Anotacja korpusów oraz osadzenia słów i tekstów
Część II: Modele przestrzeni wektorowych

Autorzy: Oliwer Krupa, Adam Bednarski, Jan Masłowski, Łukasz Lenkiewicz

October 23, 2024

Contents

1 Word Embeddings

1.1 Osadzenia słów w anotowanym korpusie (Word2Vec i Fasttext)

W celu zbadania i wizualizacji osadzeń słów w anotowanym korpusie, zastosowaliśmy dwie metody: **Word2Vec** oraz **FastText**. Dla obu modeli osadziliśmy słowa, a następnie zwizualizowaliśmy wyniki za pomocą algorytmu t-SNE, który redukuje wymiary danych, co pozwala na łatwiejszą analizę i porównanie rezultatów.

1.1.1 Word2Vec

Model **Word2Vec** został przetrenowany na korpusie tekstów polskich, a następnie osadziliśmy słowa z anotowanego korpusu, aby sprawdzić, jak różne słowa są reprezentowane w przestrzeni wektorowej. Wyniki zostały przedstawione na poniższym wykresie, gdzie kolory reprezentują różne kategorie sentymentu — niektóre słowa zostały oznaczone jako "Brak etykiety", a inne jako "Wzmacnianie".

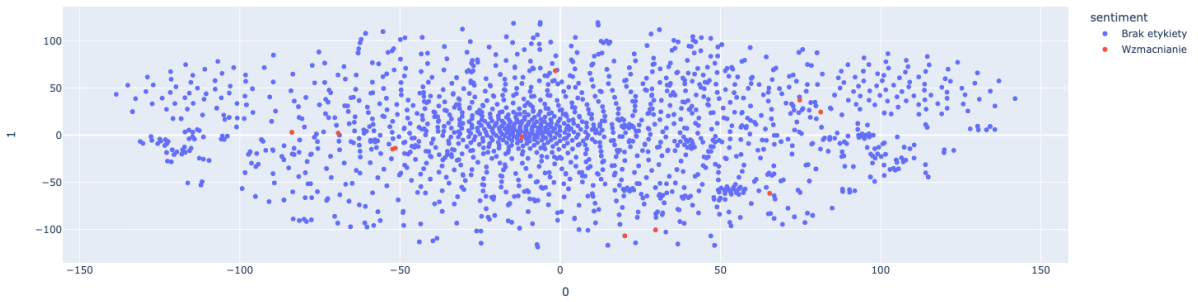

Figure 1: Wizualizacja osadzeń słów za pomocą Word2Vec

Jak widać na rysunku 1, większość słów jest zgrupowana w centralnej części wykresu, jednak niektóre słowa przypisane do kategorii "Wzmacnianie" znajdują się różnych obrszarach przestrzeni wektorowej.

1.1.2 FastText

Podobną analizę przeprowadzono z użyciem modelu **FastText**, który również został przetrenowany na korpusie polskim. Wyniki osadzeń słów z tego modelu zostały przedstawione na poniższym wykresie, który podobnie jak poprzedni, został wygenerowany za pomocą t-SNE.

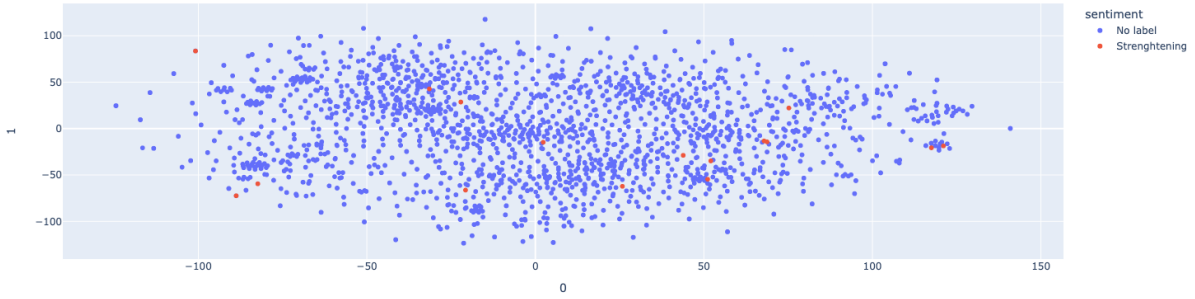

Figure 2: Wizualizacja osadzeń słów za pomocą FastText

Rysunek 2 pokazuje, że wyniki dla modelu FastText są podobne do tych uzyskanych z Word2Vec, chociaż niektóre grupy słów znajdują się w różnych częściach przestrzeni, co wynika z różnic w sposobie reprezentacji słów przez te dwa modele.

