Informatyka, studia STACJONARNE, I st. semestr VI

Komputerowe systemy rozpoznawania **2020/2021**

Prowadzący: prof. dr hab. inż. Adam Niewiadomski poniedziałek, 12:00

Data oddania: _____ Ocena: _____

Julia Szymańska 224441
Przemysław Zdrzalik 224466

Projekt 1. Klasyfikacja dokumentów tekstowych

1. Cel projektu

Celem projektu jest stworzenie aplikacji klasyfikującej zadany zbiór danych tekstowych metodą K najbliższych sąsiadów (k-NN). Aplikacja ma za zadanie dokonać ekstrakcji cech na zbiorach tekstów[1] oraz następnie dokonać ich klasyfikacji.

2. Klasyfikacja nadzorowana metodą k-NN

Metoda K najbliższych sąsiadów, w skrócie metoda k-NN[1], jest to algorytm stosowany do klasyfikacji, który nie wymaga etapu uczenia. Polega na zaklasyfikowaniu rozpatrywanego elementu do grupy ze zbioru uczącego, gdzie spośród k najbliższych rozpatrywanemu elementowi sąsiadów najwięcej z nich należy do tej grupy. Klasyfikator przyjmuje cztery parametry wejściowe takie jak: warotść k - ilość rozpatrywanych sąsiadów, proporcje podziału zbiorów na zbior uczący i zbiór testowy, zbiór cech, a także metrykę i/lub miarę prawdopodobieństwa. Wynikiem klasyfikacji jest zaklasyfikowanie elementu do jednego ze zbiorów uczących.

2.1. Ekstrakcja cech, wektory cech

Na zbiorach danych tekstowych należy dokonać ekstrakcji cech, które będą wartościami rzeczywistymi oraz tekstowymi. Dane cechy będą repre-

zentowały tekst w postaci wektora cech podczas procesu klasyfikacji. Przed dokonaniem ekstrakcji cech, z tekstów usuwane są słowa znajdujące się na stop liście. Teksty ze zbioru danych tekstowych posiadają strukturę:

$$\begin{aligned} &< TEXT > \\ &\quad < TITLE/ > \\ &\quad < AUTHOR/ > \\ &\quad < DATELINE/ > \\ &\quad < BODY/ > \\ &< /TEXT > \end{aligned} \tag{1}$$

1. Liczba słów - cecha ta oznacza liczbę słów które składają się na pobrany tekst. Cecha ta będzie charakteryzowała długość dokumentu w postaci liczby całkowitej
$$c_1 = len \tag{2}$$
gdzie len - liczba słów w tekście.

2. Data z tagu <Dateline> - Każdy tekst w swoim body posiada tag <Dateline>, w którym znajduje się miasto oraz data podana w postaci miesiąca i dnia. Data będzie konwertowana na wartość liczbową, gdzie liczbą tą będzie numer podanego dnia w ciągu roku, licząc rok tak jakby rok był rokiem przestępnym, przykładowo data 1 marca będzie reprezentowana poprzez wartość 61. Cechę traktujemy jako cechę w postaci liczby całkowitej. Wartość będzie oznaczana poprzez symbol c_3.

3. Lokacja z tagu <Dateline>- jak wyżej. Lokację traktujemy jako cechę tekstową. Wartość będzie oznaczana poprzez symbol c_4.

4. Tytuł z tagu <Title>- Każdy tekst w swoim body posiada tag <Title>. Tytuł traktujemy jako cechę tekstową. Wartość będzie oznaczana poprzez symbol c_5.

5. Autor z tagu <Author>- Większość tekstów w swoim body posiada tag <Author>. Autora traktujemy jako cechę tekstową. Wartość będzie oznaczana poprzez symbol c_6.

6. Najczęściej występująca nazwa kraju - wybieramy najczęściej występującą w analizowanym tekście nazwę kraju. Nazwy krajów pobieramy z dołączonego pliku all-places-strings.lc, przykładowo krajem występującym w pliku jest 'albania'.Nazwę kraju traktujemy jako cechę tekstową.Wartość będzie oznaczana poprzez symbol c_7.

