12 maja 2024 – tokenizacja

https://github.com/jstefaniak99/Tokeny

Pytania dt. Implementacji programu do tokenizacji:

1. Dlaczego wybraliście taką metodę
2. Jak to zaimplementowaliście
3. Jakie klopoty z implementacją zauważacie

Odpowiedzi:

1. Wybór metody dokonałem inspirując się artykułem - „Analiza algorytmów syntezy mowy na potrzeby zastosowania w urządzeniu przenośnym”, w którym została omówiona problematyka Tokenizacji, w tym sposób przechowywania informacji w zdaniach oraz ich reprezentację np.:

1. Przykład tokenizacji dla tekstu “Sklep otwarty od 08:30”:
   <te id=”1” word=”Sklep” type=”wordFirstUppercase” value=”sklep” />
   <te id=”2” word=”otwarty” type=”wordLowercase” value=”otwarty” />
   <te id=”3” word=”od” type=”wordLowercase” value=”od” />
   <te id=”4” word=”08:30” type=”hour” value=”ósmatrzydzieści” />

   W oparciu o ten przykład, postanowiłem zaimplementować kod zgodnie z moimi własnymi preferencjami.
2. Wykorzystałem bibliotekę „re”, która pozwoliła mi na podział zdania na słowa, oraz do wykrywania kiedy zostaje zdanie zakończone i rozpoczyna się nowe. Funkcja „tokenize_text” przetwarza wprowadzony tekst, przypisując każdemu zdaniu token, który jest resetowany po wykryciu końca zdania. Dodatkowo funkcja ta identyfikuje, czy w danym zdaniu występuje liczba, dostosowując odpowiednio typ tokena.
3. Według mnie, głównym problemem jest, że w przypadku kiedy pojawią się w naszym zdaniu liczby, trudno jest je przetworzyć inaczej niż statycznie. To samo tyczy się problemu z wartościami strike liczbowymi. Kolejny problem może być w przypadku dużych zdań, biblioteka regex może wpływać na wydajność.

Jakie kłopoty rozwiązuje mój kod?

Kod może pomóc w analizie struktury tekstu, dzieląc go na zdania i wyodrębniając poszczególne słowa oraz ich typy. Przypisuje do konkretnego słowa Token, dzięki czemu pozwala rozróżniać zdania.