|  |  |  |  |
| --- | --- | --- | --- |
| agh_nzw_s_pl_1w_wbr_rgb_150ppi  **WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI, INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ**  Informatyka w sterowaniu i zarządzaniu  Optymalizacja wielokryterialna  *Aplikacja do rekomendacji filmów z wykorzystaniem metod optymalizacji wielokryterialnej* | | | |
| ***L.p.*** | **Członek** | **Numer albumu** | **Adres e-mail** |
| *1* | Artur Mzyk | 400658 | arturmzyk@student.agh.edu.pl |
| *2* | Joanna Nużka | 400561 | [joannanuzka@student.agh.edu.pl](mailto:joannanuzka@student.agh.edu.pl) |
| *3* | Adrian Poniatowski | 401346 | [adrianponiat@student.agh.edu.pl](mailto:adrianponiat@student.agh.edu.pl) |

Spis treści

[1. Wstęp 3](#_Toc153811445)

[1.1. Cel projektu 3](#_Toc153811446)

[1.2. Akwizycja danych 3](#_Toc153811447)

[2. Badany problem 3](#_Toc153811448)

[2.1. Opis problemu 3](#_Toc153811449)

[3. Propozycja rozwiązania 4](#_Toc153811450)

[3.1. TOPSIS 4](#_Toc153811451)

[3.2. UTA 4](#_Toc153811452)

[3.3. Metoda zbiorów referencyjnych 4](#_Toc153811453)

[4. Aplikacja 4](#_Toc153811454)

[5. Eksperymenty 5](#_Toc153811455)

[6. Podsumowanie/wnioski 5](#_Toc153811456)

[6.1. Zrealizowane punkty 5](#_Toc153811457)

[6.2. Napotkane problemy 5](#_Toc153811458)

[6.3. Kroki dalszego rozwoju 5](#_Toc153811459)

[7. Spis literatury 5](#_Toc153811460)

[8. Podział pracy 6](#_Toc153811461)

# Wstęp

## Cel projektu

Celem projektu było zbudowanie aplikacji do rekomendacji użytkownikowi filmów na podstawie jego preferencji. Wykorzystywane są poznane na zajęciach metody optymalizacji wielokryterialnej, głównie metody rankingowe, takie jak Topsis, UTA czy metoda zbiorów referencyjnych. Użytkownikowi proponowane są trzy filmy najbardziej zbliżonych do jego wymagań, na które składają się:

* gatunek,
* rok produkcji,
* obsada,
* ilość zastosowanych efektów specjalnych,
* autor książki, na podstawie której powstał film,
* ogólna ocena użytkowników portalu.

## Akwizycja danych

Dane zaczerpnięte zostały z ogólnodostępnej bazy filmów IMDB za pośrednictwem pakietu Pythona IMDbPy. Oprócz zbioru filmów zawiera ona kluczowe informacje na ich temat   
i udostępnia szeroki zakres funkcji ułatwiających filtrowanie.

Dane zostały na początku poddane preprocessingowi i zweryfikowane pod względem kompletności. Interfejs graficzny pozwala na filtrowanie filmów ze względu na gatunek.

# Badany problem

## Opis problemu

Ważne jest zdefiniowanie funkcji dopasowania określającej, jak dobre jest dopasowanie danego filmu do podanych przez użytkownika preferencji. Pozwala ona ocenić film na podstawie trzech kryteriów, z których każde ma określoną skalę:

* liczba pokrywających się z preferencjami aktorów i reżyserów: 0-10,
* ocena na portalu IMDB: 0-10,
* odległość od średniego roku produkcji obejrzanych przez użytkownika filmów: 0-3 (danemu interwałowi przypisana jest odpowiednia ocena ze skali).

# Propozycja rozwiązania

## TOPSIS

Metoda TOPSIS to metoda optymalizacji wielokryterialnej bazująca na wyznaczaniu odległości badanych punktów od danego punktu idealnego i antyidealnego dla badanego problemu. Na podstawie tych odległości wyliczany jest współczynnik scoringowy służący do uszeregowania wyników. Największa wartość tego współczynnika odpowiada najlepszemu obiektowi dla danego problemu.