1.2 Porównanie k-najbardziej podobnych słów dla modeli Word2Vec i FastText

W ramach analizy wygenerowano listy pięciu najbardziej podobnych słów dla zaanotowanych terminów, wykorzystując modele **Word2Vec** oraz **FastText**. Wyniki pokazują różnice w sposobie modelowania podobieństw między słowami.

1.2.1 Wyniki dla Word2Vec

Dla modelu **Word2Vec**, lista najbardziej podobnych słów zawierała m.in.:

- **by**: aby (0.89), żeby (0.86), więc (0.69)
- **kto**: dlaczego (0.66), czemu (0.63), jeśli (0.59)
- **człowiek**: osobnik (0.64), osoba (0.64), tubylec (0.62)
- **seks**: sexu (0.76), masturbacja (0.71), erotyka (0.63)
- **fakt**: stwierdzenie (0.66), twierdzenie (0.65), hipoteza (0.59)

1.2.2 Wyniki dla FastText

Dla modelu **FastText**, podobne słowa to:

- **by**: By (0.77), żeby (0.61), muc (0.60)
- **kto**: Kto (0.80), ktoś (0.78), ktokolwiek (0.72)
- **człowiek**: Człowiek (0.80), czlowiek (0.76), człowiek. (0.69)
- **seks**: sex (0.80), seks. (0.78), seks- (0.76)
- **fakt**: faktem (0.68), faktu (0.66), Fakt (0.65)

1.2.3 Dyskusja wyników

Oba modele pokazują różnice w sposobie reprezentacji podobieństw. **FastText** lepiej radzi sobie z różnymi formami morfologicznymi i fleksyjnymi, np. dla słowa „człowiek" uwzględnia różne jego odmiany. Z kolei **Word2Vec** ma tendencję do generowania bardziej ogólnych podobieństw, co sprawia, że jest mniej precyzyjny w przypadku rzadkich wyrażeń. FastText dzięki n-gramom lepiej odwzorowuje złożoność morfologiczną języka polskiego.

2 Text Embeddings

3 Osadzenia zdań za pomocą dwóch różnych modeli i wizualizacja t-SNE

W ramach tej części eksperymentu wygenerowano osadzenia zdań z anotowanego korpusu, wykorzystując dwa różne modele: **TF-IDF** oraz **BERT**. Następnie wyniki osadzeń zostały zwizualizowane za pomocą algorytmu **t-SNE**, który redukuje wymiary danych, ułatwiając analizę i porównanie osadzeń zdań.

3.1 TF-IDF

Model **TF-IDF** (ang. Term Frequency-Inverse Document Frequency) mierzy częstość występowania słowa w danym dokumencie w stosunku do jego częstości w całym korpusie. Jest to popularny model bazujący na statystycznym podejściu, który nadaje większe wagi słowom występującym rzadziej w całym korpusie, co pozwala na lepsze odwzorowanie kluczowych informacji w osadzeniach tekstów.

Wizualizacja uzyskanych osadzeń za pomocą algorytmu t-SNE dla modelu TF-IDF jest przedstawiona na Rysunku 3. Kolory punktów odpowiadają różnym kategoriom sentymentu: "Mowa nienawiści" (kolor czerwony) oraz "Neutralny" (kolor niebieski). Każdy punkt na wykresie reprezentuje jedno zdanie z korpusu, a po najechaniu na punkt można odczytać treść zdania. Poza teśćią można zauważyć 'sentiment' czyli etykietę sentymentu przypisaną do danego zdania.

