

Data oddania: \_\_\_\_\_

Ocena: \_\_\_\_\_

Piotr Traczyk      123123  
Bartosz Jurczewski   210209

## Zadanie 1: Ekstrakcja cech, miary podobieństwa, klasyfikacja\*

### 1. Cel

Celem zadania było stworzenie aplikacji do klasyfikacji tekstów metodą k-NN, korzystając z różnych sposobów ekstrakcji wektorów cech i metryk. Dodatkowo przeanalizowanie dla jakich parametrów program ma największą skuteczność.

### 2. Wprowadzenie

Zagadnieniem, w okół którego skupiona jest nasza aplikacja jest klasyfikacja statystyczna, która jest rodzajem algorytmu statystycznego przydzielającego elementy do klas, bazując na cechach owych elementów. W ramach przeprowadzanego eksperymentu zaimplementowaliśmy klasyfikator k-najbliższych sąsiadów.

Algorytm k najbliższych sąsiadów, nazywany także algorytmem k-NN, należy do grupy algorytmów leniwych, czyli takich, które nie tworzą wewnętrznej reprezentacji danych uczących, lecz szukają rozwiązania dopiero w momencie pojawienia się wzorca testującego. Przechowuje wszystkie wzorce uczące, względem których wyznacza odległość wzorca testowego [2]. Metoda

---

\* Repozytorium github: <https://github.com/jurczewski/KSR>

k-NN wyznacza k sąsiadów, do których badany element ma najmniejszą odległość w danej metryce, a następnie wyznacza wynik w oparciu o najczęstszy element, wśród k najbliższych. W przypadku naszego projektu odległość definiujemy jako skalę podobieństwa tekstów.

W ramach zadania zostały użyte 2 metody ekstrakcji cech:

- o Inverse document frequency - metoda polegająca na wyznaczeniu, czy dane słowo występuje powszechnie we wszystkich dokumentach. Jest to logarytmicznie skalowana odwrotna część dokumentów zawierających wybrane słowo (uzyskana poprzez podzielenie całkowitej liczby dokumentów przez liczbę dokumentów zawierających ten termin). Obliczana jest z poniższego wzoru:

$$idf_i = \log \frac{|D|}{|\{d : t_i \in d\}|}$$

- o Term frequency - metoda polegająca na zliczeniu częstości występowania danego słowa w dokumencie. Obliczana jest z poniższego wzoru:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Do obliczenia odległości tekstów posłużyliśmy się 3 metrykami:

- metryka Euklidesowa - w celu obliczenia odległości  $d_e(x, y)$  między dwoma punktami  $x, y$  należy obliczyć pierwiastek kwadratowy z sumy drugich potęg różnic wartości współrzędnych o tych samych indeksach, zgodnie ze wzorem:

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

- metryka uliczna (Manhattan, miejska) - w celu obliczenia odległości  $d_e(x, y)$  między dwoma punktami  $x, y$  należy obliczyć sumę wartości bezwzględnych różnic współrzędnych punktów  $x$  oraz  $y$ , zgodnie ze wzorem:

$$d_m(x, y) = \sum_{k=1}^n |x_k - y_k|$$

- metryka Czebyszewa - w celu obliczenia odległości  $d_e(x, y)$  między dwoma punktami  $x, y$  należy obliczyć maksymalną wartość bezwzględnych różnic współrzędnych punktów  $x$  oraz  $y$ , zgodnie ze wzorem:

