Portfolio Optimizer

# Numery indeksów członków grupy projektowej:

* s25822
* s25819
* s24187

# Repozytorium kodu:

Link do repozytorium na **GitHubie**: <https://github.com/kacp-k01/PROJEKT_SUML_L_25>

# Opis funkcjonalności aplikacji

Aplikacja służy do predykcji przyszłych cen wybranych instrumentów finansowych na podstawie danych historycznych pobranych dzięki bibliotece Yahoo Finance (yfinance).

Do samej predykcji aplikacja wykorzystuje sieć neuronową LSTM. Użyta zostanie wersja z pakietu Keras. Będzie trenowana na zawołanie, na podstawie danych wybranych przez użytkownika.

Portfolio Optimizer posiada również opcję:

* wizualizacji wyników za pomocą raportu tekstowego i komponentów graficznych
* zapisywania ewentualnych wyników lokalnie, w celu porównania różnych analiz.

# Typ aplikacji

Portfolio Optimizer to aplikacja desktopowa, z częścią GUI wykonaną z użyciem biblioteki Tkinter. Wygenerowany plik aplikacji ważył prawie 1 GB, więc nie jest przechowywany bezpośrednio w zewnętrznym repozytorium. Zamiast tego, w pliku ***README.md*** zawarto instrukcję opisującą komendy potrzebne do wygenerowania pliku wykonawczego ***.exe*** lokalnie (użyto do tego biblioteki PyInstaller). Wersja Python’a to **3.12**, a wersje bibliotek użytych komponentów znajdują się standardowym pliku requirements.txt i są następujące:

* matplotlib==3.9.2
* tk==0.1.0
* tkhtmlview==0.3.1
* yfinance==0.2.62
* scikit-learn~=1.5.1
* pandas~=2.2.2
* numpy~=1.26.4
* services~=0.1.1
* tf-keras~=2.17.0

# Obraz zawierający tekst, zrzut ekranu, Czcionka Zawartość wygenerowana przez AI może być niepoprawna.Struktura aplikacji

**.gitignore** – zawiera listę obiektów, które nie są przechowywane na zdalnym repozytorium – zawiera przede wszystkim pliki i foldery powstałe przy lokalnym tworzeniu pliku wykonawczego .exe przy użyciu biblioteki PyInstaller:

* Foldery:
  + **build/**
  + **dist/**
* Plik: ***projekt\_suml.spec***

Dokumentacja oraz wymagania zawarte są w plikach:

* ***README.md***
* ***requirements.txt***
* oraz tym raporcie, znajdującym się w folderze **raport/**.

Pliki kodu aplikacji:

* ***screen.py*** – zawiera kod GUI aplikacji
* ***data\_fetcher.py*** – zawiera metody pobierające dane instrumentów finansowych do treningu modelu oraz informacyjne
* **model.py** – zawiera kod trenowanie modelu i wstępnej obróbki danych
* ***predictor.py*** – zawiera metodę wyliczającą wartości instrumentu finansowego w przyszłości
* ***resource\_saver.py*** – zawiera metody zapisujące wyniki analizy w lokalnym folderze
* ***project\_suml.py*** – plik główny projektu.

# Dataset

Zbiór danych to historyczne dane instrumentów finansowych (akcji, funduszy) pobierane z biblioteki Yahoo Finance.

Są one pobierane "*on-demand*" przez użytkownika, z wykorzystaniem darmowego API (zakres czasowy ustanowiono na ostatnie 5 lat).

Zapytania zwracają serię danych na podstawie podanych "ticker'ów" (oznaczeń instrumentów finansowych) - wprowadzi je użytkownik.

Dodatkowo, z tej samej biblioteki pobrane zostaną podstawowe dane informacyjne o instrumencie finansowym, takim jak jego waluta, pełna nazwa, sektor itp.

# GUI Aplikacji

## Startowe:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Zawartość wygenerowana przez AI może być niepoprawna.

## Podczas predykcji

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Zawartość wygenerowana przez AI może być niepoprawna.

## Obraz zawierający tekst, Czcionka, zrzut ekranu, biały Zawartość wygenerowana przez AI może być niepoprawna.Okno parametrów

**Wybór Tickera** – to tuja podaje się nazwę skróconą instrumentu finansowego – w tym wypadku MSFT to akcje firmy Microsoft.

Obraz zawierający tekst, Czcionka, zrzut ekranu, biały

Zawartość wygenerowana przez AI może być niepoprawna.Obraz zawierający tekst, Czcionka, zrzut ekranu

Zawartość wygenerowana przez AI może być niepoprawna.3 okna wyboru dostosowujące parametry modelu.

**Horyzon predykcji** może być ustawiony na 1, 5, 30, 60 albo 100 dni w przyszłość

**Epoki treningu** (ang. *epochs*) oraz **wielkość grupy** (ang. *batch*) są podawane porzez wpisanie wartości.

## Wykres cen instrumentów

Obraz zawierający tekst, Wykres, linia, zrzut ekranu

Zawartość wygenerowana przez AI może być niepoprawna.

Zawiera porównawczy wykres cen z ostatnich 5-ciu lat (kolor granatowy) oraz przewidywane wartości w wybranym oknie czasowym (kolor pomarańczowy).

## Raport

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Zawartość wygenerowana przez AI może być niepoprawna.Składa się z 3 elementów:

**Informacji o spółce**

**Parametrów predykcji** – tych wybranych przez użytkownika, jak i pobranych z Yahoo Finance

**Statystyki modelu** – są to:

* ***Błąd średniokwadratowy*** (ang. *Mean Square Error*, MSE) – jest wartością oczekiwaną kwadratu „błędu”, czyli różnicy między estymatorem a wartością estymowaną.
* ***Strata końcowa wyniku*** – finalna wartość funkcji MSE, aktualizowana po każdej epoce treningu.
* ***Pierwiastek błędu średniokwadratowego*** (ang. *Root Mean Square Error*, RMSE) – jest to pierwiastek MSE – pokazuje różnicę w jednostce w której jest wartość wejściowa
* ***Średni błąd bezwględny*** (ang. *Mean Absolute Error*, MAE) - średnia wielkość błędów bezwzględnych między przewidywanymi a rzeczywistymi wartościami.

## Przyciski kontrolne:

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Zawartość wygenerowana przez AI może być niepoprawna.Służą do kontroli nad aplikacją:

**Uruchom predykcję** – uruchamia predykcję, jeśli parametry wejściowe są poprawne.

**Zapisz wynik** – zapisuje wybrane wyniki predykcji (wykres i/lub raport) do wybranego lokalnego folderu, jeśli predykcja została wcześniej uruchomiona.

**Zamknij program** – zamyka program.