
is_staff_pick = input("Is it staff pick? (True/False) ").lower() == 'true'
category = input("Choose your category: ")

# Створення датафрейму з вхідними даними
test = pd.DataFrame([[goal, int(is_us), campaign_duration, preparation_duration,
                      len(description), int(is_staff_pick)]],
                    columns=['goal_usd', 'country_US', 'cam_duration', 'prep_duration',
                             'description_length', 'staff_pick_True'])

# Закодування категорії
categories = ['Comics', 'Crafts', 'Dance', 'Design', 'Fashion', 'Film_Video',
              'Food', 'Games', 'Journalism', 'Music', 'Photography',
              'Publishing', 'Technology', 'Theater']
category_encoded = [1 if cat == category else 0 for cat in categories]
test = pd.concat([test, pd.DataFrame([category_encoded], columns=categories)], axis=1)

# Прогнозування
prediction = loaded_model.predict(test)

# Виведення результату
if prediction[0] == 1:
    print("Your company is predicted to be successful.")
```

```
else:
    print("Your company is predicted to be unsuccessful.")
```

Посилання на GitHub репозиторій проєкту.

https://github.com/billyxd3/Crowdfunding-Prediction-Model/tree/main