Semantyczna Wyszukiwarka Gry Wideo

Autorzy: Kamil bajorek i Marcin Kalandyk

Abstrakt:

Projekt polega na stworzeniu semantycznej wyszukiwarki do wybranej kategorii na wikipedii, w naszym przypadku wybraliśmy gry wideo jednakże może to być dowolna wybrana przez nas tematyka. Program pobiera informacje o grach z polskiej wersji językowej Wikipedia, przetwarza dane i umożliwia wyszukiwanie gier na podstawie podanego zapytania.

1. Wstęp

Aktualnie dostęp do informacji jest łatwiejszy niż kiedykolwiek wcześniej, istotne jest posiadanie efektywnych narzędzi do przeszukiwania tych informacji. Problemem, który nasz projekt rozwiązuje, jest brak łatwego dostępu do semantycznego wyszukiwania informacji w Wikipedii. Tradycyjne wyszukiwarki często nie są w stanie zrozumieć kontekstu zapytań, co prowadzi do wyników niezgodnych z oczekiwaniami użytkownika.

2. Rozwinięcie

Nasz projekt składa się z kilku kluczowych komponentów:

Pobieranie danych: Używamy biblioteki wikipediaapi do pobierania danych o wybranej tematyki z polskiej wersji językowej Wikipedia. Pobierane są strony z różnych kategorii związanych daną tematyką, a następnie są przetwarzane i zapisywane w formie JSON.

Przetwarzanie danych: Dane są następnie przetwarzane za pomocą metody preprocess_and_vectorize_data z stworzonego przez nas modułu semantic_search.py. Używamy biblioteki nltk do usunięcia stopwords z języka polskiego, a następnie biblioteki TfidfVectorizer do wektoryzacji danych.

Wyszukiwanie semantyczne: Wyszukiwanie semantyczne odbywa się poprzez porównanie wektora zapytania z wektorami dokumentów za pomocą cosine_similarity * z biblioteki sklearn.

*cosine_similarity to moduł pozwalający na porównanie podobieństwa zapytania i dokumentów za pomocą Cosine similarity, czyli podobieństwa kosinusowego, to miara używana do obliczenia podobieństwa między dwoma wektorami. Jest to metoda stosowana w wielu dziedzinach, w tym analizie tekstu, gdzie wektory reprezentują dokumenty lub frazy.

Podobieństwo kosinusowe mierzy kąt między dwoma wektorami. Jeśli wektory są identyczne, kąt między nimi wynosi 0 stopni, a podobieństwo kosinusowe wynosi 1. Jeśli wektory są całkowicie różne, kąt między nimi wynosi 90 stopni, a podobieństwo kosinusowe wynosi 0.

W kontekście naszego projektu, cosine_similarity z biblioteki sklearn jest używane do porównania wektora zapytania z wektorami dokumentów. Każdy dokument (w tym

przypadku strona z informacjami o grze wideo) jest reprezentowany jako wektor w przestrzeni wielowymiarowej. Zapytanie jest przekształcane w tę samą przestrzeń wektorową. Następnie, podobieństwo kosinusowe jest używane do obliczenia kąta między wektorem zapytania a wektorami dokumentów. Dokumenty są następnie sortowane według ich podobieństwa do zapytania, a te o najwyższym podobieństwie kosinusowym są zwracane jako wyniki wyszukiwania.

W efekcie, cosine_similarity pozwala na semantyczne wyszukiwanie, ponieważ dokumenty, które są "podobne" do zapytania (w sensie, że mają podobne wektory), są uważane za bardziej relewantne.

I**nterfejs użytkownika**: Używamy biblioteki tkinter do stworzenia prostego interfejsu użytkownika, który pozwala na wprowadzanie zapytań i wyświetlanie wyników.

3. Podsumowanie

W naszym projekcie dzięki podobieństwo kosinusowemu sprawdzamy podobieństwo frazy z zapytania z dokumentami z Wikipedii co pozwala sprawdzić ich podobieństwo i na tej podstawie przedstawić wyniki relatywne do zapytania co pozwoliło stworzyć funkcjonalną semantyczną wyszukiwarkę gier wideo. Aplikacja pobiera dane z polskiej wersji językowej Wikipedia, przetwarza je i pozwala na wyszukiwanie na podstawie zapytań. Wyszukiwarka jest w stanie zrozumieć kontekst zapytania i dostarczyć odpowiednie wyniki.

4. Bibliografia

cosine_similarity:
https://fda.readthedocs.io/en/latest/modules/autosummary/skfda.misc.cosine_similarity.html
Semantic Search with Cosine Similarity:
https://owehrens.com/semantic-search-with-cosine-similarity/
Wikipedia API Documentation. https://wikipedia.readthedocs.io/en/latest/code.html#api
Tkinter Documentation. https://docs.python.org/3/library/tkinter.html