7. Zbiór występujących słów kluczowych. Za słowa kluczowe przyjmujemy słowa znajdujące się w dołączonych plikach o rozszerzeniach .lc.txt. Cechę traktujemy jako cechę tekstową.
$$c_8 : c_8 \in N \cap t \tag{3}$$

gdzie N - zbiór wszystkich słów kluczowych, t - zbiór słów należących do tekstu

8. Liczba wystąpień słów kluczowych - traktujemy jako cechę w postaci liczby całkowitej.

$$c_9 = |c_8| \tag{4}$$

gdzie c_8 - zbiór występujących słów kluczowych

9. Nasycenie tekstu ilością słów kluczowych - traktujemy jako cechę w postaci liczby zmienno przecinkowej.

$$c_{10} = c_9/c_1 \tag{5}$$

gdzie c_9 - ilość wystąpień słów kluczowych w tekscie, c_1 - liczba słów w tekście

10. Najczęściej występujące słowo kluczowe - wybieramy najczęściej występujące w analizowanym tekście słowo kluczowe. Cechę traktujemy jako cechę tekstową. Wartość będzie oznaczana poprzez symbol c_{11}.

11. Liczba unikatowych słów - zliczamy liczbę unikatowych słów, to znaczy występujących dokładnie raz w analizowanym tekście. Cechę traktujemy jako cechę w postaci liczby całkowitej. Wartość będzie oznaczana poprzez symbol c_{12}.

Wektor cech będzie reprezentowany w postaci:

$$w = [c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}] \tag{6}$$

2.2. Miary jakości klasyfikacji

W celu określenia jakości wykonanej klasyfikacji korzystamy z czterech miar jakości klasyfikacji. Aby obliczyć każdą z miar tworzymy tablicę pomyłek, inaczej macierz błędu [4]. Tablica składa się z dwóch wierszy i dwóch kolumn, gdzie wiersze to klasy predykowane, a kolumny to klasy rzeczywiste. Dane oznaczone jako dane pozytywne i negatywne poddawane są klasyfikacji, która przypisuje im predykowaną klasę pozytywną bądź negatywną.

Stosowane miary jakości klasyfikacji:

— Dokładność (ang. accurancy), ACC - jest to stopień zgodności wartości ze średnią arytmetyczną uzyskanych wyników.

$$ACC = (TP + TN)/(TP + FN + FP + TN) \tag{7}$$

— Precyzja (ang. precision), PPV - jest to stopień zgodności wyników uzyskanych w określonych warunkach z wielokrotnych pomiarów.

$$PPV = TP/(TP + FP) \tag{8}$$

		Klasa rzeczywista	
		Pozytywna	Negatywna
Klasa predykowana	Pozytywna	prawdziwie pozytywna (TP)	fałszywie pozytywna (FP)
	Negatywna	prawdziwie negatywna (TN)	fałszywie negatywna (FN)

Tabela 1. Wzór tablicy pomyłek.

— Czułość (ang. recall), TPR - jest to stosunek wyników prawdziwie dodatnich do sumy prawdziwie dodatnich i fałszywie ujemnych.

$$TPR = TP/(TP + FN) \tag{9}$$

— Swoistość (ang. specificity), TNR - jest to stopień zgodności wartości ze średnią arytmetyczną uzyskanych wyników.

$$TNR = TN/(FP + TN) \tag{10}$$

3. Klasyfikacja z użyciem metryk i miar podobieństwa tekstów

W procesie klasyfikacji możliwe jest wykorzystanie jedne z trzch metryk: Metryka Euklidesowa, Metryka Czebyszewa, Metryka Uliczna. Metryki służą obliczeniu odległości pomiędzy dwoma wektora o dowolnym rozmiarze.

Metryka Eukliedesowa[1] jest opisana wzorem:

$$d(x, y) = \sqrt{(y_1 - x_1)^2 + ... + (y_n - x_n)^2} \tag{11}$$

gdzie: d(x, y) - odległość pomiędzy wektorem x i y; x, y - wektory o tym samym rozmiarze; n - rozmiar wektorów x i y; x_n, y_n - składowe wektora.

