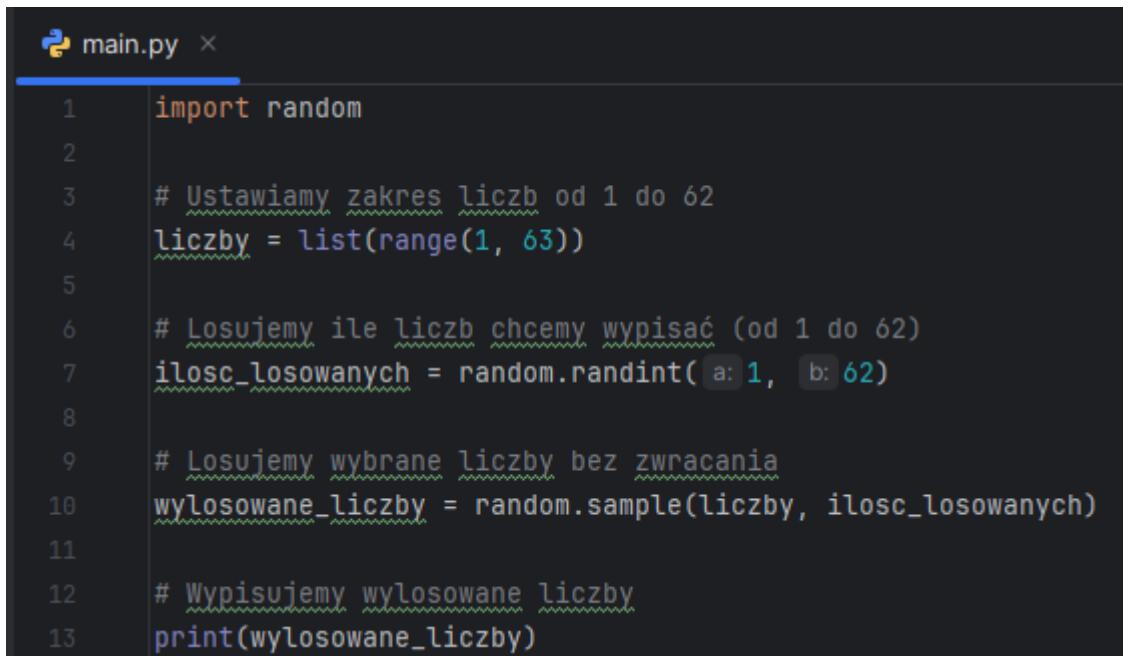


# STATYSTYKI

Statystyki i wyniki zostały zapisane w dokumencie arkusza kalkulacyjnego ( [nai](#) ). Aby zmaksymalizować szanse na otrzymanie poprawnego przekroju działania i skuteczności aplikacji, skorzystaliśmy z losowych inputów - członek zespołu projektowego przygotował szybki randomizer do losowania Interest Input, Trip Duration i POIs Per Day. Pozbywamy się w ten sposób czynnika ludzkiego - na dodatek tego, który ten system opracowuje i zna jego ograniczenia, co pozwala uniknąć podświadomego dopasowywania się do nich.



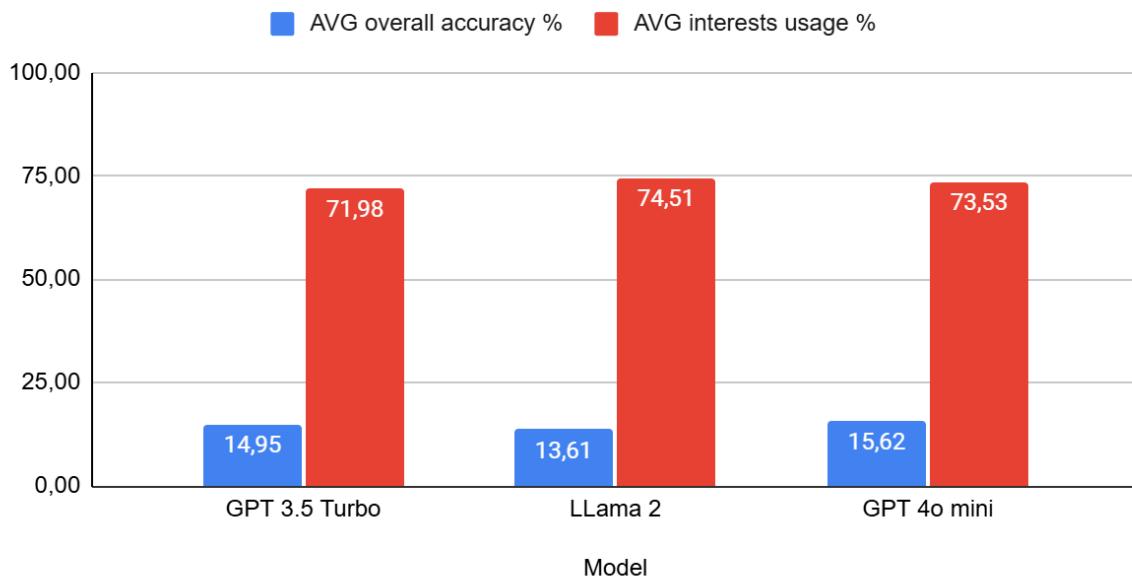
```
main.py ×

1 import random
2
3 # Ustawiamy zakres liczb od 1 do 62
4 liczby = list(range(1, 63))
5
6 # Losujemy ile liczb chcemy wypisać (od 1 do 62)
7 ilosc_losowanych = random.randint(a: 1, b: 62)
8
9 # Losujemy wybrane liczby bez zwracania
10 wylosowane_liczby = random.sample(liczby, ilosc_losowanych)
11
12 # Wypisujemy wylosowane liczby
13 print(wylosowane_liczby)
```