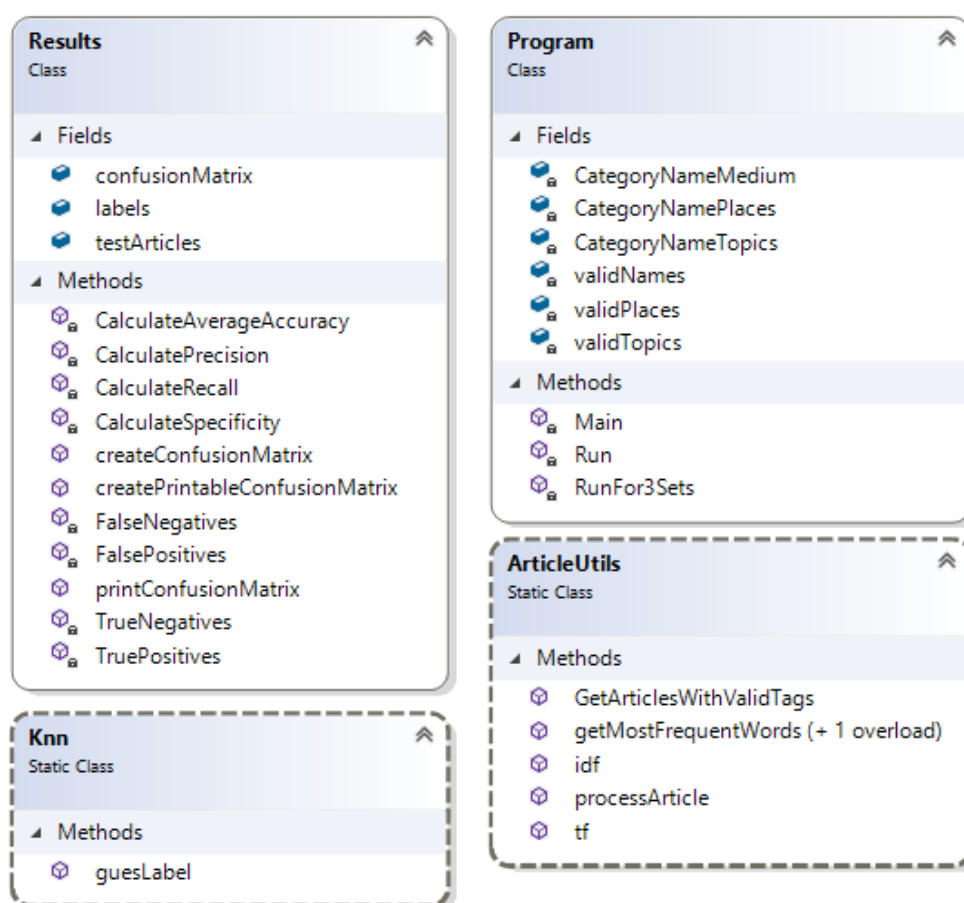
$$d_{ch}(x, y) = \max_i |x_i - y_i|$$

### 3. Opis implementacji

Program został w całości stworzony w języku C# (.NET Framework, v4.6.1).

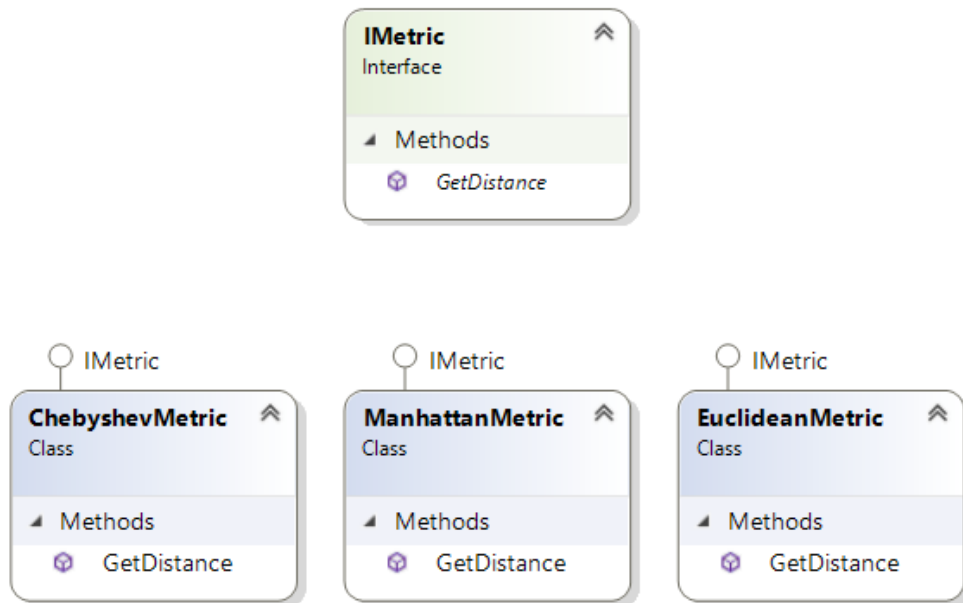
W programie znajdują się następujące klasy:

- Knn - implementacja algorytmu K-nn
- Program - klasa główna
- ArticleUtils - klasa pomocnicza
- Results - klasa odpowiedzialna za dokładne zestawienie rezultatów



