Specyfikacja wymagań systemu do analizy przemówień Wołodymyra Zełenskiego

1. **Wprowadzenie**

System ma na celu automatyzację analizy tekstu przemówień prezydenta Ukrainy Wołodymyra Zełenskiego od 24.02.2022, dostępnych na stronie rządowej. Dzięki niemu możliwa będzie ekstrakcja treści przemówień, ich przetwarzanie tekstowe oraz wizualizacja wyników, co wspomoże badania nad narracją i nastrojami. Projekt realizowany jest w języku R, z podejściem *Reproducible Research*, co oznacza, że wszystkie skrypty i dane są udokumentowane i łatwe do powtórzenia.

2. **Cele systemu**

Celem systemu jest wykonanie kompleksowej analizy treści przemówień Zełenskiego. Obejmuje to:

- **Pozyskanie danych:** automatyczne pobranie lub wczytanie tekstów przemówień z oficjalnej strony prezydenckiej z wybranego okresu czasowego.
- **Przetwarzanie tekstu:** wczytanie i przygotowanie tekstu (tokenizacja, konwersja do małych liter, usunięcie nieistotnych znaków i *stopwords*).
- **Eksploracja częstości:** obliczenie częstości występowania poszczególnych słów oraz wartości TF-IDF.
- **Wizualizacja:** generowanie graficznych prezentacji, takich jak chmury słów (na podstawie TF i TF-IDF).
- **Analiza sentymentu:** wyznaczenie wartości sentymentu całego korpusu przy użyciu słowników Loughran-McDonald, Bing, NRC i Afinn (sumaryczna wartość emocjonalna tekstu).
- **Analiza tematyczna:** zbudowanie modelu LDA w celu wykrycia kluczowych tematów (pięć tematów) wraz z przedstawieniem 10 najważniejszych słów w każdym z nich.
- **Generowanie raportu:** eksport wyników analizy do pliku HTML zawierającego wykresy i podsumowanie.

3. **Wymagania funkcjonalne**

System powinien wykonywać następujące zadania:

- **Pozyskiwanie i wczytywanie danych:** umożliwić pobranie tekstów przemówień z oficjalnej strony (webscraping) oraz wczytanie danych z plików (np. CSV) do środowiska analitycznego. Obsługa kodowania UTF-8 jest wymagana.
- **Przetwarzanie i czyszczenie tekstu:** tokenizować tekst na słowa, konwertować je do małych liter oraz usuwać znaki specjalne, liczby i interpunkcję. Usuwać *stopwords* (z języka angielskiego – treść przemówień była pobierana w języku angielskim.
- **Analiza częstości słów:** zliczać liczbę wystąpień każdego słowa i obliczać TF-IDF. Skrypt powinien umożliwiać obliczenie wartości TF-IDF dla każdego słowa. Wyniki częstości prezentowane będą m.in. w formie chmur słów (osobno dla miar TF i TF-IDF).
- **Analiza sentymentu:** dokonywać oceny sentymentu każdego dokumentu lub całego zbioru z wykorzystaniem wymienionych słowników. System powinien zsumować uzyskane wartości sentymentu i przedstawić rozkład wyników. Dla celów porównania analizy bazuje na kilku słownikach jednocześnie.
- **Modelowanie tematów (LDA):** tworzyć model ukrytej alokacji (LDA) na zbiorze przemówień. System wyodrębni ustaloną liczbę tematów i przedstawi je poprzez podanie najczęściej występujących słów dla każdego tematu.
- **Wizualizacje wyników:** generować wykresy i diagramy przedstawiające wyniki analiz – chmury słów, wykresy słupkowe sentymentu, wykresy zmieniających się nastrojów, a także wizualizacje tematów (np. wykresy słów tematycznych).
- **Eksport raportu:** umożliwiać zapis raportu analizy w formacie HTML zawierającym wszystkie uzyskane wykresy i podsumowania.

Wszystkie powyższe operacje powinny być możliwe do wykonania przez użytkownika poprzez uruchomienie skryptu w R, przy założeniu minimalnej dodatkowej interwencji ze strony użytkownika.

4. Wymagania niefunkcjonalne

- **Reproducibility:** Kod źródłowy (skrypt R) oraz dane wyjściowe muszą być dostępne w formie pozwalającej na łatwe powtórzenie analizy – na platformie GitHub.
- **Wydajność:** Kod powinien być na tyle zoptymalizowany, by analiza pełnego zbioru przemówień trwała jak najmniej.

- **Niezawodność:** System musi działać stabilnie dla różnych zestawów danych wejściowych, obsługiwać błędy pobierania lub niespodziewane formaty stron (logowanie i raportowanie błędów).
- **Łatwość użycia:** Skrypt powinien być prosty do uruchomienia, a wyniki powinny być czytelne i estetyczne.
- **Kompatybilność:** Kod powinien korzystać z popularnych bibliotek R związanych z analizą tekstu i działać niezależnie od systemu operacyjnego (Windows/Linux/Mac).
- **Bezpieczeństwo:** Brak specjalnych wymagań w zakresie bezpieczeństwa (system nie przechowuje poufnych informacji).

