Mateusz Soroka 250999
grupa 3

Sprawozdanie z projektu na przedmiot “Inteligencja obliczeniowa”

Gdańsk, 20.01.2020

1. Wstęp

Tematem projektu jest wyszukiwanie filmów na podstawie słów kluczowych, które generowane są na podstawie:

- tytułu filmu
- opisu filmu
- kategorii wiekowej
- gatunków filmu
- producentów filmu.

2. Narzędzia

Wykorzystane narzędzia:

- Python 3.7
- biblioteki / paczki:
    - pandas
    - re
    - ntlk
    - gensim

3. Dane

Wykorzystano bazę danych filmów z serwisu kaggle.com, link: https://www.kaggle.com/rounakbanik/the-movies-dataset#movies_metadata.csv

Baza zawiera dane odnośnie 45466 filmów.

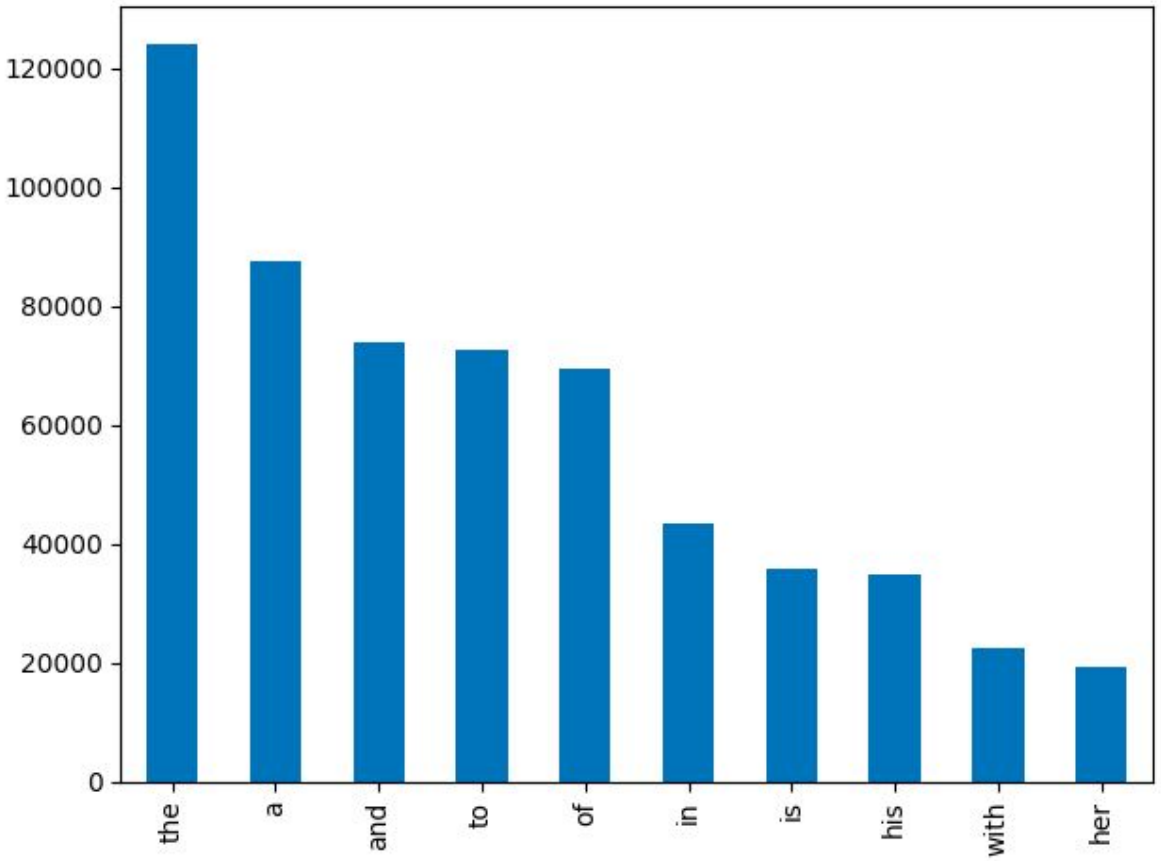
4. Teoria

Analiza tekstu jest dzisiaj szeroko wykorzystywana w świecie technologii. W medycynie używa się jej na przykład do wstępnej diagnozy chorób na podstawie wprowadzonych symptomów. Można jej również użyć do analizy sentymentu tweetów, postów na Facebooku itp. Kampania prezydencka Donalda Trumpa wykorzystała analizę tekstu do określenia preferencji wyborców, aby dopasować tę kampanię w taki sposób by była jak najbardziej skuteczna.