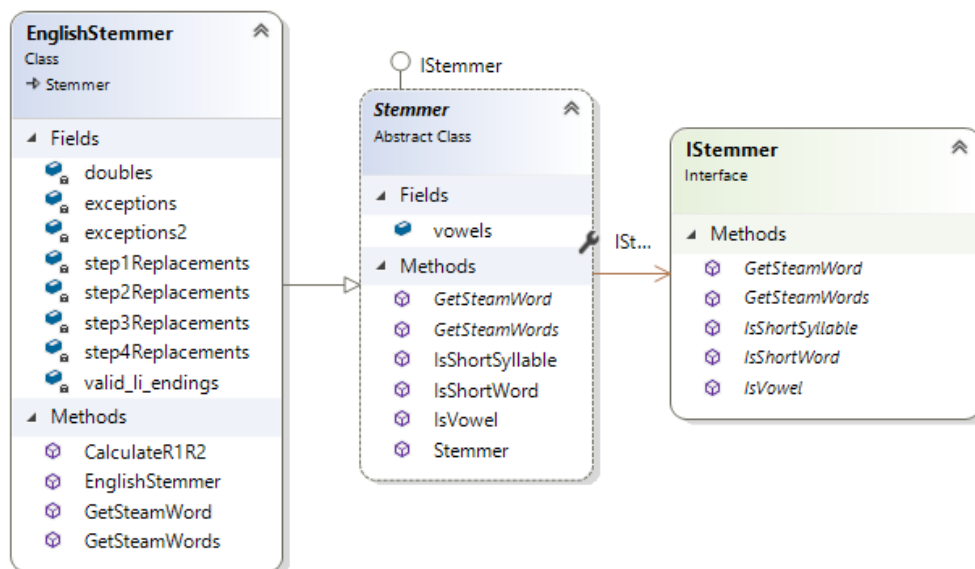
Rysunek 1. Diagram UML wygenerowany dla klas ogólnych.

- IMetric - interfejs dla metryk, klasy implementujące owy interfejs:
  - ChebyshevMetric - klasa odpowiedzialna za metrykę Czebyszewa
  - ManhatattanMetric - klasa odpowiedzialna za metrykę uliczną
  - EuclideanMetric - klasa odpowiedzialna za metrykę euklidesową



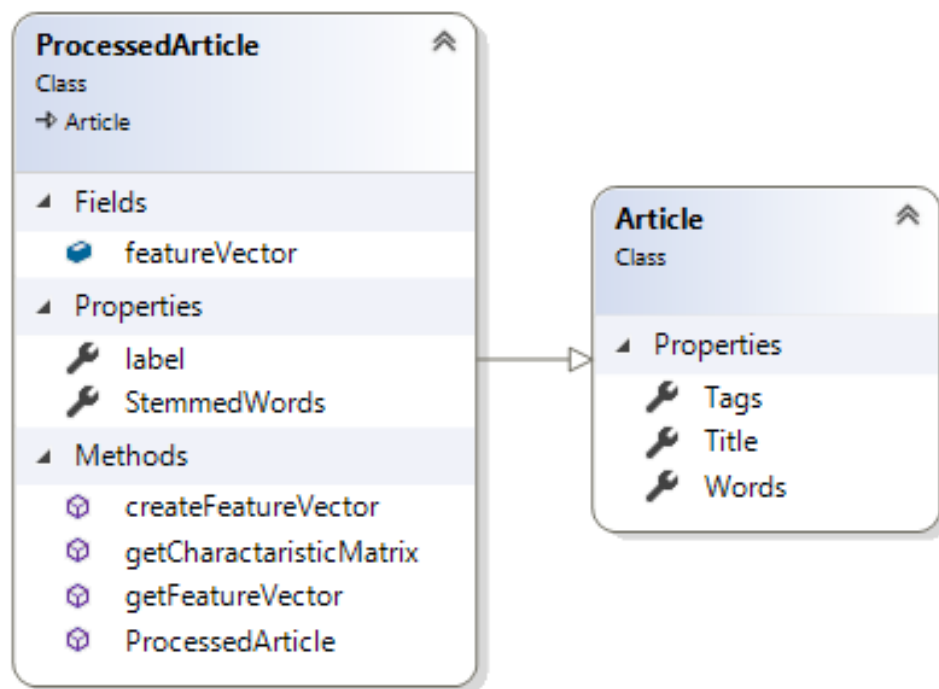
Rysunek 2. Diagram UML wygenerowany dla klas dotyczących metryk.