Metryka Czebyszewa[1] jest opisana wzorem:

$$d(x, y) = max(|y_i - x_i|) \tag{12}$$

gdzie: d(x, y) - odległość pomiędzy wektorem x i y; x, y - wektory o tym samym rozmiarze; n - rozmiar wektorów x i y; x_i, y_i - i-ta składowa wektora;

Metryka Uliczna[1] jest opisana wzorem:

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i| \tag{13}$$

gdzie: d(x, y) - odległość pomiędzy wektorem x i y; x, y - wektory o tym samym rozmiarze; n - rozmiar wektorów x i y; x_n, y_n - składowe wektora.

By móc obliczyć odległość pomiędzy wektorami cech zadanych tekstów, należy wcześniej skorzystać z miar podobieństwa tekstu by zamienić tekst na liczbę w wektorach. W programie zostało użyte podobieństwo cosinusowe[3]. Należy utworzyć po jedym wektorze dla każdego z dwóch rozpatrywanych tekstów - x = { x_1, x_2, ... , x_n }, y = { y_1, y_2, ... , y_n }, gdzie x_n, y_n to ilość wystąpień n-tego słowa, ze zbioru wszystkich słów występujących w tekstach cech obu wektorów, w zadanej cesze wektora cech. Dla tak utworzonych wektrów liczony jest kosinus kąta pomiędzy nimi ze wzoru:

$$r_{cos} = \frac{\sum_{i=1}^{n} xy}{\sqrt{\sum_{i=1}^{n} x^2}\sqrt{\sum_{i=1}^{n} y^2}} \tag{14}$$

gdzie r_{cos} - to cos kąta pomiędzy wektorem x i y; x, y - rozpatrywane wektory; n - długość wektora.
Porównanie wyników klasyfikacji metody k-NN dla 10 różnych wartości parametru k. Pozostałe parametry pozostają niezmienne - proporcje podziału zbioru wektrów na zbiór treningowy i zbiór testowy - 80/20, a wybrana metryka to metryka Euklidesowa.

Tabela 2. Wstępne wyniki klasyfikacji dla różnych proporcji podziału zbioru wektrów na zbiór treningowy i zbiór testowy.

Wartość k	**Dokładność klasyfikacji -Accurancy**
1	0.5926966292134831
2	0.6713483146067416
3	0.6629213483146067
4	0.6657303370786517
5	0.6741573033707865
7	0.6657303370786517
8	0.6741573033707865
9	0.6741573033707865
13	0.6629213483146067
200	0.6769662921348315

Różne wartości parametru k nie mają znacznego znaczenia dal dokładności klasyfikacji. Najgorszy wynik był dla wartości parametru 1. Pozostałe wartości miały bardzo podobną dokładność klasyfikacji.

Kolejne porównanie jest dla różnych stosunków zbiorów uczących i testowych. Pozostałe parametry pozostają niezmienne - wartość parametru k to 2, a wybrana metryka to metryka Euklidesowa.

Tabela 3. Wstępne wyniki klasyfikacji dla różnych 10 warotści parametru k.

Stosunek zbioru treningowego do testowego	**Dokładność klasyfikacji -Accurancy**
1/99	0.45224719101123595
20/80	0.648876404494382
50/50	0.6713483146067416
70/30	0.6629213483146067
80/20	0.6348314606741573

Najgorzej wypadającym stosunkiem zbiorów uczących i testowych, gdzie stosunek wynosił 1/99, taki wynik został uzyskanu ponieważ mała ilość danych trenujących wpływa niekorzystnie na wyniki klasyfikacji. Najlepszy wynik był dla stosunku 50/50.

Następne porównanie jest dla różnych metryk. Pozostałe parametry pozostają niezmienne - wartość parametru k to 1, proporcje podziału zbioru wektrów na zbiór treningowy i zbiór testowy - 50/50.

Tabela 4. Wstępne wyniki klasyfikacji dla różnych 3 różnych metryk.

Metryka	**Dokładność klasyfikacji -Accurancy**
Euklidesowa	0.5758426966292135
Czebyszewa	0.5702247191011236
Uliczna	0.6235955056179775

Metryka Euklidesowa i Metryka Czebyszwa uzyskały podobny wynik jakości klasyfikacji. Metryka uliczna uzyskała najlepszy wynik.

Kolejne porównanie polega na porównaniu wpływu różnych cech na wyniki jakości klasyfikacji. W tym celu wybieramy 4 zestawy różnych cech.