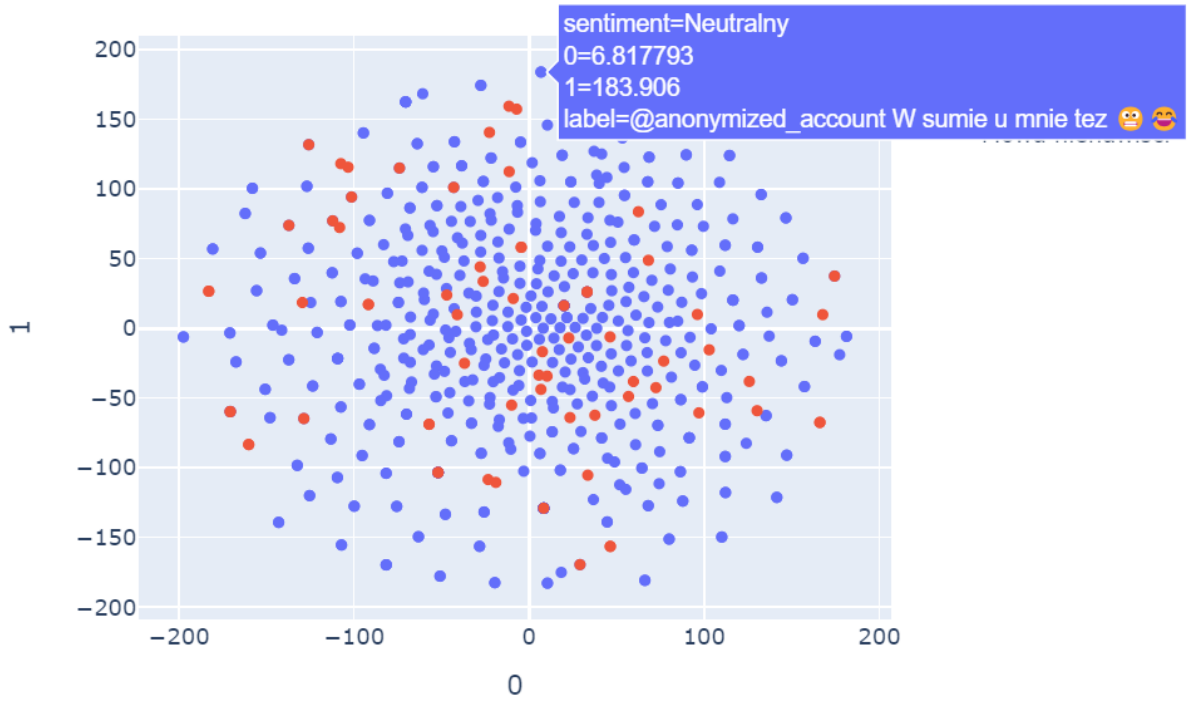

Figure 3: Wizualizacja osadzeń zdań za pomocą modelu TF-IDF przy użyciu t-SNE

3.2 BERT

Model **BERT** (ang. Bidirectional Encoder Representations from Transformers) jest jednym z najnowocześniejszych modeli do przetwarzania języka naturalnego, wykorzystującym głębokie sieci neuronowe typu transformer. BERT jest w stanie generować osadzenia zdań z uwzględnieniem kontekstu zarówno przed, jak i po danym słowie, co czyni go bardziej skutecznym w modelowaniu złożonych relacji semantycznych w tekście.

Rysunek 4 przedstawia wyniki osadzeń zdań wygenerowanych przy użyciu modelu BERT, zwizualizowane za pomocą t-SNE. Podobnie jak w przypadku TF-IDF, kolory odpowiadają różnym kategoriom sentymentu, a każdy punkt reprezentuje jedno zdanie.