- IStemmer - interfejs reprezentujący stemmer
- Stemmer - reprezentacja stemera
- EnglishStemmer - stemer dla języka angielskiego



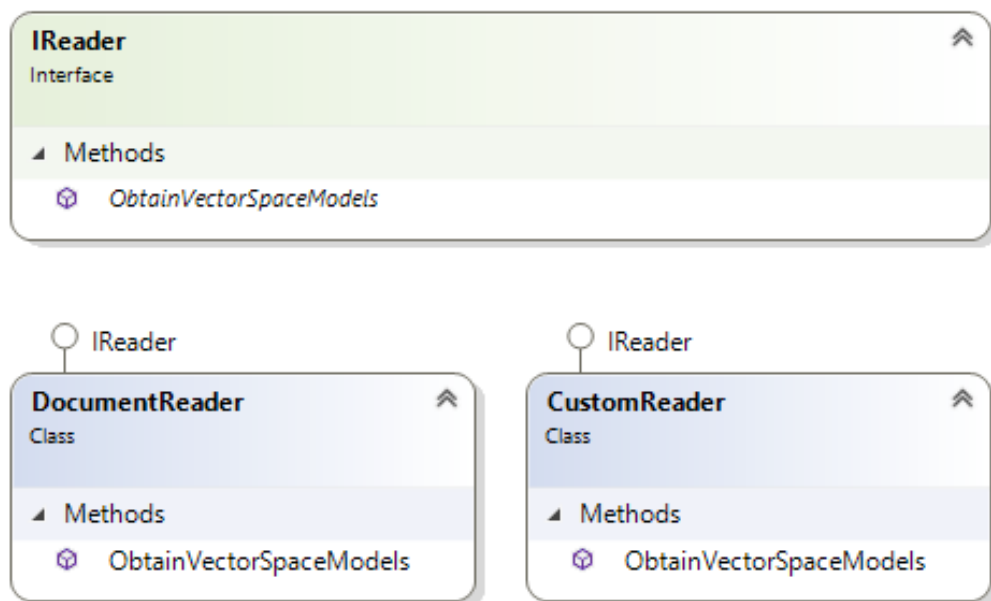
Rysunek 3. Diagram UML wygenerowany dla klas dotyczących procesu stemizacji.

- Article - klasa reprezentująca pojedynczy artykuł
- ProcessedArticle - klasa dziedzicząca po Article, reprezentuje przetworzony artykuł



Rysunek 4. Diagram UML wygenerowany dla klas reprezentujących artykuły.

- IReader - interfejs dla klas wczytujących dane, klasy implementujące owy interfejs:
  - DocumentReader - klasa odpowiedzialna za wczytanie plików z zestawu [4]
  - CustomReader - klasa odpowiedzialna za wczytanie plików z przygotowanego przez nas zestawu [5]



Rysunek 5. Diagram UML wygenerowany dla klas wczytujących dane.

## 4. Materiały i metody

Klasyfikacja tekstów została wykonana wszystkimi dostępnymi metodami ekstrakcji cech dla wszystkich trzech metryk. Dla każdego przypadku testowego dokonano klasyfikacji tekstu dla  $k \in \{2, 3, 5, 7, 10, 15, 20\}$  najbliższych sąsiadów. Wyniki porównano z faktyczną etykietą danego artykułu.

Przyjeliśmy trzy różne proporcje zbioru treningowego do zbioru testowego, było to:

- zbiór treningowy 20%, zbiór testowy 80%
- zbiór treningowy 60%, zbiór testowy 40%
- zbiór treningowy 80%, zbiór testowy 20%

Klasyfikacja dotycząca lokalizacji przeprowadzana była jedynie na danych, których pole places przyjmowało jedną z wartości: west-germany, usa, france, uk, canada, japan.

Klasyfikacja dotycząca tematów przeprowadzana była jedynie na danych, które pole topics przyjmowało jedną z wartości: gold, cocoa, sugar, coffe, grain.

Danymi jakimi przygotowaliśmy do analizy były opisy dzieł kultury. Posiadały one jedną kategorię - medium. Pole przyjmowało wartość: book lub movie.