1. Lokalizacja z tagu Dateline, data z tagu Dateline, tytul z tagu Title, autor z tagu Author
2. Najczęściej występujące słowo kluczowe, nasycenie tekstu słowami kluczowymi, liczba słów kluczowych, zbiór słów kluczowych
3. Długość tekstu, najczęściej występująca nazwa kraju, liczba unikatowych słów,
4. Długość tekstu, lokalizacja z tagu dateline, zbiór słów kluczowych, najczęściej występująca nazwa kraju
5. Tytuł z tagu Title
6. Lokalizacja z tagu Dateline
7. Tytuł z tagu Title, lokalizacja z tagu Dateline

Pozostałe parametry pozostają niezmienne - wartość parametru k to 2, proporcje podziału zbioru wektrów na zbiór treningowy i zbiór testowy - 70/30.

Najlepiej wypadającym zbiorem cech jest zbióroznaczony numerem 6. Przy czym najgorzej wypadającym zbiorem cech jest zbiór cech oznaczony numerem 3.

Tabela 5. Wstępne wyniki klasyfikacji dla różnych zbiorów cech.

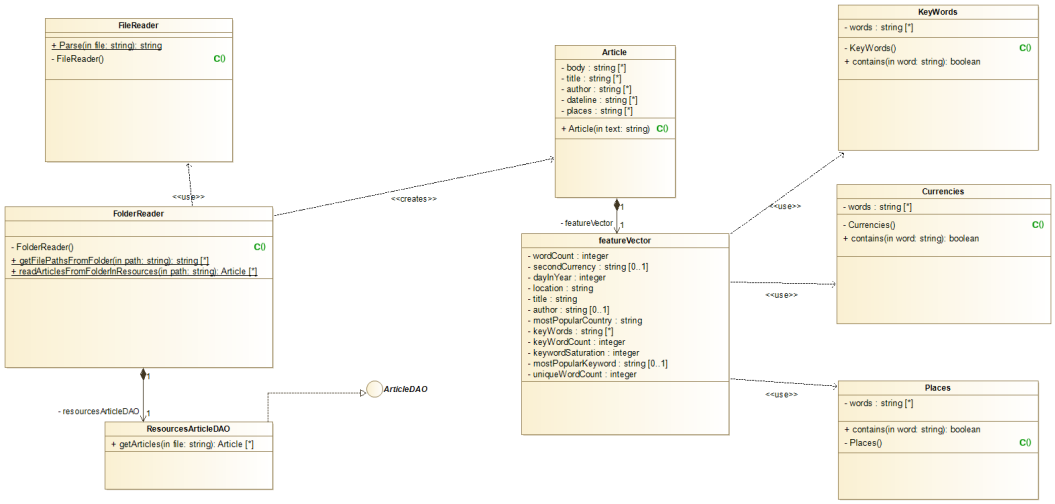
Numer zbioru cech	Dokładność klasyfikacji -Accurancy
1	0.7471910112359551
2	0.6741573033707865
3	0.6544943820224719
4	0.7584269662921348
5	0.800561797752809
6	0.9719101123595506
7	0.9634831460674157

Większość parametrów klasyfikacji nie ma większego wpływu na wyniki dokładności klasyfikacji. Największy wpływ na dokładność klasyfikacji ma wybrany zestaw cech.