Zaimplementowana przez nas metoda TOPSIS składa się z opisanych dalej etapów   
i elementów. Pierwszym krokiem jest zbadanie kierunków optymalizacji dla każdego kryterium. Dla kryterium maksymalizowanego następuje odwrócenie wszystkich wartości tak, aby otrzymać problem minimalizacji. Następnie dokonywane jest odfiltrowanie punktów zdominowanych a dalsza analiza występuje tylko dla punktów niezdominowanych. Punkty niezdominowane zapisywane są w macierzy ewaluacji. Macierz ta jest normalizowana przy użyciu normy euklidesowej oraz skalowana za pomocą przemnożenia przez wektor wag. Wagi są wprowadzane przez użytkownika przy pomocy interfejsu. Następnie wyznaczany jest punkt idealny (wektor najmniejszych wartości dla każdego kryterium) oraz antyidealny (wektor największych wartości) przeskalowanej macierzy. W kolejnym kroku dla każdego   
z punktów niezdominowanych mierzone są odległości od punktu idealnego i antyidealnego przy użyciu metryki euklidesowej. Na podstawie otrzymanych odległości obliczany jest współczynnik topsis z zależności:

, gdzie:

s – współczynnik topsis

– odległość od punktu antyidealnego

– odległość od punktu idealnego.

Ostatnim krokiem jest uszeregowanie punktów niezdominowanych według malejących wartości współczynnika oraz zwrócenie wyników.

## UTA

Metoda UTA jest narzędziem analizy decyzji, która pomaga w podejmowaniu decyzji w przypadku optymalizacji wielokryterialnej. Jest używana do oceny różnych alternatyw w kontekście ich wpływu na cele i preferencje osoby decydującej. Za pomocą oceny każdego z kryterium czy też każdej z alternatyw, jest tworzony ranking z uszeregowanymi wynikami.

Zaimplementowana przez nas metoda UTA, a konkretniej UTA Star składa się z opisanych dalej etapów i elementów. Pierwszym niezbędnym działaniem w tej metodzie jest zidentyfikowanie kryteriów, które są istotne dla decydenta oraz celów decyzyjnych, a więc ich maksymalizacja bądź też minimalizacja. Następnie dla każdego kryterium oraz propozycji rozwiązań (alternatyw) przypisywane są wartości oraz wagi, które odzwierciedlają kolejno stopień spełnienia danego kryterium przez konkretne alternatywy oraz ważność poszczególnych kryteriów w kontekście osiągnięcia celów. Kolejnym krokiem jest dokonanie oceny każdej alternatywy na podstawie przypisanych wartości, które uwzględniają preferencje osoby decydującej oraz wpływ każdej możliwości rozwiązania na maksymalizację bądź minimalizację czyli osiągnięcie celów. Ostatnim działaniem w realizacji metody UTA jest użycie matematycznych operacji, na przykład sumy ważonej do agregacji ocen dla każdej alternatywy, uzyskanie ogólnej oceny oraz końcowo otrzymanie wynikowego rankingu z ułożonymi propozycjami od tych najlepiej spełniających preferencje decydenta, do tych najmniej.

Metoda UTA, według informacji znalezionych w internecie, jest metodą dość często stosowaną, szczególnie użyteczna w sytuacjach z wieloma kryteriami decyzyjnymi.

## Metoda zbiorów referencyjnych

Metoda zbiorów referencyjnych, bądź też metoda zbiorów odniesienia (RSM), to technika analizy decyzji używana w procesie podejmowania decyzji w optymalizacji wielokryterialnej. Podejście to jest oparte na porównywaniu alternatyw względem jakiegoś zdefiniowanego zestawu wzorców, które mają za zadanie reprezentować preferencje osoby podejmującej decyzję w odniesieniu do poszczególnych kryteriów.

Zaimplementowana przez nas metoda RSM składa się z opisanych dalej etapów i elementów. W pierwszym kroku dokonywana jest identyfikacja kryteriów decyzyjnych, które powinny odzwierciedlać cele i priorytety decydenta, a które są istotne dla oceny propozycji końcowych. Następnie dla każdego kryterium tworzone są zbiory referencyjne reprezentujące skrajne wartości tego kryterium. Mogą to być wartości ekstremalne, optymalne lub też inne punkty odniesienia. Kolejno każda alternatywa jest oceniana w kontekście każdego zbioru referencyjnego, a jej ocena przyjmuje formę jej zgodności z utworzonymi zbiorem referencyjnym. Otrzymane wartości ocen są następnie łączone w celu otrzymania ogólnej oceny. Na jej podstawie jest tworzony ranking, z alternatywami uzyskującymi najwyższe wyniki na szczycie.