## 5. Wyniki

### 5.1. Term frequency

#### 5.1.1. Metryka Euklidesowa

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 1. Skuteczność klasyfikacji dla podziału: zbiór treningowy 20%, zbiór testowy 80%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 2. Skuteczność klasyfikacji dla podziału: zbiór treningowy 60%, zbiór testowy 40%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 3. Skuteczność klasyfikacji dla podziału: zbiór treningowy 80%, zbiór testowy 20%

#### 5.1.2. Metryka uliczna

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 4. Skuteczność klasyfikacji dla podziału: zbiór treningowy 20%, zbiór testowy 80%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 5. Skuteczność klasyfikacji dla podziału: zbiór treningowy 60%, zbiór testowy 40%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 6. Skuteczność klasyfikacji dla podziału: zbiór treningowy 80%, zbiór testowy 20%

### 5.1.3. Metryka Czebyszewa

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 7. Skuteczność klasyfikacji dla podziału: zbiór treningowy 20%, zbiór testowy 80%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 8. Skuteczność klasyfikacji dla podziału: zbiór treningowy 60%, zbiór testowy 40%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	72,47	22,22	61,25

Tabela 9. Skuteczność klasyfikacji dla podziału: zbiór treningowy 80%, zbiór testowy 20%

## 5.2. Inverse document frequency

### 5.2.1. Metryka Euklidesowa

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	78,7	32,6	57,5
3	82,6	31,5	43,8
5	82,9	29,3	65
7	82,3	33,7	66,2
10	81,7	33,7	51,2
15	81,3	35,9	46,2
20	81	35,9	46,2

Tabela 10. Skuteczność klasyfikacji dla podziału: zbiór treningowy 20%, zbiór testowy 80%



<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	78,6	34,8	57,5
3	82,3	32,6	62,5
5	83,5	28,3	45
7	83,7	28,3	67,5
10	83,4	28,3	30
15	82,8	21,7	42,5
20	82,6	26,1	45

Tabela 11. Skuteczność klasyfikacji dla podziału: zbiór treningowy 60%, zbiór testowy 40%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	80,5	30,4	35
3	83,8	34,8	65
5	83,8	21,7	55
7	83,8	26,1	45
10	83,5	26,1	50
15	83,7	43,5	50
20	83,1	34,8	70

Tabela 12. Skuteczność klasyfikacji dla podziału: zbiór treningowy 80%, zbiór testowy 20%

### 5.2.2. Metryka uliczna

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	77,8	31,5	50
3	81,2	34,8	63,7
5	81,7	33,7	52,5
7	81,3	34,8	45
10	81,2	35,9	57,5
15	81	35,9	47,5
20	81	35,9	47,5

Tabela 13. Skuteczność klasyfikacji dla podziału: zbiór treningowy 20%, zbiór testowy 80%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	78,8	19,6	60
3	81,8	19,6	52,5
5	82,6	19,6	65
7	82,8	17,4	67,5
10	82,2	19,6	50
15	81,9	19,6	70
20	81,5	26,1	75

Tabela 14. Skuteczność klasyfikacji dla podziału: zbiór treningowy 60%, zbiór testowy 40%

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	79,2	26,1	50
3	83,3	30,4	50
5	83,4	26,1	45
7	83,3	21,7	60
10	82,9	26,1	65
15	82,8	39,1	45
20	82,2	26,1	70

Tabela 15. Skuteczność klasyfikacji dla podziału: zbiór treningowy 80%, zbiór testowy 20%

### 5.2.3. Metryka Czebyszewa

<b>k</b>	<b>places [%]</b>	<b>topics [%]</b>	<b>medium [%]</b>
2	77,6	31,5	62,5
3	80,7	31,5	51,2
5	81,5	40,2	58,8
7	81,2	33,7	70
10	81	34,8	47,5
15	80,5	35,9	57,5
20	79,6	35,9	50

Tabela 16. Skuteczność klasyfikacji dla podziału: zbiór treningowy 20%, zbiór testowy 80%