5. Interfejsy użytkownika i wymagania dotyczące danych

5.1. **Dane wejściowe:** System przyjmuje dane tekstowe pochodzące z przemówień Wołodymyra Zełenskiego. Mogą to być pliki tekstowe (CSV lub TXT z kolumną *tekst* przemówień) lub adresy URL do stron z tekstem przemówień. Dane powinny być w formacie UTF-8. Przykładami są pliki CSV zawierające kolumny „data, tytuł, tekst”. Użytkownik może także wskazać zakres dat lub numerów przemówień do analizy.

5.2. **Interfejs użytkownika:** Użytkownik uruchamia analizę poprzez uruchomienie skryptu R. Proces może wymagać podania dodatkowych parametrów (np. liczby tematów w modelu LDA).

5.3. **Dane wyjściowe:** System generuje raport w formacie HTML zawierający zebrane statystyki i wykresy. Przykładowo, raport zawiera:

- tabele i wykresy częstości słów,
- chmury słów (wg TF i TF-IDF),
- wykresy słupkowe sentymentu (dla każdego słownika oraz łącznie),
- listę i wizualizację wykrytych tematów LDA (np. wykres słów dla każdego tematu),

6. Słownictwo dokumentacji

- **Korpus:** Zbiór dokumentów (teksty przemówień) używany do analizy.
- **Token:** Pojedyncza jednostka tekstu uzyskana w wyniku tokenizacji (słowo).
- **Stopwords:** Często występujące słowa (np. spójniki, zaimki), które są usuwane przed analizą ze względu na brak znaczenia statystycznego.
- **Chmura słów:** Graficzna prezentacja najczęściej występujących słów w korpusie – im większa czcionka słowa, tym częściej się ono pojawia.
- **TF (Term Frequency):** Częstość wystąpień słowa w dokumencie.
- **TF-IDF (Term Frequency–Inverse Document Frequency):** Miara ważności słowa w korpusie – łączy częstość występowania z wagą odwrotności występowania w innych dokumentach.
- **Analiza sentymentu:** Proces przypisywania wartości emocjonalnej (pozytywnej/negatywnej) do słów i podsumowania ich w całościowy wskaźnik nastroju. *Sentyment* oznacza ogólny ton emocjonalny tekstu, a *słownik sentymentów* to lista słów z przypisanymi wartościami emocjonalnymi (np. Loughran, Bing).
- **Model LDA (Latent Dirichlet Allocation):** Nienadzorowany algorytm uczenia maszynowego służący do identyfikacji tematów w zbiorze dokumentów.
- **Temat:** W kontekście LDA – zestaw słów często współwystępujących w podzbiorze dokumentów; LDA przypisuje dokumentom mieszanki takich tematów.

7. Przypadki użycia (use cases)

a) Wczytanie danych:

- *Aktor:* Użytkownik systemu.
- *Opis:* Użytkownik inicjuje proces pobrania lub wczytania plików z tekstami przemówień. System łączy się ze stroną rządową, pobiera dostępne transkrypcje lub odczytuje lokalnie przygotowane pliki CSV/TXT.
- *Warunek początkowy:* Użytkownik ma dostęp do internetu i podstawowe dane (lub plik) z przemówieniami.
- *Wynik:* Tekst przemówień został wczytany do pamięci systemu w postaci korpusu.

b) Analiza danych:

- *Aktor:* Użytkownik systemu.
- *Opis:* Użytkownik uruchamia skrypt analityczny. System wykonuje przetwarzanie (tokenizacja, czyszczenie), analizuje częstość słów, oblicza TF-IDF, przeprowadza analizę sentymentu (z użyciem słowników) i modeluje tematy LDA. Generuje przy tym wykresy (chmury słów, słupki sentymentu, słowa tematyczne).
- *Warunek początkowy:* Dane tekstowe zostały wczytane.
- *Wynik:* System przygotowuje wewnętrzne wyniki analizy – tabele częstotliwości, wartości sentymentów, strukturę tematów.

c) Generowanie raportu:

- *Aktor:* Użytkownik systemu.
- *Opis:* Użytkownik żąda wygenerowania raportu z wynikami analizy. System eksportuje dane i wykresy do pliku HTML.
- *Warunek początkowy:* Analiza została zakończona, a wyniki są gotowe.
- *Wynik:* Pojawia się raport HTML z podsumowaniem analizy (wykresy, tabele, interpretacje).

Schemat użycia (podział ról system/użytkownik): System realizuje kroki analizy: „Tworzy korpus tekstów”, „Tokenizuje dane i czyści tekst”, „Zlicza częstość słów”, „Tworzy chmury słów”, „Modeluje tematy” itp., a użytkownik uruchamia skrypt, wybiera parametry (np. liczbę tematów) i odbiera wyniki.

8. Scenariusze użytkownika (user stories)

- *Jako analityk polityczny* chcę szybko przeprowadzić analizę sentymentu i tematów w przemówieniach Wołodymyra Zełenskiego od lutego 2022, aby zidentyfikować główne motywy oraz zmiany nastrojów w przemówieniach ukraińskiego przywódcy.
- *Jako badacz komunikacji społecznej* chcę zobaczyć, jak zmienia się ton wypowiedzi prezydenta Ukrainy w czasie konfliktu, aby lepiej zrozumieć przekazy rządowe i ich wpływ na opinię publiczną.

Każdy scenariusz zakłada, że użytkownik dysponuje gotowymi do analizy danymi (listą przemówień) lub może je pobrać za pomocą systemu, a następnie otrzymuje klarowny raport wyników (wykresy nastrojów, słowa kluczowe tematów itp.).