W dużo bardziej uproszczonej formie analizę tekstu można również wykorzystać do wyszukiwania filmów na podstawie słów kluczowych. Oczywiście dane należy wcześniej przygotować, aby analiza była skuteczna, albo nawet w

ogóle możliwa. Niżej przedstawiono kolejne kroki podjęte w celu normalizacji danych:

a. Pobranie stopwords (na polski stop-lista) czyli słów o małym znaczeniu, takich jak their, not, here, other. Warto zaznaczyć, że również przeglądarki internetowe nie biorą pod uwagę tych słów w wynikach wyszukiwania.
b. Pobranie 50 najczęściej występujących słów w opisach filmów. Poniżej wykres 10 najczęściej występujących:



c. Scalenie list z punktów a. i b. oraz usunięcie ich z listy słów kluczowych. Słowa z b. zostały uznane za nieprzydatne, ponieważ występują w opisach na tyle często, że nie stanowią wartości w analizie tekstu (w głównej mierze słowa z b. zaliczają się do stopwords). Nie wykluczono z kolei słów występujących najrzadziej, ponieważ one są na tyle specyficzne, że mogą pomóc w wyszukiwaniu tekstu.
d. Dodanie do opisu filmu: tytułu, gatunków, kategorii wiekowej (dzieci/dorośli), producenci.
e. Usunięcie z tekstu znaków interpunkcyjnych.
f. Zmiana słów na małe litery.
g. Usunięcie z tekstu znaków specjalnych.

h. Lematyzacja (lematyzacja to sprowadzanie danego słowa do jego formy podstawowej (hasłowej), która reprezentuje dany wyraz, np. *fly → fli*)
i. Usunięcie powtarzających się słów

Podobnemu przetworzeniu poddano słowa kluczowe wprowadzone przez użytkownika.

Następnie na podstawie słownika (konwersja listy wyrazów na listę postaci [token_id, token_count]) utworzono model TFIDF. TFIDF (ang. TF – *term frequenc*y, IDF – *inverse document frequency*) jest to metoda obliczania wagi słów w oparciu o liczbę ich wystąpień. Każdy dokument reprezentowany jest przez wektor, składający się z wag słów występujących w tym dokumencie. TFIDF informuje o częstości wystąpienia termów uwzględniając jednocześnie odpowiednie wyważenie znaczenia lokalnego termu i jego znaczenia w kontekście pełnej kolekcji dokumentów. Przykład TFIDF:

tf oblicza się według wzoru $\frac{Liczba\ wystąpień\ słowa\ x}{Suma\ wszystkich\ słów}$

Dokument 1

To	1	Suma słów: 5 Liczba wystąpień “to”: 1 Liczba wystąpień “inny”: 0 tf(“to”, Dokument 1) = 0.2 tf(“inny”, Dokument 1) = 0
jest	1	
przykład	2	
TFIDF	1	

Dokument 2

To	1	Suma słów: 8 Liczba wystąpień “to”: 1 Liczba wystąpień “inny”: 1 tf(“to”, Dokument 2) = 0.125 tf(“inny”, Dokument 2) = 0.125
inny	1	
przykład	3	
TFIDF	3	

idf oblicza się według wzoru $log(\frac{Liczba\ dokumentów}{W\ ilu\ dokumentach\ pojawia\ się\ słowo\ x})$

idf(“to”, Dokument 1 i 2) = $log(\frac{2}{2})$ = 0

idf(“inny”, Dokument 1 i 2) = $log(\frac{2}{1})$ = 0.301, a więc

tfidf(“to, Dokument 1) = 0.2 * 0 = 0
tfidf(“to, Dokument 2) = 0.125 * 0 = 0

tfidf(“inny”, Dokument 1) = 0 * 0.301 = 0
tfidf(“inny”, Dokument 2) = 0.125 * 0.301 = 0.037625

Z powyższego wynika, że słowo “to” pojawia się w obu dokumentach i nie jest “wartościowe” przy ocenie tekstu. tfidf wynosi w tym wypadku 0. Natomiast słowo “inny” pojawia się tylko w jednym dokumencie i jego tfidf wynosi 0.037625 co sprawia, że jest bardziej “wartościowe”.