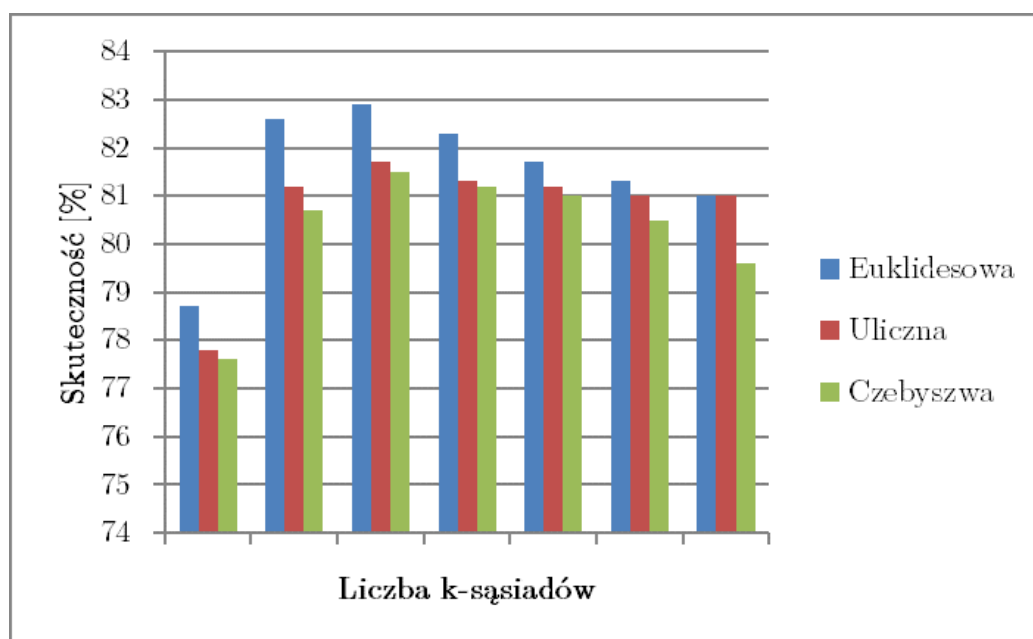
k	places [%]	topics [%]	medium [%]
2	79,8	41,3	47,5
3	81,8	63	42,5
5	81,6	28,3	62,5
7	81,7	39,1	70
10	81,6	34,8	65
15	81,6	23,9	45
20	81,5	23,9	52,5

Tabela 17. Skuteczność klasyfikacji dla podziału: zbiór treningowy 60%, zbiór testowy 40%

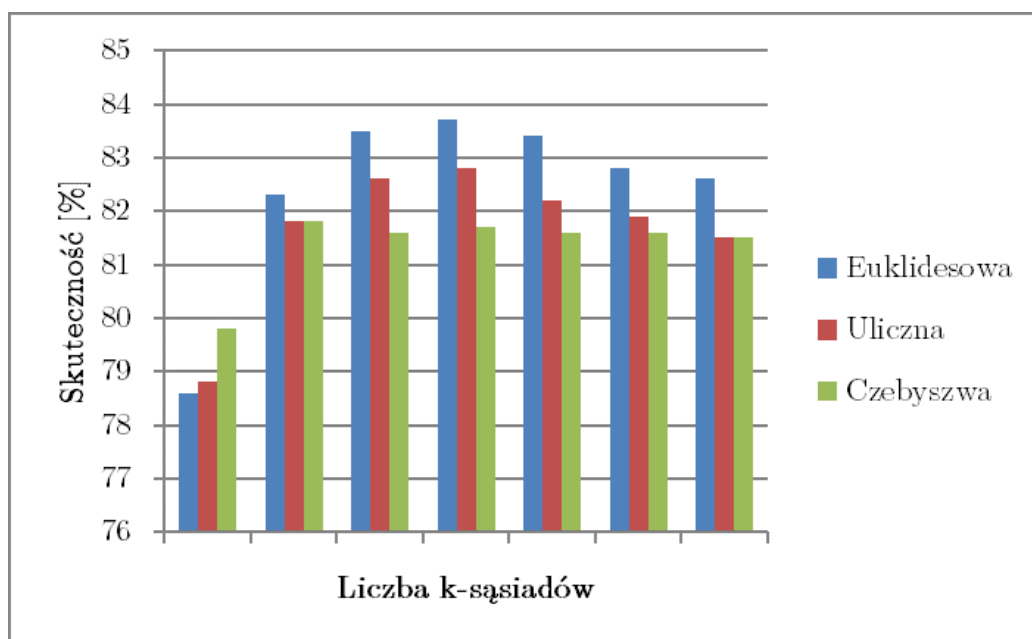
k	places [%]	topics [%]	medium [%]
2	79,1	43,5	80
3	82,4	39,1	60
5	81,6	26,1	40
7	81,7	30,4	50
10	81,9	26,1	55
15	82,2	26,1	60
20	82,1	26,1	65

Tabela 18. Skuteczność klasyfikacji dla podziału: zbiór treningowy 80%, zbiór testowy 20%

