

Data oddania: _____

Ocena: _____

Julia Szymańska 224441
Przemysław Zdrzałik 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: wartość k - liczba rozpatrywanych sąsiadów, proporcje podziału zbiorów na zbiór 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.

$$c_9 = |c_8| \quad (4)$$

gdzie c_8 - zbiór występujących słów kluczowych

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

$$c_{10} = c_9/c_1 \quad (5)$$

gdzie c_9 - liczba wystąpień słów kluczowych w tekście, 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:

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

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.

We wzorach zostały użyte oznaczenia:

- TP - liczba poprawnie zaklasyfikowanych tekstów rozpatrywanej klasy
- TN - liczba poprawnie zaklasyfikowanych tekstów pozostałych klas
- FP - liczba tekstów pozostałych klas zaklasyfikowanych do rozpatrywanej klasy
- FN - liczba tekstów rozpatrywanej klasy zaklasyfikowanych do pozostałych klas

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

Tabela 1. Wzór tablicy pomyłek[4].

Stosowane miary jakości klasyfikacji:

- Dokładność (ang. accuracy), ACC - jest to stosunek poprawnie zaklasyfikowanych tekstów do wszystkich klasyfikowanych tekstów.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

- Precyzja (ang. precision), PPV - jest to stopień zgodności wyników uzyskanych w określonych warunkach z wielokrotnych pomiarów. Precyzja to stosunek liczby poprawnie zaklasyfikowanych tekstów rozpatrywanej klasy do liczby wszystkich tekstów zaklasyfikowanych do rozpatrywanej klasy. Dla całego zbioru dokumentów wartość miary jest liczona jako średnia ważona obliczonych precyzji dla pojedynczych klas, gdzie wagą jest stosunek liczebności tej klasy do liczebności wszystkich klas.

$$PPV = \frac{TP}{TP + FP} \quad (8)$$

- Czułość (ang. recall), TPR - jest to stosunek liczby poprawnie zaklasyfikowanych tekstów do rozpatrywanej klasy do liczby tekstów z rozpatrywanej klasy. Dla całego zbioru dokumentów wartość miary jest liczona jako średnia ważona obliczonych czułości dla pojedynczych klas, gdzie wagą jest stosunek liczebności tej klasy do liczebności wszystkich klas.

$$TPR = \frac{TP}{TP + FN} \quad (9)$$

- Miara F1 - średnia harmoniczna miar Precyzja i Czułość.

$$F1 = \frac{2}{\frac{1}{PPV} + \frac{1}{TPR}} \quad (10)$$

3. Klasyfikacja z użyciem metryk i miar podobieństwa tekstów

W procesie klasyfikacji możliwe jest wykorzystanie jednej z trzech metryk: metryka Euklidesowa, metryka Czebyszewa, metryka Uliczna. Metryki służą obliczeniu odległości pomiędzy dwoma wektorami o dowolnym rozmiarze.

Metryka Euklidesowa[1] jest opisana wzorem:

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

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:

$$d(x, y) = \max(|y_i - x_i|) \quad (12)$$

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:

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

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 cechy o wartościach tekstowych na liczby w wektorach. W programie zostało użyte podobieństwo kosinusowe[3]. Należy utworzyć po jednym wektorze dla cechy o wartości tekstowej dla każdego z dwóch rozpatrywanych tekstów - $x = \{x_1, x_2, \dots, x_n\}$, $y = \{y_1, y_2, \dots, y_n\}$, gdzie x_n, y_n to liczba wystąpień n -tego słowa, ze zbioru wszystkich słów występujących w obu wektorach, w zadanej cesze tekstowej wektora cech. Dla tak utworzonych wektorów liczony jest kosinus kąta pomiędzy nimi ze wzoru:

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

gdzie r_{cos} - to cos kąta pomiędzy wektorem x i y ; x, y - rozpatrywane wektory; n - długość wektora.

Wstępna klasyfikacja na ograniczonym zbiorze tekstów została przeprowadzona dla trzech różnych zestawów parametrów wejściowych.

Parametry wejściowe dla pierwszej klasyfikacji wstępnej:

Tabela 2. Parametry wejściowe dla pierwszej wstępnej klasyfikacji.

K	Metryka	Procent zbioru trenującego	Wybrane cechy
5	Czebyszewa	80%	Wszystkie cechy

Wstępne wyniki miary Accuracy dla pierwszej klasyfikacji wstępnej:

Tabela 3. Wstępne wyniki miary Accuracy dla pierwszej klasyfikacji wstępnej.

Liczba tekstów	Liczba poprawnie sklasyfikowanych tekstów	Accuracy
394	230	0,58

Parametry wejściowe dla drugiej klasyfikacji wstępnej:

Tabela 4. Parametry wejściowe dla drugiej wstępnej klasyfikacji.

K	Metryka	Procent zbioru trenującego	Wybrane cechy
			1. Lokalizacja 2. Tytuł
3	Euklidesowa	95%	3. Najczęściej występująca nazwa państwa 4. Kluczowe słowa

Wstępne wyniki miary Accuracy dla drugiej klasyfikacji wstępnej:

Tabela 5. Wstępne wyniki miary Accuracy dla drugiej klasyfikacji wstępnej.

Liczba tekstów	Liczba poprawnie sklasyfikowanych tekstów	Accuracy
394	295	0,75

Parametry wejściowe dla trzeciej klasyfikacji wstępnej:

Tabela 6. Parametry wejściowe dla trzeciej wstępnej klasyfikacji.