Dokumenty/zdania/ciągi słów można porównywać ze sobą na wiele sposobów. Jednym z bardziej spopularyzowanych jest podobieństwo cosinusowe, które wygląda następująco:

	To	jest	inny	przykład	TFIDF	Wektory
D1	1	1	0	2	1	A = [1, 1, 0, 2, 1]
D2	1	0	1	3	3	B = [1, 0, 1, 3, 3]

W tabelce dla wszystkich dokumentów określa się częstotliwość występowanie wszystkich możliwych słów. Na tej podstawie buduje się wektory. Podobieństwo cosinusowe oblicza się według wzoru:

$$\frac{A * B}{|A| * |B|}$$

Jednak powyższy przykład nie uwzględnia wagi słów tak jak w modelu TFIDF, który w przypadku tych dwóch dokumentów wygląda następująco:

Słowo	TF		IDF	TFIDF	
	D1	D2		D1	D2
To	1/5	1/8	log(2/2) = 0	0	0
jest	1/5	0	log(2/1) = 0.301	0.0602	0
inny	0	1/8	log(2/1) = 0.301	0	0.037625
przykład	2/5	3/8	log(2/2) = 0	0	0
TFIDF	1/5	3/8	log(2/2) = 0	0	0

Łatwo zauważyć, że słowa “To”, “przykład” i “TFIDF” w tym porównaniu nie mają znaczenia, a kluczowe są słowa “jest” oraz “inny”. “jest” nie występuje w D2, a “inny” nie występuje w D1 więc te teksty z punktu widzenia modelu TFIDF nie są w ogóle do siebie podobne.

5. Eksperymenty

Do testów wybrano filmy:

- Toy Story
- Fast and Furious
- Forrest Gump

Toy Story ma kilka części, tak samo jak Fast and Furious, ale Forrest Gump już nie. Toy Story to seria filmów animowanych dla dzieci, Fast and Furious to seria filmów akcji dla dorosłych, a Forrest Gump to produkcja dla wszystkich bez kontynuacji. W przypadku filmów, które mają więcej niż jedną część, pomimo różnic w fabułach pomiędzy ów częściami, powinny zawierać podobne słowa kluczowe, a to oznacza, że podczas jednego wyszukiwania powinna zostać zwrócona cała seria filmów z jednego uniwersum.

6. Wyniki

Film	Słowa Kluczowe	Rezultaty - podobieństwo
Toy Story	toy story buzz astral andy woody night room animation child	1 'Toy Story 3' - 0.498 2 'Toy Story' - 0.426 3 'Toy Story 2' - 0.330 4 'Hawaiian Vacation' - 0.209 5 'Superstar: The Life and Times of Andy Warhol' - 0.156 6 'Child's Play 3' - 0.150 7 'Small Fry' - 0.150 8 'Love Finds Andy Hardy' - 0.146 9 'Welcome to Happiness' - 0.144 10 'Toy Story That Time Forgot' - 0.140
Fast and Furious	brian dom car fast action adult toretto race los angeles	1 'Fast Five' - 0.356 2 'Fast & Furious' - 0.353 3 'The Fast and the Furious' - 0.329 4 'Furious 7' - 0.239 5 'Welcome to L.A.' - 0.201 6 'Generation Um...' - 0.195 7 'Los Punks: We Are All We Have' -

		0.185 8 'Sunset Strip' - 0.181 9 'Out of Time' - 0.179 10 'The Fate of the Furious' - 0.178
Forrest Gump	forrest history determination love run low iq drama comedy	1 'Forrest Gump' - 0.414 2 'Genius' - 0.277 3 'The Genius Club' - 0.270 4 'Mr. Peabody & Sherman' - 0.246 5 'Astronaut: The Last Push' - 0.160 6 'Mori no densetsu' - 0.154 7 'No et moi' - 0.149 8 'Maniac Cop' - 0.149 9 'Pollyanna' - 0.144 10 'Gingerdead Man 2: Passion of the Crust' - 0.140