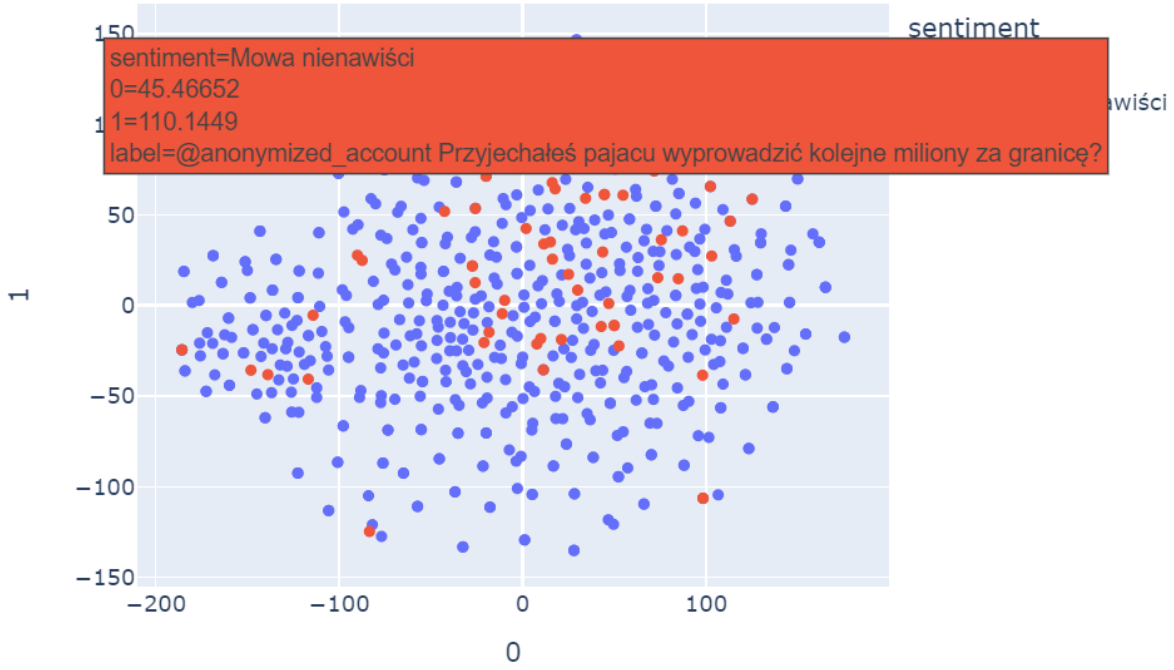

Figure 4: Wizualizacja osadzeń zdań za pomocą modelu BERT przy użyciu t-SNE

3.3 Porównanie modeli

Wyniki wizualizacji t-SNE dla modeli **TF-IDF** oraz **BERT** pokazują różne podejścia do modelowania osadzeń zdań. TF-IDF, bazując na częstościach słów, generuje osadzenia w sposób bardziej statystyczny, natomiast BERT, dzięki swoim zaawansowanym mechanizmom modelowania kontekstu, pozwala na lepsze uchwycenie semantycznych relacji między zdaniami. W obu przypadkach obserwujemy, że zdania oznaczone jako "Mowa nienawiści" są grupowane w określonych regionach przestrzeni, co może sugerować charakterystyczne cechy językowe używane w takich wypowiedziach.

3.4 Klasteryzacja osadzeń anotowanych zdań

W tej części zadania przeprowadziliśmy klasteryzację osadzeń anotowanych zdań uzyskanych z dwóch różnych modeli: TF-IDF oraz BERT. Do klasteryzacji zastosowano dwa różne algorytmy: **KMeans** oraz **HDBSCAN**. Wyniki klasteryzacji zostały przedstawione na wykresach t-SNE, które redukują wymiary wektorów osadzeń, umożliwiając wizualizację wyników w przestrzeni dwuwymiarowej.

3.4.1 Klasteryzacja za pomocą KMeans

Na pierwszym wykresie (Rysunek 5) przedstawiono wyniki klasteryzacji przy użyciu algorytmu KMeans dla osadzeń uzyskanych z modelu TF-IDF. Widać wyraźne grupowanie punktów odpowiadających różnym klastrom.

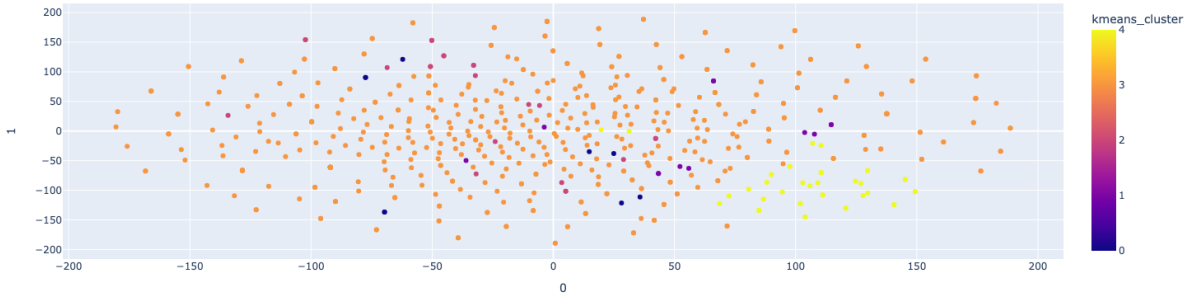

Figure 5: Klasteryzacja TF-IDF z użyciem algorytmu KMeans

Podobną analizę przeprowadzono dla modelu BERT (Rysunek 6), gdzie również widzimy wyraźne grupowanie zdań.