K	Metryka	Procent zbioru trenującego	Wybrane cechy
			1. Kluczowe słowa 2. Liczba kluczowych słów 3. Nasycenie tekstu słowami kluczowymi
9	Uliczna	73%	4. Najczęściej występujące słowo kluczowe

Wstępne wyniki miary Accuracy dla trzeciej klasyfikacji wstępnej:

Tabela 7. Wstępne wyniki miary Accuracy dla trzeciej klasyfikacji wstępnej.

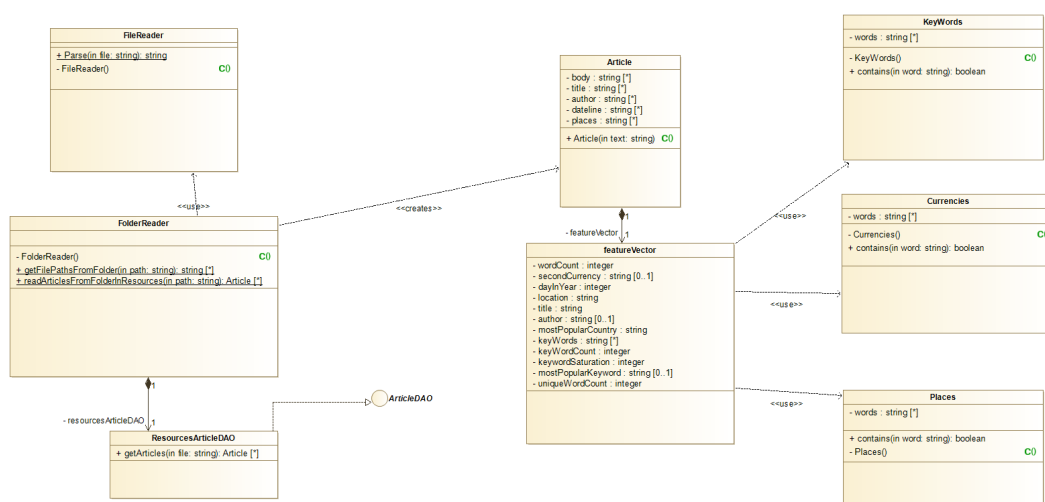
Liczba tekstów	Liczba poprawnie sklasyfikowanych tekstów	Accuracy
394	236	0,60

Najlepsze wyniki zostały uzyskane dla drugiej wstępnej klasyfikacji, w której został ograniczony zbiór cech, wybrane cechy to: kluczowe słowa, liczba kluczowych słów, nasycenie tekstu słowami kluczowymi, najczęściej występujące słowo kluczowe.

4. Budowa aplikacji

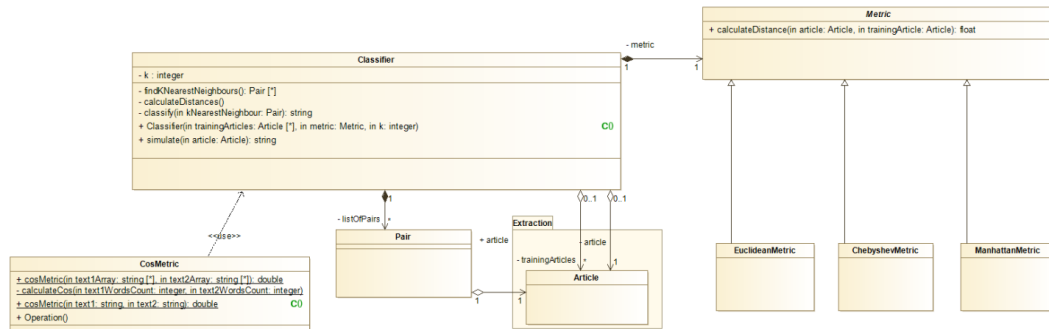
4.1. Diagramy UML

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 zadany a każdym z artykułów ze zbioru trenującego za pomocą jednej z zadanych metryk [1] : metryki Euklidesowej, metryki Ulicznej, 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 odległość 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 podać cechy tekstów 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 słowami 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.

```
Rozpoczęto wczytywanie danych.
```

```
Rozpoczęto klasyfikację.
```

Rysunek 4. Wczytywanie danych i klasyfikacja.

Po wykonanej klasyfikacji na konsoli wyświetlane są obliczone parametry dla poszczególnych klas klasyfikacji oraz wyliczone parametry dla całego zbioru dokumentów. Dla poszczególnych klas klasyfikacji do obliczonych parametrów zaliczamy liczbę tekstów klasy, liczbę poprawnie zaklasyfikowanych tekstów do rozpatrywanej klasy, liczbę tekstów innych klas zaklasyfikowanych do rozpatrywanej klasy oraz miary jakości: Precision, Recall, F1.

```
-----  
west-germany:  
Liczba tekstów klasy: 27  
Liczba poprawnie zaklasyfikowanych tekstów: 20  
Liczba tekstów innych klas zaklasyfikowanych do tej klasy: 0  
Precision: 1,00  
Recall: 0,74  
F1: 0,85  
-----
```

Rysunek 5. Wynik klasyfikacji dla pojedynczej klasy klasyfikacji - klasa west-germany.

Dla całego zbioru dokumentów do obliczonych parametrów zaliczamy liczbę tekstów testowych, liczbę poprawnie zaklasyfikowanych tekstów oraz miary jakości: Accuracy, Precision, Recall, F1.

```
-----  
wszystkie:  
Liczba tekstów testowych: 394  
Liczba dobrze zaklasyfikowanych tekstów: 367  
Accuracy: 0,93  
Precision: 0,94  
Recall: 0,93  
F1: 0,93  
-----
```

Rysunek 6. Wynik klasyfikacji dla całego zbioru dokumentów.

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 prezentuje 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.