

Wyzwanie:

Zaprojektuj Inteligentnego Asystenta Wydarzeń kulturalnych dla Krakowa



## Opis zadania

Cel zadania: Celem projektu jest stworzenie inteligentnego systemu rekomendującego wydarzenia kulturalne w Krakowie, dopasowane do indywidualnych preferencji użytkownika. Innymi słowy, zadanie polega na zaprojektowaniu asystenta (np. w formie aplikacji webowej, mobilnej lub konwersacyjnej), który pomoże mieszkańcowi lub turyście odkrywać interesujące wydarzenia odbywające się w mieście. System powinien uczyć się preferencji użytkownika i na ich podstawie proponować spersonalizowane rekomendacje – np. koncerty, wystawy, spektakle, spotkania – odpowiadające gustom oraz oczekiwaniom danej osoby.

**Założenia:** Zakładamy, że uczestnicy dysponują wstępną bazą danych wydarzeń kulturalnych z Krakowa. Dane te zostały zebrane z publicznie dostępnych źródeł (scraping) i mogą obejmować nadchodzące koncerty, wydarzenia artystyczne, festiwale, spotkania itp. Profil użytkownika (tj. informacje o jego zainteresowaniach) może być zdefiniowany na potrzeby projektu na kilka sposobów – na przykład przez wybór ulubionych kategorii wydarzeń, analizę historii uczestnictwa w wydarzeniach, lub na podstawie interakcji z samym asystentem (pytania, oceny poprzednich rekomendacji). Należy przyjąć, że im więcej interakcji, tym lepiej system może dopasować propozycje do użytkownika. Ponadto, istotne jest bieżące śledzenie zmian: baza wydarzeń powinna być aktualna (wydarzenia mogą dochodzić w miarę zbliżających się terminów), dlatego dopuszczalne jest i wręcz wskazane wzbogacanie danych poprzez własny web scraping lub wykorzystanie API z aktualnymi informacjami o krakowskich wydarzeniach.



Ograniczenia: Projekt skupia się na wydarzeniach odbywających się w Krakowie – rekomendacje nie muszą uwzględniać wydarzeń spoza miasta. System jest prototypem tworzonym w czasie hackathonu (24 godziny), dlatego rozwiązanie nie musi być komercyjnie doskonałe, ale powinno demonstrować poprawne działanie koncepcji. Uczestnicy powinni skupić się na logice rekomendacji i poprawności działania, zamiast na dopracowaniu interfejsu graficznego czy kompleksowej infrastruktury (prosty interfejs lub nawet tryb konsolowy będzie w pełni akceptowalny, o ile jasno pokaże wyniki). Jednocześnie, ze względu na brak danych o wielu użytkownikach, typowe techniki collaborative filtering mogą być trudne do zaimplementowania – rekomendacje najpewniej oprzecie na analizie treści i profilu pojedynczego użytkownika (content-based).

**Dane wejściowe:** Udostępniamy przykładowy zbiór danych w formatach: .CSV, .ds oraz .json. Zawierając informacje o wydarzeniach. Każdy rekord opisuje jedno wydarzenie kulturalne.



**Proponowane etapy realizacji:** Poniżej przedstawiamy sugerowane kroki, które mogą pomóc w realizacji zadania (nie są one obowiązkowe – możecie zrealizować projekt według własnego planu, ale poniższe wskazówki odzwierciedlają zalecany kierunek):

- Analiza profilu użytkownika Na początku określcie, jakie informacje o użytkowniku będą brane pod uwagę. Czy użytkownik deklaruje swoje zainteresowania (np. wybiera kategorie wydarzeń, które lubi)? A może system uczy się na podstawie jego zachowania (np. historii poprzednich wyszukiwań lub ocen)? Ustalcie format profilu (lista ulubionych kategorii, wektor preferencji, słowa kluczowe itp.) i przygotujcie przykładowy profil, na którym będziecie testować działanie systemu.
- Przygotowanie reprezentacji tekstowej Wykorzystajcie metody NLP do zrozumienia zawartości zarówno profilu użytkownika, jak i opisów wydarzeń. Możecie użyć embeddingów tekstowych: np. wytrenować lub użyć gotowego modelu z biblioteki sentencetransformers do zamiany tekstu (opisów wydarzeń oraz ewentualnie opisu zainteresowań użytkownika) na wektory liczbowe. Taka reprezentacja umożliwi porównywanie "odcisku" znaczenia preferencji z "odciskiem" treści poszczególnych wydarzeń. Upewnijcie się, że tekst (opisy) jest wstępnie przetworzony: oczyszczenie z HTML, usunięcie stop-słów, sprowadzenie do podstawowej formy (lematyzacja) itp., co może poprawić jakość embeddingów.
- Porównywanie preferencji z wydarzeniami Mając wektor reprezentujący profil użytkownika oraz wektory dla każdego wydarzenia, obliczcie podobieństwo między profilem a każdym wydarzeniem. Najprostszą miarą będzie cosine similarity (kosinus podobieństwa) im bliżej 1, tym bardziej zbliżone wektory (a więc bardziej pasujące wydarzenie do profilu). Na tym etapie możecie również uwzględnić dodatkowe kryteria dopasowania: np. filtrować wydarzenia po dacie (nie proponować tych, które już minęły), po lokalizacji (jeśli użytkownik preferuje określoną okolicę) czy po cenie biletów, jeśli takie informacje są dostępne i istotne.

## AI KRAK HACK

• Ranking i rekomendacja – Na podstawie wartości podobieństwa lub innych wybranych kryteriów stwórzcie ranking najbardziej pasujących wydarzeń dla użytkownika. Wybierzcie top-N (np. top-5 czy top-10) wydarzeń do rekomendacji. Przygotujcie sposób prezentacji wyników – może to być lista tytułów z krótkimi opisami i linkami, albo – jeśli tworzycie chatbota – wygenerowana wiadomość zawierająca propozycje (np. "Polecam Ci następujące wydarzenia: ..."). Upewnijcie się, że wyniki mają sens: warto zweryfikować, czy proponowane wydarzenia faktycznie odpowiadają zadeklarowanym zainteresowaniom użytkownika. Jeśli jakieś rekomendacje wydają się niepasujące, zastanówcie się, czy można ulepszyć algorytm (np. dostroić sposób obliczania podobieństwa, dodać wagę dla pewnych cech).

(Powyższe kroki stanowią jedną z możliwych dróg – możecie je modyfikować. Najważniejsze, aby finalny system był w stanie przyjąć pewien profil/wybory użytkownika i na tej podstawie zaproponować mu ciekawe, odpowiadające jego gustom wydarzenia.)