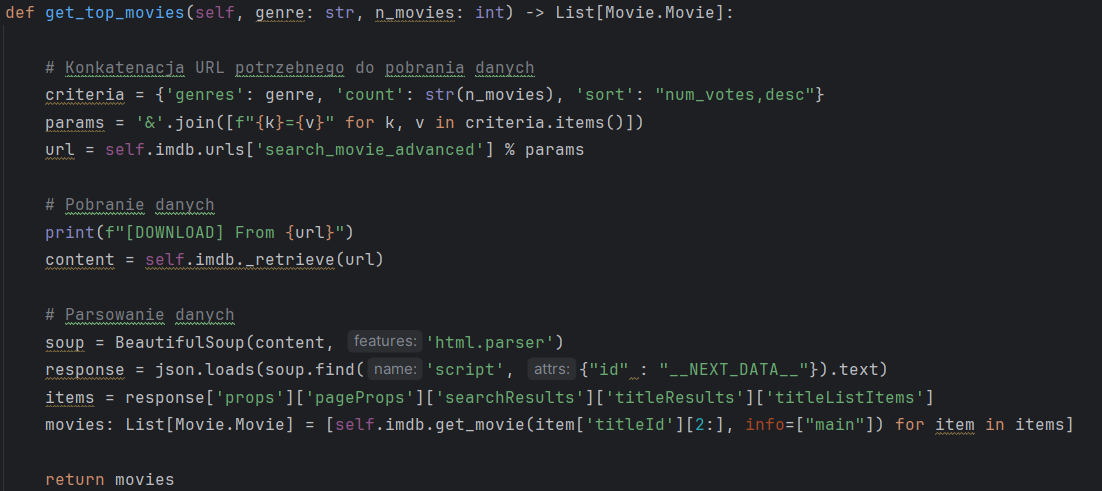
Ta metoda jest szczególnie pomocna w optymalizacji wielokryterialnej w sytuacjach, gdy ocena alternatyw względem skrajnych przypadków jest bardziej intuicyjna i dostępna dla decydenta.

# Aplikacja

Aplikacja została napisana w języku Python wraz z bibliotekami, ponieważ pozwala on na proste tworzenie interfejsów graficznych i łatwy dostęp do bazy danych filmów IMDB.   
W celu jej uruchomienia należy posiadać zainstalowane następujące wersje oprogramowania i bibliotek języka Python:

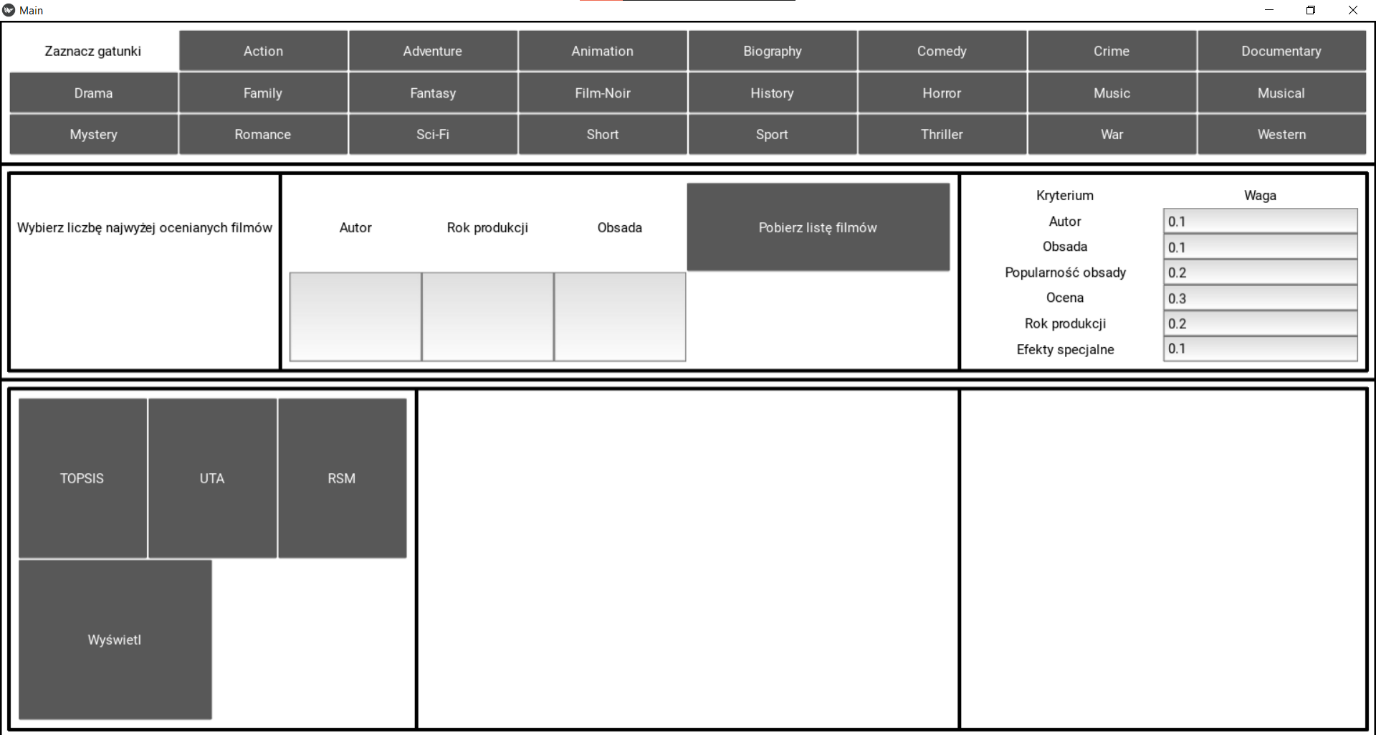
* Python w wersji 3.10,
* IMDbPY w wersji 2022.7.9,
* beautifulsoup4 w wersji 4.12.2,
* matplotlib w wersji 3.6.2,
* pandas w wersji 2.1.2,
* Kivy w wersji 2.1.0,
* Kivy-Garden w wersji 0.1.5.