4. Budowa aplikacji

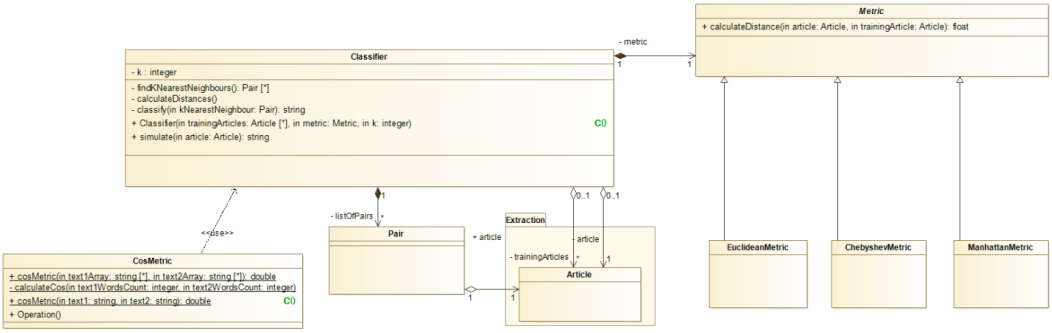
4.1. Diagramy UML

Aplikacja będzie składała się z dwóch modułów: z modułu ekstrakcji cech oraz z modułu klasyfikacji. Moduł ekstrakcji wczytuje pliki z treścią artykułów. Następnie tworzone są obiekty artykułów. Dla każdego obiektu usuwane są słowa ze stop listy oraz kolejno tworzone są wektory cech artykułów.



Rysunek 1. Diagram klas modułu ekstrakcji cech.

Moduł klasyfikacji oblicza odległości pomiędzy artykułem zadanym a każdym z artykułów ze zbioru trenującego za pomocą jednej z zadanych metryk [1] : metryki EuklidesowejUlicznej, , metryki metryki Czebyszewa. Dla cech zapisanych w postaci tekstowej ich odległość jest obliczana za pomocą podobieństwa kosinusowego. W ten sposób tworzone są pary zawierające artykuł i odleglość od zadanego artykułu. Następnie znajdowanych jest k najbliższych sąsiadów dla zadanego artykułu, gdzie poprzez słowo sąsiad rozumiemy artykuł ze zbioru trenującego. Ostatecznie artykuł jest klasyfikowany do klasy, której obiekty najczęściej wystąpiły wśród k najbliższych sąsiadów.



Rysunek 2. Diagram klas modułu klasyfikacji.

4.2. Prezentacja wyników, interfejs użytkownika

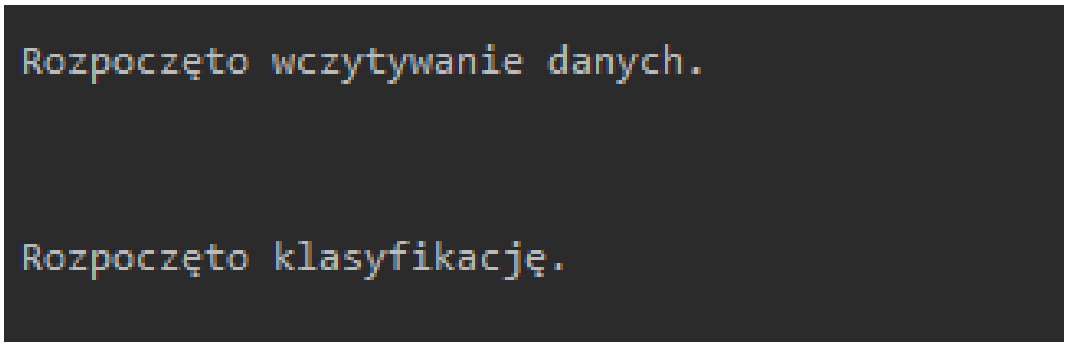
Po uruchomieniu programu użytkownik proszony jest o podanie poprzez konsolę kolejnych parametrów klasyfikacji. Na początku użytkownik podaje wartość parametru k, następnie wybiera jedną z trzech metryk, kolejno podawany jest procent zbioru treningowego w stosunku do zbioru wszystkich tekstów oraz użytkownik może podac cechy tesktów do klasyfikacji.Wybór parametrów w konsoli prezentuje się:

```
Podaj wartość k:
3
Wybierz metrykę:
1. Euklidesowa
2. Czebyszewa
3. Uliczna
2
Podaj procent artykułów treningowych:
70
Czy chcesz wybrać zestaw cech do klasyfikacji:
1. Tak
2. Nie
1
Cechy do wyboru:
1. Liczba słów
2. Autor
3. Liczba unikatowych słów
4. Data
5. Lokalizacja
6. Tytuł
7. Najczęściej występująca nazwa państwa
8. Kluczowe słowa
9. Liczba kluczowych słów
10. Nasycenie tekstu słowani kluczowymi
11. Najczęściej występujące słowo kluczowe
1 2 3 5 6
```

