Dokumentacja Specyfikacji Wymagań (SRS)

Projekt: Analiza feedbacku ChatGPT (Text Mining i analiza sentymentu)
Wersja dokumentu: 1.0
Data: 10.05.2025
Autorzy: Patrycja Rutkowska, Anastasiya Albatava, Szymon Kiczyński

1. Wprowadzenie

Niniejszy dokument opisuje specyfikację wymagań dla skryptu R realizującego kompleksową analizę tekstu i ocenę sentymentu na podstawie opinii użytkowników modelu językowego ChatGPT. Projekt realizowany jest w ramach przedmiotu Projektowanie Systemów Informatycznych na Wydziale Nauk Ekonomicznych. System integruje podstawowe techniki przetwarzania języka naturalnego, takie jak czyszczenie danych, tokenizacja, usuwanie stopwords i stemming. W analizie wykorzystano wizualizacje częstości słów (chmury słów) oraz analizę sentymentu opartą na słownikach Loughran, NRC, Bing i Afinn (w plikach CSV) oraz GI, HE, LM, QDAP z pakietu SentimentAnalysis. Wyniki prezentowane są w formie chmur słów, wykresów rodzaju sentymentu na podstawie różnych słowników oraz wykresów przedstawiających ewolucję sentymentu w czasie. Cała analiza została ujęta w raport HTML zgodny z zasadami Reproducible Research.

2. Cele systemu

- Wczytywanie danych wejściowych z pliku .csv;
- Przetwarzanie i czyszczenie tekstu (normalizacja, tokenizacja, stemming i stemCompletion, usuwanie stopwords);
- Identyfikacja i zliczanie częstości występowania słów;
- Wizualizacja dominujących słów w postaci chmury słów;
- Przeprowadzenie analizy sentymentu z użyciem słowników:
    - W plikach CSV: Loughran, NRC, Bing i Afinn;
    - Wbudowanych w pakiet SentimentAnalysis (GI, HE, LM, QDAP);
- Wizualizacja wyników sentymentu za pomocą wykresów słupkowych i czasowych;
- Porównanie sentymentu na podstawie różnych słowników;
- Umożliwienie analizy zmian sentymentu z różnych słowników w czasie;
- Przygotowanie wyników w formacie umożliwiającym wygenerowanie końcowego raportu HTML;
- Udostępnienie kodu źródłowego i danych w sposób umożliwiający pełną odtwarzalność uzyskanych wyników.

3. Wymagania funkcjonalne

- **Wczytywanie danych:**
  - Skrypt powinien umożliwiać wczytanie opinii użytkowników z lokalnego pliku .csv oraz poprawnie obsługiwać kodowanie UTF-8.
- **Przetwarzanie i oczyszczanie tekstu:**
  - Skrypt powinien ujednolicać format apostrofów do formy klasycznej ';
  - Skrypt powinien usuwać liczby, znaki interpunkcyjne, skróty z apostrofami, puste tokeny inne zbędne znaki;
  - Skrypt powinien filtrować słowa bez wartości informacyjnej przy użyciu zbiorów stopwords z pakietów tidytext i tm;
  - Skrypt powinien wykonywać stemming oraz przywracać oryginalne formy słów (stemCompletion).
- **Analiza częstości słów:**
  - Skrypt powinien liczyć liczbę wystąpień słów i sortować je według częstości;
  - Skrypt powinien przedstawiać wyniki zarówno tabelarycznie, jak i w postaci graficznej (chmura słów).
- **Wizualizacja chmury słów:**
  - Skrypt powinien umożliwiać utworzenie chmury słów dla najczęściej występujących terminów z określonym progiem częstości.
- **Sentyment na podstawie słowników CSV:**
  - Skrypt powinien umożliwiać analizę sentymentu przy użyciu zewnętrznych słowników (Afinn, Bing, NRC, Loughran) zapisanych w formacie .csv;
  - Skrypt powinien dopasowywać słowa do słowników i zliczać kategorie sentymentu;
  - Skrypt powinien umożliwiać filtrowanie słów pozytywnych i negatywnych.
- **Analiza sentymentu z pakietu SentimentAnalysis:**
  - Skrypt powinien analizować sentyment tekstu z wykorzystaniem słowników GI, HE, LM, QDAP;
  - Skrypt powinien konwertować wyniki sentymentu na wartości kierunkowe;
  - Skrypt powinien umożliwiać podział tekstu na segmenty o stałej długości znaków.
- **Wizualizacja wyników analizy sentymenty:**
  - Skrypt powinien tworzyć wykresy z użyciem ggplot2;
  - Skrypt powinien prezentować skumulowane wyniki sentymentu oddzielnie dla każdego słownika;
  - Skrypt powinien umożliwiać porównanie słowników na wspólnym wykresie;
  - Skrypt powinien tworzyć wykresy liniowe zmian sentymentu w czasie.
- **Agregacja danych:**
  - Skrypt powinien łączyć dane sentymentu z różnych słowników w jedną ramkę danych;
  - Skrypt powinien identyfikować i usuwać wartości brakujące (NA) powstające np. przy niedopasowaniu słów do słowników.