Dane pobrano, wykorzystując REST API z dedykowanej do filmów biblioteki Pythona IMDbPy, a następnie poddane je parsowaniu, co zostało przedstawione na *Rys. 1.*



Rys. 1. Skrypt do pobrania i parsowania danych

Zbudowany został interfejs graficzny pozwalający na intuicyjne ustawienie i modyfikowanie preferencji w celu znalezienia odpowiedniego filmu. Umożliwia on wybór gatunków,   
z jakich pobierane będą filmy, wpisanie preferowanego autora, roku produkcji i obsady,   
a także ustawienie preferowanych wag dla poszczególnych kryteriów. Po pobraniu listy filmów możliwy jest wybór algorytmu, za pomocą którego będzie rozwiązany problem. Po zakończeniu działania algorytmu wyświetlane są 3 najlepsze filmy zwrócone przez dany algorytm. Możliwe jest także wyświetlenie wykresów przedstawiających wykresy punktowe wartości dla każdego kryterium. Na czerwono zaznaczone są wartości dla najlepszego punktu zwróconego przez wybrany algorytm. Projekt interfejsu graficznego został przedstawiony na *Rys. 2.* oraz *Rys. 3.*



Rys. 2. Interfejs graficzny

Obraz zawierający tekst, zrzut ekranu, diagram, numer

Opis wygenerowany automatycznie

# Eksperymenty

Eksperyment 1:

Arthur Conan Doyle, Harlan Coben, Lee Child, Agatha Christie

Alan Ritchson, Joaquin Phoenix, Robert De Niro

Obraz zawierający zrzut ekranu, design

Opis wygenerowany automatycznie

Topsis:

Obraz zawierający tekst, zrzut ekranu, diagram, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, biały, narzędzie

Opis wygenerowany automatycznie

UTA:

Obraz zawierający tekst, diagram, zrzut ekranu, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, biały, design

Opis wygenerowany automatycznie

RSM:

Obraz zawierający tekst, zrzut ekranu, diagram, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, biały

Opis wygenerowany automatycznie

Eksperyment 2:

Dane:

Obraz zawierający zrzut ekranu, design

Opis wygenerowany automatycznie

TOPSIS:

Obraz zawierający tekst, zrzut ekranu, diagram, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, narzędzie, design

Opis wygenerowany automatycznie

UTA:

Obraz zawierający tekst, diagram, zrzut ekranu, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, biały, narzędzie

Opis wygenerowany automatycznie

RSM:

Obraz zawierający tekst, diagram, zrzut ekranu, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, biały

Opis wygenerowany automatycznie

Eksperyment 3:

Dane:

# Obraz zawierający zrzut ekranu, tekst, design Opis wygenerowany automatycznie

Topsis:

Obraz zawierający tekst, diagram, zrzut ekranu, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, narzędzie, design

Opis wygenerowany automatycznie

UTA:

Obraz zawierający tekst, diagram, zrzut ekranu, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, biały, design

Opis wygenerowany automatycznie

RSM:

Obraz zawierający tekst, diagram, zrzut ekranu, linia

Opis wygenerowany automatycznie

Obraz zawierający tekst, Czcionka, biały

Opis wygenerowany automatycznie

# 6. Podsumowanie/wnioski

## 6.1. Zrealizowane punkty

## 6.2. Napotkane problemy

## 6.3. Kroki dalszego rozwoju

# Spis literatury

<https://www.sciencedirect.com/science/article/pii/S187705091631273X>

<https://www.researchgate.net/publication/236153959_Decision_Support_Systems_Based_on_Reference_Sets>

<https://imdbpy.readthedocs.io/en/latest/>

# Podział pracy

|  |  |  |  |
| --- | --- | --- | --- |
| **Funkcjonalność** | **Artur [wkład w %]** | **Joanna [wkład w %]** | **Adrian [wkład w %]** |
| Preprocessing danych | - | 50 | 50 |
| Wstępne GUI do pobierania filmów | 100 | - | - |