

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»



Факультет комп'ютерних наук та технологій
Кафедра «Комп'ютерні системи та мережі»

ЛЕВЧЕНКО АНДРІЙ МИХАЙЛОВИЧ
Група КНТ-622м

КОМП'ЮТЕРНА СИСТЕМА ДЛЯ УПРАВЛІННЯ
ВИТРАТАМИ НА ОСНОВІ ШТУЧНОГО ІНТЕЛЕКТУ

АВТОРЕФЕРАТ

магістерської роботи на здобуття освітньо-кваліфікаційного
рівня «магістр» 123 «Комп'ютерна інженерія»
освітньої програми «Спеціалізовані комп'ютерні системи»

2023 р.

Магістерська робота є рукопис.

Робота виконана в Національному університеті «Запорізька політехніка», на кафедрі комп'ютерних систем та мереж

Керівник

кандидат технічних наук, доцент
Скрупський Степан Юрійович,
Національний університет «Запорізька
політехніка», доцент кафедри
комп'ютерних систем та мереж

**Офіційний
рецензент:**

кандидат фізико-математичних наук,
доцент
Самойлик Сергій Сергійович,
Національний університет «Запорізька
політехніка», доцент кафедри
"Радіотехніки та телекомунікацій"

Захист відбудеться "13" грудня 2023 р.

Секретар екзаменаційної комісії, старший викладач
кафедри комп'ютерних систем та мереж Н.А. Куликовська

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. Після виникнення перших спільнот людей з'явилася необхідність в обміні матеріальними речами. Людина могла щось виготовляти чи мати більше ніж їй необхідно та при цьому потребувала щось інше. Бартер став першою формою обміну. Його сутність складалась з обміну певної кількості чогось на певну кількість чогось іншого. Через це виникала проблема, що якщо у людей нема необхідних для одне одного речей, то обмін неможливий. Цю проблему можна вирішити з застосуванням універсальної міри обміну. З часом така міра обміну була зроблена й отримала назву гроші.

Сьогодні існує безліч різних національних валют, що признаються світом. Також популярність набирають альтернативні валюти такі як криптовалюта. Гроші для людей це можливість купити товар чи послугу, що зробить їх життя краще, тому люди намагаються збільшити своє багатство. Для цього існують два шляхи – оптимізація витрат та інвестиція. Перший варіант ефективний та не потребує особливих передумов, але він має обмеження. Другий варіант цих обмежень не має, але має ризики. Для розв'язання цих проблем можливе застосування комп'ютерної системи управління витратами на основі штучного інтелекту, що може з деяким успіхом прогнозувати майбутні витрати чи ризики при інвестиції.

Зважаючи на об'ємність теми управління витратами сконцентровано увагу на прогнозуванні доходів та, особливо, витрат, а також на підвищенні точності передбачень шляхом дослідження різних підходів. Тема є важливою та принциповою, оскільки прогнозування є складним завданням через багато факторів, які впливають на результат, та має низьку точність для довгострокових передбачень.

Таким чином, актуальною та важливою є розробка комп'ютерної системи для управління витратами на основі штучного інтелекту, що дозволить робити аналіз над даними минулих транзакцій та з певною точністю прогнозувати майбутні доходи та витрати. Також для завдання індивідуального управління витратами бажано, щоб комп'ютерна система мала можливість розміщуватись на мобільних пристроях.

Мета і завдання дослідження. Метою проєкту (роботи) є розробка та дослідження комп'ютерної системи для управління персональними витратами на мобільних пристроях.

Для досягнення поставленої мети необхідно виконати такі основні завдання:

- провести аналіз проблеми та засобів її вирішення;
- розробити структуру та алгоритм роботи системи відповідно до результатів першого пункту;
- реалізувати та дослідити комп'ютерну систему для управління витратами.

Об'єктом дослідження є комп'ютерна система для управління витратами на основі штучного інтелекту.

Предметом дослідження є використання методів машинного навчання для прогнозування.

Методи дослідження: мова програмування Dart, фреймворк Flutter, система керування базами даних SQLite, інтегроване середовище розробки Android Studio, мова програмування Python, бібліотека TensorFlow Lite, рекурентна нейронна мережа LSTM.

Наукова новизна отриманих результатів:

Наукова новизна даного проєкту (роботи) є у розробці та тренуванні власної моделі на основі штучного інтелекту з використанням LSTM, що може бути розміщена на мобільних пристроях.

Практичне значення отриманих результатів:

Практична цінність даного проєкту (роботи) полягає у реалізації комп'ютерної системи на основі моделі, яка дозволяє покращити конфіденційність, швидкодію на 8,7%, не вимагає підключення до мережі Інтернет та адаптує модель на основі штучного інтелекту до даних користувача у порівнянні з аналогами.

