

Dokumentacja generatora parafraz tekstów

Opis projektu

Program umożliwia proste generowanie parafraz tekstów według kryteriów wybranych w interfejsie użytkownika. Zaimplementowane są trzy metody zamiany tekstu (zamiana rymów, podmiana synonimów, rozbudowa opisu o dodatkowe przymiotniki). Każdą z tych metod można zastosować na pierwszym, ostatnim, losowym bądź każdym wyrazie w wersie. Użytkownik może dopasować również dokładność oraz różnorodność zmienianych słów. Jako teksty do sparafrazowania użytkownik może wyszukać słowa piosenki lub wygenerować losowy wiersz.

Interfejs

Za obsługę interfejsu odpowiada klasa `Interface` w pliku `interface.py`. Po uruchomieniu programu użytkownikowi wyświetla się główne menu w konsoli, po którym porusza się wpisując cyfry odpowiadające poszczególnym opcjom. Ma do wyboru opcje pomocy, generacji tekstu, generacji parafrazy, zapisania parafrazy do bazy danych, wyszukania parafrazy w bazie danych oraz zamknięcia programu.

1. Wybór pomocy wyświetla dłuższy opis każdej z głównych opcji oraz umożliwia pokazanie szczegółowej pomocy dla dwóch bardziej złożonych opcji.
2. Generacja tekstu pozwala wyszukać tekst piosenki poprzez Genius API podając autora oraz tytuł lub wygenerować losowy wiersz z bazy PoetryDB API.
3. Generacja parafrazy po konfiguracji metody zamiany tekstu, miejsca słowa w wersie oraz dokładności podmiany parafrazuje wygenerowany wcześniej tekst oraz wyświetla go z zaznaczonymi podmienionymi słowami.
4. Kolejna opcja zapisuje parafrazę do bazy danych (słownika, kluczem każdej parafrazy jest jej tytuł).
5. Zapisane parafrazy można wyszukać i wypisać poprzez podanie tytułu oryginalnego tekstu dla odpowiadającej mu parafrazy.
6. Ostatnia opcja kończy program.

Liryka

Za załadowanie pliku konfiguracyjnego odpowiada funkcja `setup_config`, za pobieranie danych z API odpowiada funkcja `get_text`. Klasa `Lyrics` formatuje oraz przechowuje tekst utworu wraz z tytułem oraz autorem. Powyższe funkcje oraz klasa znajdują się w pliku `lyrics.py`.

Piosenka

Za pobranie danych z API Genius oraz przetworzenie ich do odpowiedniego formatu odpowiada klasa `Song` w pliku `song.py`.

Parafraza

Za logikę zamiany tekstu na parafrazę odpowiada klasa `Paraphrase` w pliku `paraphrase.py`.

- Po otrzymaniu w konstruktorze tekstu do przetworzenia oraz specyfikacji zamiany jest on przetwarzany linia po linii.
- Słowo w odpowiednim miejscu wiersza (według parametru `change`) jest wysyłane do metody, która pobiera listę odpowiedzi z Datamuse API dla tego słowa według kryteriów danej metody zamiany tekstu (według parametru `option`). Lista ma długość zadaną przez użytkownika (według parametru `variety`[odpowiedzi w api są posegregowane od najbardziej

do najmniej pasującej, mniejsza liczba odpowiedzi zapewnia zatem większą dokładność podmiany, za to większa liczba zmniejsza ryzyko powtórzeń, ale zmniejsza dokładność parafrazy]).

- Następnie z listy odpowiedzi wybierane jest losowe słowo, korygowana jest jego interpunkcja oraz kolor, tak by wyświetlało się na zielono w konsoli.
- Słowo to zastępuje oryginalne słowo w danej linii dla opcji rymu i synonimu lub “wstawia się” przed oryginalne słowo dla opcji dodania przymiotnika.
- Analogiczny algorytm wykonywany jest dla wszystkich kolejnych linii tekstu.
- W przypadku niezalezienia “matchu” dla zapytania w Datamuse API na swoim miejscu pozostaje oryginalne słowo.

Aby zachować minimalną dokładność parafrazy nawet dla większych różnorodności, podmienianie bądź dodawanie przymiotników do słowa działa dla słów dłuższych niż trzy znaki. Wyklucza to ingerencję w słowa takie jak “I”, “you”, “we”, “he”, “she”, “the” i tym podobne, których naruszenie całkowicie zmieniałoby treść wersu.

Instrukcja użytkowania

Po upewnieniu się, że pliki lyrics.py, paraphrase.py, song.py, interface.py i config.json są w jednym folderze oraz po zainstalowaniu bibliotek z pliku requirements.txt uruchamiamy program pliku interface.py. Następnie obsługujemy program w konsoli. W pliku konfiguracyjnym znajdują się adresy wywołań API, wzorce odpowiedzi dla testów i treść pomocy.

Podsumowanie

Udało się osiągnąć główne założenia projektu:

- Program generuje tekst przez wyszukanie użytkownika bądź losowo
- Użytkownik może sparafrazować wygenerowany tekst według trzech różnych metod, zmieniając pierwsze, ostatnie, losowe lub wszystkie słowa w wersie, dostosowując przy tym różnorodność i dokładność
- Wygenerowane parafrazy z wyróżnionymi na zielono podmienionymi bądź dodanymi słowami mogą być zapisane w bazie danych oraz łatwo w niej wyszukane

Nie udało się niestety znacząco zmniejszyć czasu generowania parafrazy. Chociaż różnica pomiędzy podmianną wszystkich słów a podmianną jednego słowa w wierszu jest bardzo duża to nadal ta druga opcja dla trochę dłuższego tekstu potrafi zajmować kilkanaście sekund. Spowodowane jest to wieloma zapytaniami do API, ciężko więc zoptymalizować tę funkcjonalność.

Program działa jedynie dla tekstów w języku angielskim. API Geniusa obsługuje polski oraz raczej wszystkie inne języki, lecz PoetryDB i co ważniejsze Datamuse są wyłącznie po angielsku. Gdyby owe API dodały kiedyś obsługę innych języków dostosowanie do nich programu polegałoby jedynie na zmianie 2-3 linijek w pliku konfiguracyjnym.

Parafraza to modyfikacja utworu z zachowaniem jego sensu, co ogranicza nieco możliwości ingerencji w tekst. Opisany projekt mógłby jednak z powodzeniem zmieniać tekst w inny sposób, np. na antonimy, homonimy, wyrazy kojarzące się z danym słowem itp. Taka zmiana byłaby kwestią dodania kilku linijek w pliku konfiguracyjnym oraz nazwy nowej opcji. Projekt zachowuje więc zasadę open-close principle.