- **Generowanie raportu:**
  - Skrypt powinien umożliwiać użytkownikowi wygenerowanie raportu HTML zawierającego kod, wizualizacje i wyniki analizy.
- **Odtwarzalność analizy:**
  - Skrypt powinien zapewniać pełną odtwarzalność wyników poprzez udostępnienie kodu źródłowego i danych wejściowych w niezmienionej formie.

4. Wymagania niefunkcjonalne

- **Wydajność:**
  - Czas przetwarzania i analizy dla dostarczonego zbioru danych powinien być rozsądny (np. nie dłuższy niż kilkanaście minut), umożliwiając swobodne eksperymentowanie z parametrami (np. wyborem słowników lub minimalną częstością słów w chmurze słów).
- **Niezawodność:**
  - Skrypt poprawnie przetwarza dane tekstowe zawierające zbędne znaki, puste tokeny oraz typowe błędy formatowania (np. znaki specjalne, apostrofy, interpunkcja).
- **Użyteczność:**
  - Skrypt powinien generować przejrzyste wykresy z użyciem ggplot2;
  - Chmura słów powinna być wizualizowana z użyciem palety RColorBrewer;
  - Obsługa kodu powinna być prosta dla użytkownika pracującego w środowisku RStudio.
- **Łatwość utrzymania:**
  - Kod powinien być modularny i podzielony logicznie na etapy analizy (np. wczytanie danych, przetwarzanie, analiza sentymentu);
  - Kluczowe fragmenty kodu powinny być opatrzone komentarzami ułatwiającymi zrozumienie i rozwój projektu.
- **Kompatybilność:**
  - Skrypt powinien być zgodny z R w wersji 4.0 lub nowszej;
  - Powinien wykorzystywać popularne biblioteki: tm, tidytext, stringr, wordcloud, ggplot2, ggthemes, SentimentAnalysis, SnowballC, tidyverse, RColorBrewer, stringi.

5. Interfejsy użytkownika i dane

Wejście:

- Plik .csv zawierający teksty opinii oraz dane dodatkowe;
- Słowniki sentymentów w formacie .csv (AFINN, Bing, NRC, Loughran) lub wbudowane w pakiet SentimentAnalysis (GI, HE, LM, QDAP).

Wyjście:

- Tabela częstości słów
- Chmura słów (wordcloud)

- Wykresy słupkowe i liniowe przedstawiające sentyment;
- Ogółem, wyniki prezentowane są w formie wizualnej i tabelarycznej — użytkownik może zapisać je jako raport HTML.

Wymagania dotyczące danych:

- Skrypt zakłada, że dane tekstowe są w języku angielskim;
- Dane wejściowe muszą być zapisane w pliku .csv z kodowaniem UTF-8;
- Plik wejściowy powinien zawierać jedną kolumnę tekstową zawierającą opinie lub więcej, jeśli dostępne są metadane (np. data, źródło). Inne kolumny nie są wymagane, ale mogą być użyteczne w rozszerzonej analizie.
- Treści tekstowe mogą zawierać znaki specjalne, interpunkcję, cyfry i inne zakłócenia — system powinien być przygotowany na ich usuwanie. Nie wymagane jest wcześniejsze ręczne czyszczenie danych.
- Skrypt wykorzystuje słowniki sentymentów dostępne w plikach .csv oraz w pakiecie SentimentAnalysis;
- Pliki tekstowe (np. .txt) są przetwarzane tylko pomocniczo, jeśli zostały wcześniej wczytane;
- System powinien poprawnie obsługiwać pliki zawierające od kilkuset do kilku tysięcy opinii.

6. Słownictwo dokumentacji

- **Token** – pojedynczy element tekstu, uzyskany po podziale tekstu.
- **Stopwords** – słowa pozbawione istotnej wartości informacyjnej.
- **Sentyment** – ogólny ładunek emocjonalny wyrażony w tekście.
- **Skumulowany sentyment** – suma ocen sentymentu dla całego tekstu.
- **Wartości kierunkowe** – konwersja ciągłych wartości sentymentu na kategorie (np. pozytywny, negatywny, neutralny).
- **Ewolucja sentymentu** – zmiana sentymentu wzdłuż czasu narracyjnego.
- **Stemming** – proces redukcji słów do ich rdzeni.
- **StemCompletion** – uzupełnienie przyciętych słów po stemmingu na podstawie najczęstszej formy w zbiorze danych.
- **Chmura słów** – graficzna reprezentacja najczęściej występujących słów w tekście, gdzie rozmiar czcionki odpowiada częstości.
- **Słowniki sentymentu** – zbiory słów przypisanych do emocji lub biegunowości (np. pozytywny/negatywny), używane do automatycznej analizy sentymentu.
- **Segmentacja tekstu** – podział tekstu na fragmenty o określonej długości (np. co 1000 znaków), w celu analizy zmian sentymentu w czasie.
- **Reproducible Research** – zasada zapewniająca, że analiza może być powtórzona z tymi samymi wynikami dzięki pełnej dokumentacji kodu i danych.