Апробація результатів магістерської роботи. Основні положення магістерської роботи та результати досліджень не подано до участі на конференції.

Структура та обсяг роботи. Магістерська робота складається з переліку скорочень та умовних познач, вступу, трьох розділів, висновків та списку використаних джерел. Основна частина містить 73 сторінки, 25 рисунків, 1 таблицю, 11 лістингів та список використаних джерел зі 56 найменувань.

ОСНОВНИЙ ЗМІСТ РОБОТИ

У першому розділі було визначено поняття управління витратами, сформульовано завдання та обрано кросплатформний підхід для мобільної розробки. Мова програмування Dart та фреймворк Flutter були обрані для створення мобільного застосунку, тоді як для розробки моделі на основі штучного інтелекту у мобільному додатку був обраний фреймворк TensorFlow Lite. Також проведено аналіз мови програмування Python у контексті машинного навчання та розглянуті різні методи машинного навчання.

Після аналізу видів розробки мобільних застосунків був зроблений наступний висновок, що кросплатформний підхід краще через наступні переваги:

- дешева та швидка мобільна розробка;
- невелике падіння швидкодії порівняно з нативним підходом;
- простіші мови програмування для опанування.

Мова програмування Dart була обрана для реалізації кросплатформного підходу через свої переваги, а саме простоти та підтримки від Google. Проте мови Java та Kotlin для Android, а також Objective-C та Swift для iOS можуть бути використані для написання нативного коду у частині застосунку, що буде найпродуктивнішим рішенням.

Фреймворк Flutter був обраний через кращу швидкодію на мобільних пристроях у порівнянні з іншими кросплатформними фреймворками, сучасний підхід у розробці та дизайні застосунків та підтримкою від Google, а також використання мови програмування Dart.

Для розробки моделі на основі штучного інтелекту в комп'ютерній системі було обрано фреймворк TensorFlow Lite через велику кількість документації, довгий строк у релізі та підтримку від Google.

При аналізі мови програмування Python для використання у машинному навчанні було визначено, що він може ефективно працювати з TensorFlow та використовувати всі наявні ресурси пристрою, а також IDE JupyterLab для Python є зручним інструментом для тренування моделей на основі штучного інтелекту.

У другому розділі при розробці мобільного застосунку був реалізований архітектурний шаблон MVC, підключена система керування базами даних SQLite, а саме бібліотека sqflite, що оптимізована для використання з фреймворком Flutter. Були розроблені віджети представлення для перегляду історії транзакцій, додавання нової транзакції та перегляду результатів прогнозів у вигляді графіків за допомогою бібліотеки fl_chart.

При розробці моделі на основі штучного інтелекту було представлено процес тренування на прикладі лінійної регресії. Після навчання модель була збережена у форматі TensorFlow Lite для подальшої інтеграції з мобільним застосунком, використовуючи інтерпретатор з бібліотеки tf_lite_flutter, що була розроблена для використання з фреймворком Flutter.

Розробка комп'ютерної системи складається з розробки мобільного застосунку в IDE Android Studio за допомогою фреймворку Flutter та мови програмування Dart та розробки моделі на основі штучного інтелекту інструментами мови Python та TensorFlow Lite.

Створення мобільного застосунку складається з трьох етапів:

- вибір та впровадження патерну проєктування;
- розробка та налаштування бази даних для зберігання та обробки інформації;
- створення віджетів представлення для користувацького інтерфейсу.

Архітектурним шаблоном було обрано MVC (Model-View-Controller). Патерн проєктування MVC визначає, що застосунок складається з моделі даних (Model), інформації для відображення (View) та інформації для управління (Controller). Цей шаблон вимагає розділення кожного з цих компонентів на окремі об'єкти. MVC основним чином стосується інтерфейсного/взаємодійного рівня застосунку.

У теці моделей є файл app_transaction.dart, що відповідає за дані транзакцій у застосунку. У теці контролеру є файл database_controller, що відповідно до назви відповідає за зв'язок з базою даних. У теці представлення є файли analytics.dart, history.dart, home_screen.dart та main.dart, що відповідають за відповідні екрани застосунку. Файл main.dart, що знаходиться у теці lib має в собі точку входу застосунку.

На рисунку 1 зображено скриншот IDE Android Studio на якому показано склад теки lib, яка містить в собі теки проєкту розбиті за принципами патерну проєктування MVC.

