WUM projekt 2

Kraszewski Konstanty, Jung Jakub

Maj 2021

1 Wstęp

Raport ten opisuje proces tworzenia optymalnego modelu klasteryzującego dla zbioru tekstów religijnych. Jest to drugi projekt z przedmiotu *Wstęp do Uczenia Maszynowego* na wydziale Matematyki i Nauk Informacyjnych Politechniki Warszawskiej.

2 Opis problemu

Naszym zadaniem było stworzenie modelu grupującego teksty w klastry na podstawie wybranych przez nas cech.

3 Opis zbioru danych

Dane pochodzą ze strony: *A study of Asian Religious and Biblical Texts*.

Dane składają się z dwóch części: ramki danych z przetworzonymi tekstami (każdy wiersz to osobny rozdział, a w kolumnach znajdują się wyrazy z tych tekstów) oraz surowych tekstów ośmiu ksiąg religijnych:

- Buddyzm,
- Daodejing,
- Upaniszad,
- Jogasutra,
- Księga Przysłów,
- Księga Koheleta,
- Mądrość Syracha,
- Księga Mądrości.

Zamieszczona na stronie tabela okazała się problematyczna, ponieważ zawierając 590 wierszy miała aż 8267 kolumn i zdecydowana większość wartości była równa zero. Z tego powodu zdecydowaliśmy się na pracę z surowymi tekstami. Ostatecznie wykorzystaliśmy dwa różne podejścia do tych tekstów.

4 Preprocessing

W ramach pierwszego podejścia z tekstów zostały wybrane cztery opisujące je w pewien sposób statystyki:

- liczba słów w rozdziale,
- średnia długość słowa w rozdziale (w literach),
- liczba zdań w rozdziale,
- średnia długość zdania w rozdziale (w słowach).

Okazały się one wystarczające do uzyskania zadowalającego podziału tekstów.

Drugie podejście opierało się na porównywaniu tematów poszczególnych tekstów na podstawie występujących w nich słów. W tym celu na surowych danych zastosowaliśmy *tokenizer*, który z każdego tekstu usunął interpunkcję oraz podzielił teksty na poszczególne słowa. Pominęliśmy także stopwords (słowa o małym znaczeniu), jako że nie powinny one mieć żadnego wpływu na wykrywane tematy w tekstach, na podstawie których mielibyśmy je porównywać. W ten sposób otrzymaliśmy jeden zbiór danych. Do powstania drugiego zastosowaliśmy dodatkowo *stemmer*, który wszystkie słowa sprowadza do ich tematu słowotwórczego. Powinno to zapewnić lepsze wykrywanie podobnych tematów, ponieważ bez zastosowania tego procesu dwa słowa o praktycznie identycznym znaczeniu dla tekstu, ale np. w innej osobie lub w liczbie, są traktowane przez model jako zupełnie różne, natomiast po zastosowaniu stemmera i sprowadzeniu obu tych słów do tego samego tematu słowotwórczego model może lepiej wykrywać podobieństwa.

Następnie dla tak przetworzonych słów z obu zbiorów danych wytworzyliśmy dwie macierze częstości wyrazów przy użyciu metody TFIDF - "Term Frequency – Inverse Document Frequency". Metoda ta dla każdego słowa w każdym tekście wylicza wartość, która jest tym większa, im więcej razy dane słowo występuje w danym tekście oraz tym mniejsza, w im większej liczbie tekstów występuje to słowo. Dodatkowo odrzuca ona wszystkie słowa, które występują w większej liczbie tekstów niż ustalona wartość (jako że takie słowa najwyraźniej nie wpływają na znaczenie tekstu i nie pomagają w znalezieniu pomiędzy nimi podobieństw - podobnie jak stopwords) oraz takie, które występują w mniejszej liczbie tekstów niż ustalona (jako że takie słowa są zbyt specyficzne i mogą sprawiać, że podobne do siebie teksty zostaną przez model uznane za różniące się). Po przetestowaniu kilku wartości ustaliliśmy, że odrzucimy słowa występujące

w więcej niż 80% tekstów oraz w mniej niż 5% tekstów.

Dodatkowo dla wektorów z tej macierzy zastosowaliśmy miarę podobieństwa *cosine similarity* w celu stworzenia funkcji odległości dla jednego z modeli.

5 Model

Wstępna analiza tekstów z pomocą narzędzi TSNE, PCA oraz miary silhouette ujawniła, że można wyraźnie podzielić je na dwie grupy, czyli teksty chrześcijańskie oraz teksty niechrześcijańskie.

Utworzonych zostało siedem modeli klasteryzujących, które korzystały z czterech wygenerowanych wcześniej wartości. Powstałe podziały zobrazowaliśmy z użyciem PCA, a do ich oceny stworzona została metryka porównująca nowy podział na klastry z podziałem utworzonym na podstawie rzeczywistych etykiet. Większość modeli uzyskała dobre wyniki, co widać w tabeli:

	model	wynik
0	KMeans	0.969492
1	KMedoids	0.961017
2	AgglomerativeClustering (ward)	0.896610
3	AgglomerativeClustering (single)	0.808475
4	MiniBatchKMeans	0.967797
5	DBSCAN	0.933898
6	GaussianMixture	0.915254

Tablica 1: Wyniki modeli w pierwszym podejściu.

Następnie przeszliśmy do drugiego podejścia. Jako że chcieliśmy zobaczyć, jak modele poradzą sobie z tworzeniem klastrów zawierających teksty z ksiąg czterech różnych religii, początkowo przyporządkowaliśmy wszystkie teksty do czterech klastrów przy użyciu algorytmu *KMeans*. Następnie użyliśmy metody *tSNE*, aby móc zwizualizować nasz podział. Podobnie jak przy pierwszym podejściu model bardzo dobrze dzielił teksty na chrześcijańskie i niechrześcijańskie, natomiast gorzej radził sobie z podziałem samych tekstów niechrześcijańskich na poszczególne podgrupy. Wyniki zarówno dla zbioru danych bez stemmingu i ze stemmingiem były bardzo zbliżone.

Ciekawsze wyniki udało się uzyskać przy użyciu klasteryzacji hierarchicznej z połączeniami Ward'a na podstawie utworzonej wcześniej funkcji odległości. Model znów podzielił teksty na dwa duże klastry - teksty chrześcijańskie i niechrześcijańskie. Po stworzeniu dla niego dendrogramu zauważyliśmy jednak całkiem satysfakcjonującą kolejność łączenia tekstów w klastry. Na niższym poziomie model bardzo często łączył w klastry najpierw teksty z tych samych

ksiąg, co wskazuje na ich większe podobieństwo między sobą niż z tekstami z innych ksiąg tej samej religii.

6 Podsumowanie

Pierwsze z zastosowanych przez nas podejść pokazało, że wystarczy kilka prostych statystyk z danych tekstów, aby skutecznie podzielić je na dwie jasno określone grupy.

Natomiast w drugim przypadku model był w stanie na podstawie porównywania tematów tekstów wyróżnić teksty należące do poszczególnych ksiąg na wczesnych etapach klasteryzacji hierarchicznej, jednak finalnie, podobnie jak przy pierwszym podejściu, tworzył tylko dwa klastry. Możliwe, że przy lepszym doborze parametrów modelu podejście to pozwoliłoby na faktyczny podział tekstów na właściwe księgi.