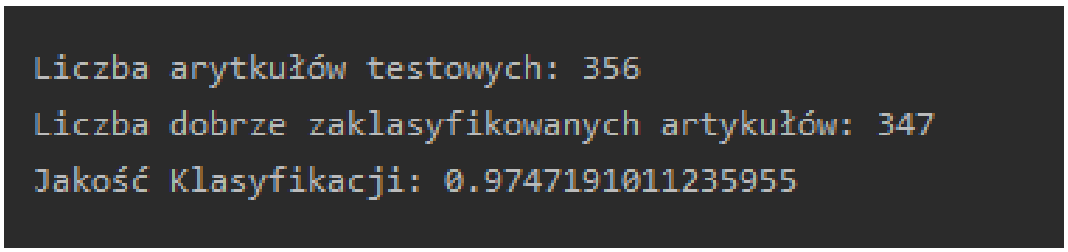
Rysunek 3. Wybór parametrów klasyfikacji przez użytkownika.

Po wprowadzeniu przez użytkownika wszystkich parametrów klasyfikacji, rozpoczynane jest wczytywanie danych oraz wykonanie klasyfikacji.

Po wykonanej klasyfikacji na konsoli wyświetlany jest wynik jakości klasyfikacji oraz liczba artykułów testowych, a także liczba dobrze zklasyfikowanych artykułów.



Rysunek 4. Wczytywanie danych i klasyfikacja.



Rysunek 5. Wynik klasyfikacji.

Do uruchomienia programu wymagana jest wersja Javy: 11.

5. Wyniki klasyfikacji dla różnych parametrów wejściowych

Wyniki kolejnych eksperymentów wg punktów 2.-8. opisu projektu 1. Wykresy i tabele obowiązkowe, dokładnie opisane w „captions" (tytułach), konieczny opis osi i jednostek wykresów oraz kolumn i wierszy tabel.

Ewentualne wyniki realizacji punktu 9. opisu Projektu 1., czyli „na ocenę 5.0" i ich porównanie do wyników z części obowiązkowej.

Sekcja uzupełniona jako efekt zadania Tydzień 05 wg Harmonogramu Zajęć na WIKAMP KSR.

6. Dyskusja, wnioski

Dokładne interpretacje uzyskanych wyników w zależności od parametrów klasyfikacji opisanych w punktach 3.-8 opisu Projektu 1. Szczególnie istotne są wnioski o charakterze uniwersalnym, istotne dla podobnych zadań. Omówić i wyjaśnić napotkane problemy (jeśli były). Każdy wniosek/problem powinien mieć poparcie w przeprowadzonych eksperymentach (odwołania do konkretnych wyników: wykresów, tabel).
Dla końcowej oceny jest to najważniejsza sekcja sprawozdania, gdyż prezen-

tuje poziom zrozumienia rozwiązywanego problemu.

** Możliwości kontynuacji prac w obszarze systemów rozpoznawania, zwłaszcza w kontekście pracy inżynierskiej, magisterskiej, naukowej, itp. **

Sekcja uzupełniona jako efekt zadania Tydzień 06 wg Harmonogramu Zajęć na WIKAMP KSR.

7. Braki w realizacji projektu 1.

Wymienić wg opisu Projektu 1. wszystkie niezrealizowane obowiązkowe elementy projektu, ewentualnie podać merytoryczne (ale nie czasowe) przyczyny tych braków.

Literatura

[1] R. Tadeusiewicz: Rozpoznawanie obrazów, PWN, Warszawa, 1991.

[2] A. Niewiadomski, Methods for the Linguistic Summarization of Data: Applications of Fuzzy Sets and Their Extensions, Akademicka Oficyna Wydawnicza EXIT, Warszawa, 2008.

[3] A. Niewiadomski, ksr-wyklad-2009.pdf, 2009.

[4] Internet forum. Wikipedia: The Free Encyclopedia, Dostępny w: `https://pl.wikipedia.org/wiki/Tablica_pomy%C5%82ek?fbclid=IwAR1yFbhG8HoSicSBnyA43YhpyU0tJiaIpI6ghUdNZvzDhPtMPwAWHtrdPUQ`

[5] Machine Learning Repository. UCI:, Dostępny w: `http://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection`

Literatura zawiera wyłącznie źródła recenzowane i/lub o potwierdzonej wiarygodności, możliwe do weryfikacji i cytowane w sprawozdaniu.