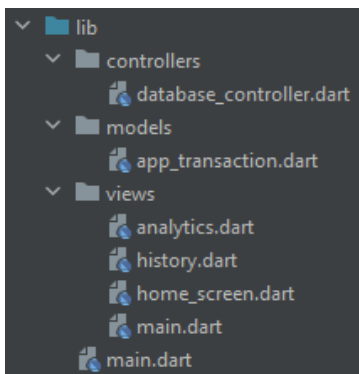


Рисунок 1 – Теки проєкту розбиті за принципами патерну проєктування MVC

Для мобільного застосунку була обрана полегшена реляційна система керування базами даних SQLite, а саме бібліотека sqflite версії 2.3.0, що є пакетом оптимізованим для використання у фреймворці Flutter.

Для повного контролю над базою даних у файловій системі пристрою було використано бібліотеку path версії 1.8.3.

База даних для зберігання транзакцій має наступні поля: ідентифікатор, назва, сума та дата транзакції. Відповідно до існуючих типів даних у SQLite та Dart ці поля визначені наступним чином:

- id (INTEGER PRIMARY KEY, int);
- name (TEXT, String);
- amount (REAL, double);
- date (TEXT за стандартом ISO8601, DateTime).

Для форматування дати та додання української локалізації при відображенні назв місяців була використана бібліотека intl версії 0.18.1.

Для візуалізації результатів прогнозування у вигляді графіків у мобільному застосунку представлений клас Analytics. Для цього була використана бібліотека fl_chart версії 0.64.0.

На рисунках 2-3 наведені скриншоти з емулятора Android Studio, що демонструють роботу представлення мобільного застосунку, а саме перегляд історії та ввід нової транзакції відповідно.



Рисунок 2 – Перегляд історії

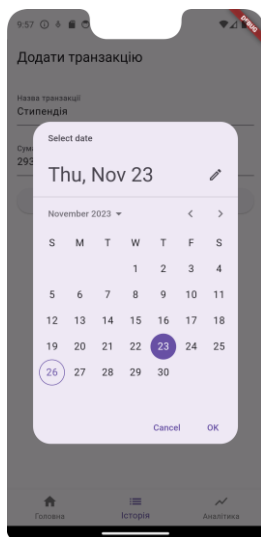


Рисунок 3 – Додавання транзакції

Розробка моделі на основі штучного інтелекту складається з тренування моделі засобами мови Python, збереження у форматі TensorFlow Lite та подальшою інтеграцією у комп'ютерну систему.

Для машинного навчання на мові Python були використані наступні бібліотеки: NumPy, Pandas та Matplotlib. Вони допоможуть у математичних обчисленнях, поданні масивів даних та представленні результатів через візуалізацію на графіках.

Для впровадження моделі на основі штучного інтелекту в мобільний застосунок була використана бібліотека `tflite_flutter` версії 0.10.3. Цей плагін для Flutter надає гнучке та швидке рішення для доступу до інтерпретатора TensorFlow Lite та інтеграції моделі з системою. Його API схоже на TFLite Java та Swift API. Він прямо пов'язаний з TFLite C API, що робить його ефективним через низьку латентність.

Варто зазначити, що для коректної роботи TensorFlow Lite необхідно мати операційну систему як мінімум Android 8 (API Level 26), тобто версії, які будуть старші за вказану не будуть підтримуватись та відповідно і працювати.

У третьому розділі було проведено дослідження моделі на основі штучного інтелекту, що показало ефективні та слабкі методи для прогнозування. Так було виявлено, що метод опорних векторів при всіх параметрах робить прогнозування неточним та у випадку з поліноміальним та радіально базисним ядром протилежну тенденцію росту передбачень. Також лінійна регресія через свою простоту та RNN через помилкові очікування майбутніх значень не є гарними методами машинного навчання для прогнозування, але LSTM (архітектура рекурентної нейронної мережі) виявилася дуже ефективною мережею для прогнозування майбутніх значень часових рядів.

Після випробування комп'ютерної системи було виявлено, що значення прогнозу розраховуються згідно з очікуваними значеннями, що показує правильність та правдивість роботи застосунку. Також було обчислено час виконання програми при прогнозуванні та порівняння з аналогом, який має інше архітектурне рішення в розміщенні моделі на основі штучного інтелекту, а саме на сервері, тобто всі обчислення проходять там, а мобільний пристрій лише надсилає та отримує запити. Порівняння по критерію час обчислення прогнозу показало, що комп'ютерна

система швидше на 0.1 с (8,7%) ніж аналог при найкращих умовах. А при відсутності доступу до мережі Інтернет застосунок має абсолютну перевагу над аналогами, які не мають змоги отримати прогноз у таких умовах.