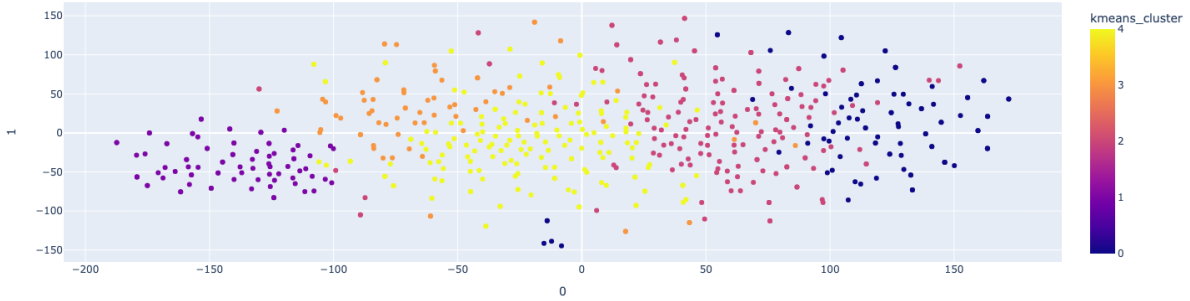

Figure 6: Klasteryzacja BERT z użyciem algorytmu KMeans

3.4.2 Klasteryzacja za pomocą HDBSCAN

Dla porównania zastosowano również algorytm HDBSCAN, który jest nienadzorowanym algorytmem klasteryzacji potrafiącym identyfikować klastery o nieregularnych kształtach. Wyniki klasteryzacji modelu TF-IDF za pomocą HDBSCAN przedstawiono na Rysunku 7, a dla BERT na Rysunku 8.

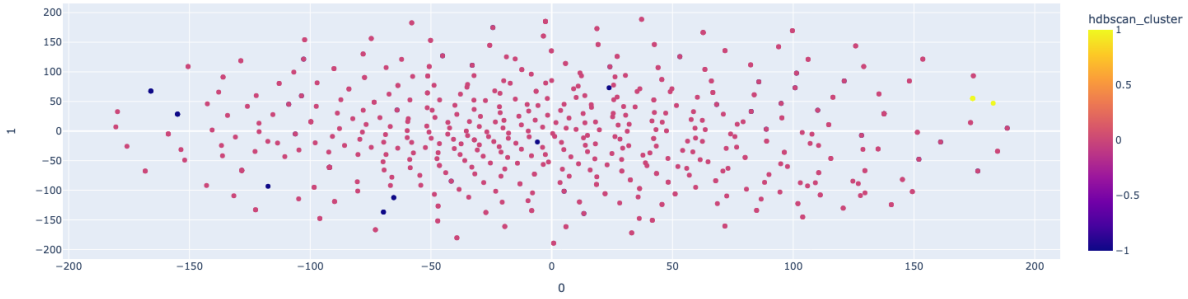

Figure 7: Klasteryzacja TF-IDF z użyciem algorytmu HDBSCAN

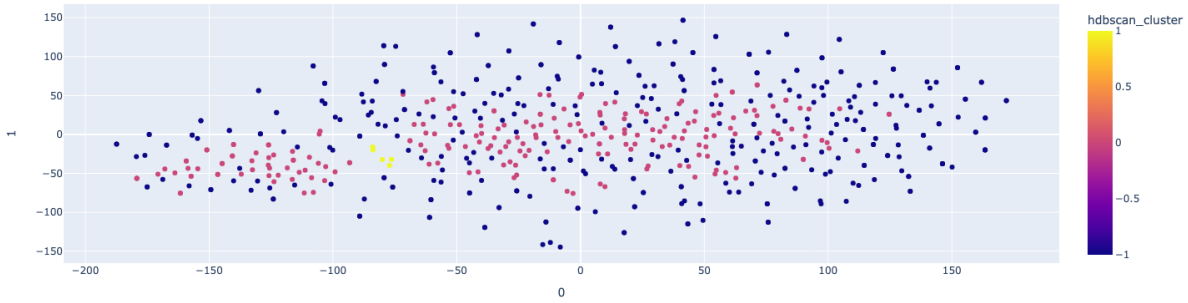

Figure 8: Klasteryzacja BERT z użyciem algorytmu HDBSCAN

3.4.3 Dyskusja wyników klasteryzacji

Analizując wyniki klasteryzacji dla obu algorytmów, możemy zauważyć różnice w rozkładzie punktów na wykresach. KMeans wyraźnie dzieli przestrzeń na równe klastry, podczas gdy HDBSCAN jest bardziej elastyczny i potrafi identyfikować mniej regularne struktury klastrów.