

Wirtualna Informacja Turystyczna

Autorzy: Hanna Jarlaczyńska, Kamil Barszczak

Spis treści

1. Opis	1
2. Algorytm rekomendacji	2
a. Wejście algorytmu	2
b. Wyjście algorytmu	2
c. Kroki algorytmu	2
d. Dane uczące modelu	3
e. Model	3
f. Wyniki	4
3. Pozyskanie danych o atrakcjach	4
4. Aplikacja	5
a. Architektura	5
b. Layout	5
c. Sposób użycia	5
5. Podsumowanie	5

1. Opis

Celem projektu jest stworzenie wirtualnej informacji turystycznej. Aplikacji, która umożliwi użytkownikowi automatyczne tworzenie planu zwiedzania wybranego miasta na podstawie rozmowy z wirtualnym asystentem. Aby stworzyć takiego asystenta wykorzystane zostały najnowsze osiągnięcia z dziedziny AI. W tym szeroko wykorzystywany model transformera opisany w artykule "Attention is all you need". Aplikacja oprócz rekomendacji miejsc, umożliwia również podanie dodatkowych parametrów takich jak czas przeznaczony na zwiedzanie oraz kolejność zwiedzania wskazanych miejsc. Wynikiem działania systemu jest gotowy zestaw miejsc do odwiedzenia, dopasowany do profilu użytkownika, który uwzględnia wszystkie podane parametry wejściowe.

2. Algorytm rekomendacji

a. Wejście algorytmu

Do poprawnego działania algorytm na wejściu wymaga zbioru atrakcji odwiedzanego miejsca oraz opisu aktualnych preferencji użytkownika. Zbiór atrakcji to lista potencjalnych miejsc, w której każde z miejsc opisane jest przy pomocy następujących parametrów:

- adres
- współrzędne geograficzne
- tagi opisujące miejsce
- opis miejsca
- lista opinii użytkowników
 - każdy element tej listy to tekst opinii danego użytkownika

Preferencje użytkownika to tekst wyodrębniony na podstawie rozmowy z aplikacją.

b. Wyjście algorytmu

Wyjściem algorytmu jest posortowana lista atrakcji danego miejsca. Pierwsze miejsce na liście to miejsce, które powinno być odwiedzone w pierwszej kolejności i które najbardziej pasuje do preferencji użytkownika.

c. Kroki algorytmu

Działanie algorytmu oparte jest o model klasyfikatora, który jest zbudowany zgodnie z architekturą transformera opisanego we wcześniej wspomnianym artykule. Rolą tego klasyfikatora jest przypisanie wagi do jednej z ustalonych kategorii: Beaches, Bowling, Climbing, Diving, Fishing, Golf, Paintball, Museums, Cinema, Parking, Barbers, Massage, Perfume, Tattoo, Bakeries, Grocery, Hospitals, Pharmacy, Saunas, Airports, Hostels, Hotels, Tours, Transportation, Bars, Karaoke, Embassy, Libraries, Cafes, Diners, Kebab, Pizza, Noodles, Sandwiches, Seafood, Soup i Steakhouses. Model zwraca 37-elementowy wektor opisujący wagę z zakresu od 0 do 1 dla każdej kategorii. Na podstawie tej wagi oparte jest działanie algorytmu, który można opisać następującymi krokami:

- Obliczenie wag każdej z kategorii dla każdego miejsca z wejścia algorytmu. Dane te mogą być obliczone jednorazowo i przechowane w pamięci tak długo jak nie zostanie zmodyfikowana lista miejsc.
- Obliczenie wag kategorii dla uzyskanych preferencji użytkownika.

- Przypisanie do każdego z miejsc błędu pomiędzy wektorem opisującym dane miejsce a wektorem opisującym preferencje użytkownika. Błąd ten wyznaczony jest jako błąd średniokwadratowy (MSE) opisany wzorem $f(x, y) = \frac{1}{n} \sum (x - y)^2$ gdzie:
 - x - wektor preferencji użytkownika
 - y - wektor opisujący dane miejsce
 - n - ilość kategorii
- Sortowanie miejsc według wyznaczonego błędu zgodnie z porządkiem rosnącym.
- Oddzielenie wartości błędu od każdego z miejsc oraz zwrócenie wyniku.

Po wykonaniu wszystkich powyższych kroków otrzymujemy listę miejsc, które posortowane są według preferencji użytkownika

d. Dane uczące modelu

Dane uczące zostały pozyskane ze zbioru danych YELP, w którym znajdują się recenzje różnego rodzaju atrakcji oraz biznesów. Zbiór ten nie posiada gotowych danych potrzebnych do nauczania modelu tj. tekstów z przypisanymi kategoriami. Aby taki zbiór uzyskać podjęte zostały następujące kroki:

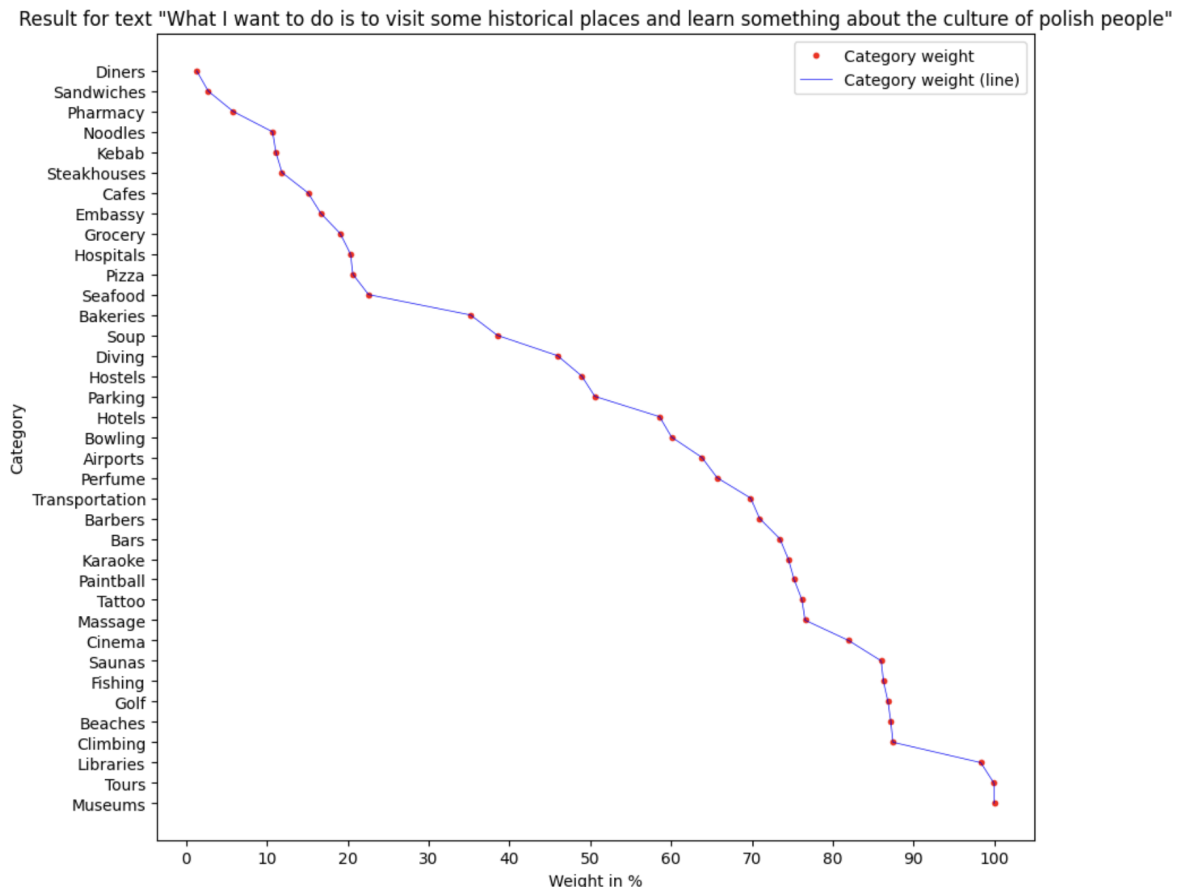
- Dla każdej z recenzji odnaleziony został biznes do którego dana recenzja została przypisana
- Dla każdego biznesu przypisane są kategorie, które dany biznes opisują
- Na podstawie relacji pomiędzy recenzją a biznesem i jego kategoriami utworzony został zbiór tekstów z przypisanymi kategoriami, które ten tekst opisują
- Spośród kilkuset kategorii wybrane zostało 37 kategorii, które najlepiej odzwierciedlają wymagania aplikacji

e. Model

Model użyty do klasyfikacji tekstów to transformer zmodyfikowany tak, aby mógł zostać wykorzystany do opisanego zadania. Jego wejście stanowi sekwencja 500 tokenów opisujących słowa. Wielkość słownika została określona na 20 000 więc model jest w stanie efektywnie rozróżnić do 20 000 różnych słów. Trening został przeprowadzony na zbiorze danych, o którym mowa w punkcie d.

f. Wyniki

Niestety ze względu na niedoskonałość danych treningowych oraz ich względnie niewielką ilość, skuteczność osiągnięta przez model wynosi około 70%. Jednak dla systemu rekomendacji jest to wartość satysfakcjonująca, pozwalająca na dosyć dokładną personalizację trasy zwiedzania. Przykładowe działanie modelu obrazuje poniższy wykres, w którym do każdej z kategorii została przypisana procentowa wartość wagi.



3. Pozyskanie danych o atrakcjach

Pozyskanie odpowiedniej jakości danych, dokładnie opisujących daną atrakcję nie jest zadaniem trywialnym. W aktualnej wersji systemu informacje te pozyskiwane są z API Google Maps. Niestety możliwości korzystania z tych danych są ograniczone a same dane nie są idealne. Odsetek miejsc z wyczerpującym opisem jest niewielki. Aby zaradzić coś na te problemy, wykorzystane zostały wszystkie informacje udostępnione przez google maps w celu stworzenia możliwie dokładnego opisu miejsca.

4. Aplikacja

- a. Architektura
- b. Layout
- c. Sposób użycia

5. Podsumowanie