- **UTF-8** – standard kodowania znaków używany do prawidłowego odczytu tekstów zawierających różne znaki i symbole.

7. Przypadki użycia (Use Cases)

- Użytkownik:
  - Wybiera plik .csv zawierający rzeczywiste opinie użytkowników ChatGPT;
  - Uruchamia skrypt analizujący treść i sentyment wypowiedzi;
  - Przegląda wyniki analizy w postaci wykresów i chmury słów;
  - Generuje raport HTML z wynikami, który może wykorzystać do prezentacji lub dalszych wniosków.
- Skrypt/system*:*
  - Wczytuje dane tekstowe z pliku .csv zakodowanego w UTF-8;
  - Czyści i przekształca tekst;
  - Dopasowuje słowa do słowników sentymentu;
  - Generuje chmurę słów;
  - Generuje wykresy sentymentu;
  - Generuje wykres porównujący rodzaj sentymentu wg słowników;
  - Segmentuje tekst i pokazuje, jak sentyment zmienia się w czasie;
  - Umożliwia generowania podsumowującego raportu HTML z kodem i wynikami.

Testowe przypadki użycia:

- Test pozytywny: plik .csv z wypowiedziami użytkowników, w których dominują słowa o pozytywnym wydźwięku (np. „amazing”, „helpful”, „love”);
- Test negatywny: plik zawierający skargi i negatywne emocje (np. „frustrated”, „bad”, „waste”);
- Test „pusty sentyment”: dane, w których słowa nie występują w żadnym słowniku sentymentu — analiza nie wykrywa emocji;
- Test mieszany: opinie zawierające zarówno pochwały, jak i krytykę — sprawdzana jest równowaga sentymentów;
- Test braków: plik zawierający niepełne dane tekstowe lub puste pola;
- Test odporności: opinie zawierające nietypowe znaki, liczby, skróty i błędy językowe — sprawdzana jest odporność przetwarzania.

8. Scenariusze użytkownika (User Stories)

Scenariusz 1: Identyfikacja najczęściej występujących słów w opiniach

- **Jako:** Menedżer produktu
- **Chcę:** przeanalizować teksty opinii, aby zobaczyć, które tematy, cechy lub funkcje ChatGPT są najczęściej wspominane przez użytkowników

- **Aby:** zrozumieć, które obszary narzędzia budzą największe zainteresowanie lub powtarzają się najczęściej w kontekście użytkowania

Kryteria akceptacji:

- Użytkownik może wczytać plik .csv z opiniami użytkowników w języku angielskim.
- Skrypt przetwarza tekst: usuwa zakłócenia, wykonuje tokenizację, stemming i filtruje nieistotne słowa (stopwords).
- Tworzona jest tabela częstości oraz chmura słów, która wizualnie przedstawia najczęściej występujące słowa.
- Użytkownik może szybko ocenić, które pojęcia dominują w opiniach i zapisać wyniki do raportu HTML.

Scenariusz 2: Analiza sentymentu opinii użytkowników

- **Jako:** Analityk danych
- **Chcę:** określić, czy opinie użytkowników na temat ChatGPT są pozytywne, negatywne, czy nie wykazują wyraźnego sentymentu
- **Aby:** dostarczyć zespołowi produktowemu i marketingowemu mierzalną informację o nastrojach klientów

Kryteria akceptacji:

- Skrypt wczytuje dane z pliku .csv i przetwarza tekst przy użyciu technik czyszczenia i normalizacji.
- Wykonywana jest analiza sentymentu z wykorzystaniem słowników w plikach .csv (Afinn, Bing, NRC, Loughran) oraz pakietu SentimentAnalysis (GI, HE, LM, QDAP).
- Wyniki prezentowane są w formie wykresów słupkowych — osobno dla każdego słownika.
- Użytkownik może porównać rozkład sentymentu w zależności od zastosowanego słownika oraz zapisać wizualizacje w raporcie HTML.

Scenariusz 3: Śledzenie zmian sentymentu w czasie

- **Jako:** Badacz UX lub komunikacji
- **Chcę:** prześledzić, jak zmienia się nastrój w długich ciągach wypowiedzi, aby wykryć momenty wzmożonego entuzjazmu lub frustracji
- **Aby:** lepiej zrozumieć narrację użytkownika i jej wpływ na emocjonalny odbiór

Kryteria akceptacji:

- Skrypt łączy cały przetworzony tekst w jeden ciąg i dzieli go na segmenty (np. co 1000 znaków).

- Dla każdego segmentu liczony jest sentyment przy użyciu funkcji analyzeSentiment() z pakietu SentimentAnalysis.
- Tworzony jest wykres liniowy pokazujący ewolucję sentymentu w czasie.
- Użytkownik może zidentyfikować fragmenty tekstu, w których występują największe wahania emocjonalne, i podjąć dalszą analizę jakościową.