Прогноз LSTM на наступний місяць становить 18780,30 ₴, що не сильно відрізняється від попереднього місяця, але найголовніше прогноз на рік має тенденцію до росту. Через що цей прогноз можна вважати правдоподібним.

На рисунку 4 зображено прогноз доходу при використанні LSTM.

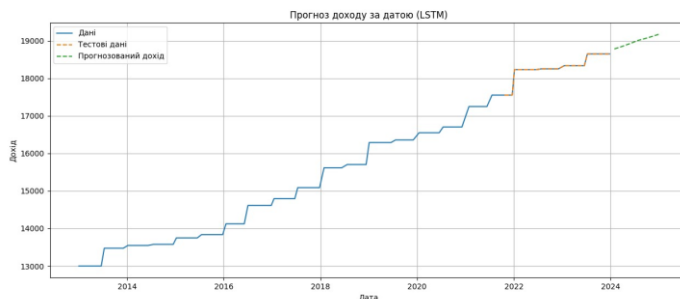


Рисунок 4 – Прогноз доходу при застосуванні LSTM

Для аналізу отриманих результатів необхідно подивитись на значення середньоквадратичної похибки при кожній епосі (крок навчання). Графік цієї залежності наведено на рисунку 5.

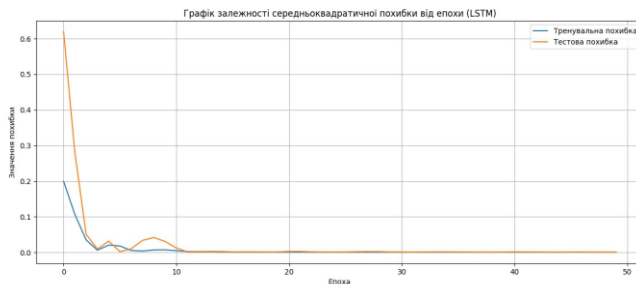


Рисунок 5 – Середньоквадратична похибка при прогнозуванні доходів LSTM

Як видно з графіку значення похибки досягає значення 0.01 до десяти епох, а далі наближається до нуля, але не досягає його. З цього можна зробити висновок, що після 15 епох тренування модель машинного навчання на основі LSTM вже має достатню навченість для ефективного прогнозування наданих вхідних даних.

Тепер розглянемо використання LSTM при прогнозуванні витрат, які були визначені раніше. Цей прогноз зображено на рисунку 6.

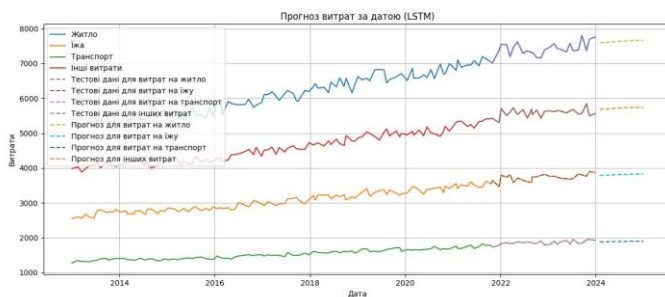


Рисунок 6 – Прогноз витрат при застосуванні LSTM

Як ми бачимо з графіку та результатів прогнозування результати є правдоподібними, але задача передбачення на цілий рік не дає чітких представлень, а лише обмежується майже лінійною залежністю.

Через це можна зробити висновок, що LSTM підходить для прогнозування витрат у наступному місяці, але не дає корисних результатів для довгострокового результату.

На рисунку 7 зображено екран мобільного застосунку, що представляє графік прогнозу.

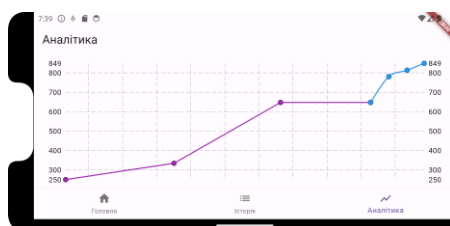


Рисунок 7 – Графік прогнозу

Фіолетовим кольором позначено графік, що побудований згідно з даними доходів, які зберігаються в локальній базі даних. Голубим кольором позначенні чотири значення: поточний місяць та прогнозовані значення на три наступних. Історія за минулі транзакції та передбачені результати, що представлені на графіку відповідають очікуванням.

На рисунку 8 зображено головний екран застосунку, на якому є можливість ввести транзакцію, а також показані згідно обчислень локальної моделі на основі штучного інтелекту прогнози на наступний місяць.