Mimo tego, że wybór dużej ilości zainteresowań (np. dokładnie 56 lub 62 Interests) przez użytkownika jest mało prawdopodobny, to uznaliśmy to za odpowiedni sposób na sprawdzenie systemu i porównanie różnych modeli do przetwarzania języka naturalnego (w tym przypadku gpt-3.5-turbo, gpt-4o-mini oraz Groq Llama). Zdecydowaliśmy się na 20 zbiorów do przetworzenia dla każdego z modeli. Dla ujednolicenia wyników, za każdym razem nasz fikcyjny użytkownik wybiera poszczególne zainteresowania (np. 2, 1, 3, 7), długość wycieczki (w dniach od 1 do 14) i wybiera ile czynności chce wykonywać dziennie (od 1 do 5). Następnie za każdym razem wybieraliśmy wszystkie dostępne regiony (Dublin, Kerry, Galway, Cork, Clare, Donegal, Wicklow) i dla każdego zbioru część parametrów była ustalona na stałe, m.in tempo (slow/moderate/fast)- *moderate*, preferowany typ transportu (car/public/walking) - *car* oraz poziom budżetu (low/moderate/high): *moderate*. Wyбралиśmy te konkretne opcje, w celu ujednolicenia preferencji pojedynczego użytkownika co do transportu, cen i tempa zwiedzania kraju.

Przy większej liczbie zainteresowań użytkownika, skuteczność powinna spadać. Warto zaznaczyć, że przy wyborze niewielu dni, model ma wskazane maksymalnie wykorzystać potencjał POI i powinien wybrać te, które mają najwięcej wspólnych Tagów z zainteresowaniami użytkownika.

## Stosunek AVG overall accuracy % i AVG interests usage % w każdym porównywanym modelu



Interest Matches przypisuje wartość procentową do każdego zainteresowania na bazie tego, ile razy unikalne zainteresowanie pojawi się w wygenerowanym planie podróży.

Overall Accuracy to średnia z Interest Matches. Im wyższe Overall Accuracy tym częściej przypisuje aktywności pasujące do wybranych zainteresowań, a im niższe Overall Accuracy, tym bardziej można określić wycieczkę jako bardziej różnorodną lub dobrane POI miały więcej pasujących Tagów. Jednakże średnia również spada, jeśli jakieś konkretne POI nie znalazło dopasowania do Tagu i ma wartość 0.

Z tego powodu powołujemy się na kolejną statystykę, którą jest Interests Usage - ilość dopasowanych User Interests do Tagów z konkretnego POI. Ta statystyka ma również swoją wartość procentową, zamieszczoną na wykresie.

Wyciągając wnioski, najważniejszą dla nas statystyką jest AVG Interests Usage - średnia z dopasowanych zainteresowań użytkownika do Tagów. Najlepszą skutecznością wykazuje się model LLama 2, osiągając 74,51% AVG Interests Usage. Kolejnym modelem w kolejności do wyboru byłby GPT 4o mini ze względu na zbliżony wynik AVG Interests Usage oraz dość wysoki wskaźnik overall accuracy co może oznaczać że rzadziej generuje plan bez wykorzystania któregoś zainteresowania podanego przez usera.

Idąc tym tokiem rozumowania, można założyć, że LLama 2 jest najlepiej dostosowana do naszego zadania bo najczęściej zawiera w planach podawane przez usera interests. Jednak nie można zapominać, że zrobiliśmy tylko 20 testów (ograniczenia czasowe, finansowe oraz ograniczenia do zapytań). Zakładamy że przy 1000 lub większym zbiorze testowym, wyniki byłyby bardziej zbliżone do siebie. Istnieje spora szansa, że LLama 2 lepiej nadaje się do bardziej losowych i mniej typowych zadań

Dodatkową funkcją projektu jest generowanie zdjęcia (thumbnail) dla każdego podsumowania wycieczki. Korzystamy z modelu DALLE-3 - po eksperymentach z DALLE-2

które wykazały że ten model jest przestarzały zdecydowaliśmy się na nowszą wersję.  
Prompt został opracowany tak aby jak najlepiej oddawać podsumowanie wycieczki i zainteresowania użytkownika.

Przykładowe zdjęcia:





Klucze TODO dla prezentacji:

**OPENAI\_API\_KEY**=sk-proj-HymQJiCD3Gc7OkE168zRwCDmqzsmXjLVGkkBzRrH8ZQy\_fVbYXYYCzGt7ZgufENvM26vLz-FsIT3BlbkFJm1MIZ-HtCizM14CefLYi2J1u1Gv2mxaHnhgNctUPXRXc6mt02hgKzRGKSln0\_8BTbL9Om7GkA

**GROQ\_API\_KEY**=gsk\_TApEKXai3LwmONreZnn8WGdyb3FYq9pesFsBKlu7H2AcneSoGh6P  
**GROQ\_API\_KEY**=gsk\_thG3Jw0wJgE5G3M9KXDHWGdyb3FYwGybJT4tkmWfSLWPjbprAD4c  
**GROQ\_API\_KEY**=gsk\_UTPG5yeLZC5CWxgzLOcNWGdyb3FYDy08xbd0vqoQCmDXoDWuAYP4  
**GROQ\_API\_KEY**=gsk\_dkV8cnw2ER5fVboh7VspWGdyb3FYNnWJ6HMSp0eGj4hIUc8MYmmb