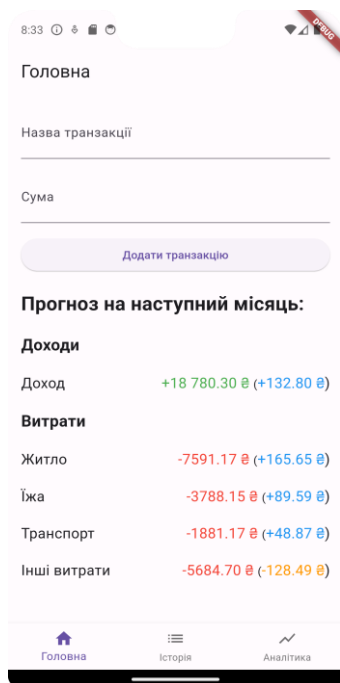


Рисунок 8 – Головна сторінка застосунку

Значення доходу відображені зеленим кольором, у той час, як витрати червоним, при цьому якщо прогноз вигідний для користувача, то він має голубий колір, а якщо поганий, то помаранчевий. Прогнозовані результати відповідають очікуванням та вважаються вірогідними.

Для вимірювання ефективності роботи комп'ютерної системи було проведено заміри для визначення необхідної кількості часу для обчислення прогнозу. Також були проведені подібні вимірювання для схожих застосунків, щоб порівняти їх між собою за критерієм часу виконання програми. Через те що не було знайдено подібного мобільного застосунку персонального управління витрат з архітектурним рішенням у вигляді розміщення моделі на основі штучного інтелекту на пристрої, то лише є змога порівняти зі схожими застосунками, але які замість обчислень локально відправляють дані на сервер.

На рисунку 9 зображено порівняння мобільного застосунку з аналогом за витраченим часом на обчислення прогнозу.

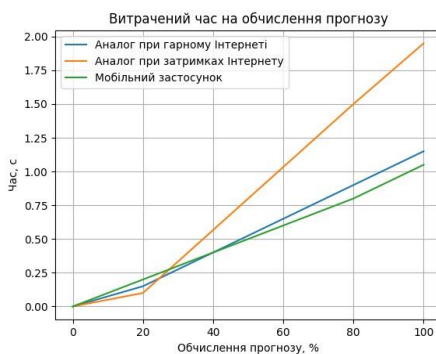


Рисунок 9 – Витрачений час на обчислення прогнозу

Зеленим кольором на графіку позначено розроблений мобільний застосунок, синім кольором аналог при сприятливих умовах в мережі та помаранчевим випадок коли існують затримки в мережі Інтернет. З цього графіку можна визначити те, що розроблена комп'ютерна система робить обчислення прогнозу на 0,1 с швидше ніж наведений аналог, але це не значить, що реальні обчислення виходять швидше. Через обмеженість апаратних ресурсів мобільної платформи застосунок не може проводити розрахунки швидше ніж сервер, але через те що дані доводиться пересилати мережею Інтернет, то втрачається час за який комп'ютерна система з локальною моделлю на основі штучного інтелекту стає ефективнішою.

Також очевидним є той факт, що при відсутності доступу до мережі Інтернет взагалі, то такі обчислення неможливі в принципі.

У майбутньому розроблена комп'ютерна система при достатній навченості моделі та збільшенні апаратних ресурсів мобільних пристроїв стане альтернативою для застосунків, які мають модель на основі штучного інтелекту на сервері.

ВИСНОВКИ

Результати, отримані в магістерській роботі, є рішенням практичної задачі підвищення ефективності прогнозування для управління витратами.

Отримано такі теоретичні та практичні результати:

1. Розглянуто поняття управління витратами, виявлені основні способи мобільної розробки та був обраний кросплатформний підхід. Проаналізовано мови програмування, фреймворки для розробки та штучного інтелекту. Обрана мова Dart, фреймворк Flutter та TensorFlow Lite. Проведено аналіз використання мови Python та методів машинного навчання.

2. Проведена розробка комп'ютерної системи у якій було впроваджено архітектурний шаблон MVC, під'єднано систему керування базами даних SQLite та розроблені віджети мобільного застосунку. Проведено тренування та інтеграція у комп'ютерну систему моделі на основі штучного інтелекту.

3. Проведено дослідження методів машинного навчання, яке показало перевагу LSTM у прогнозованні часових рядів над іншими підходами. Проведено випробування системи, яке підтвердило відповідність всіх вимог визначеного завдання, а саме реалізація комп'ютерної системи на основі моделі, яка дозволяє покращити конфіденційність, швидкодію на 8,7%, не вимагає підключення до мережі Інтернет та адаптує модель на основі штучного інтелекту до даних користувача у порівнянні з аналогами.

Таким чином, всі поставлені в роботі задачі виконані й мета досягнута.