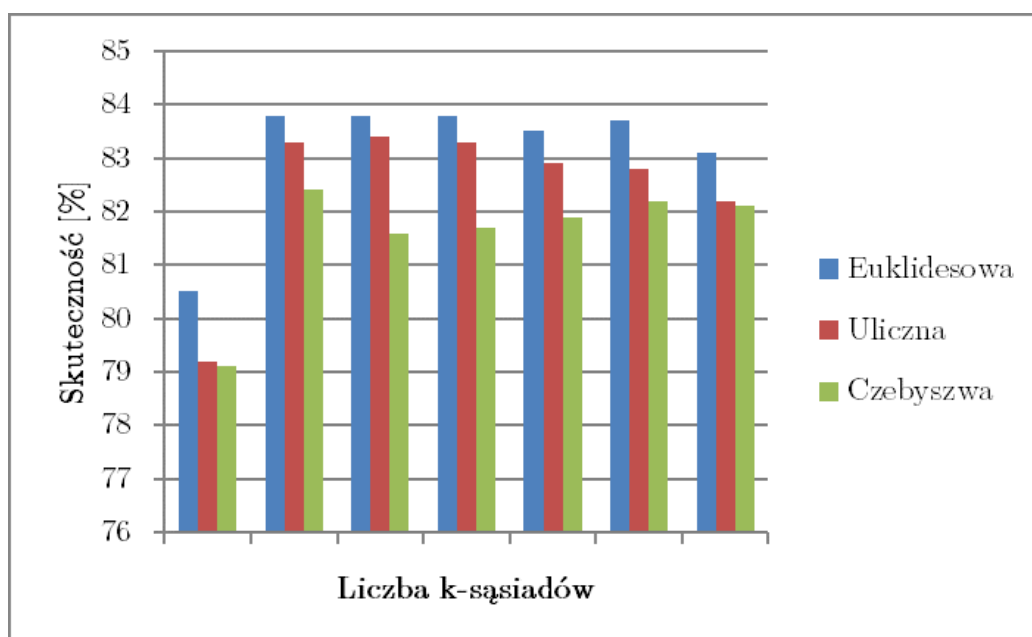
### 5.3. Inverse document frequency - podsumowanie wykresami



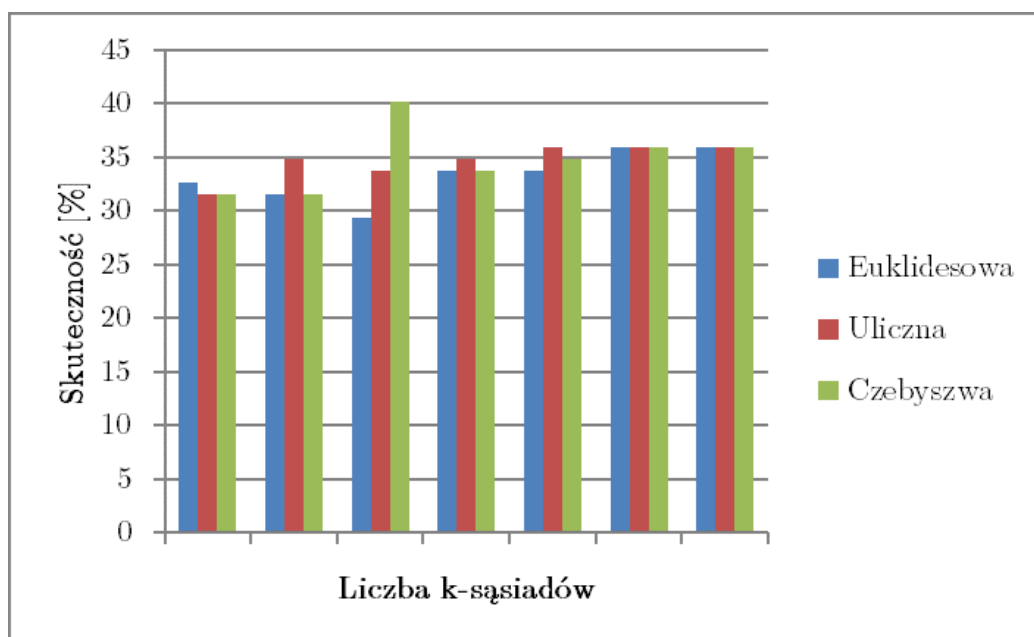
Rysunek 6. Dane z tabel 1-9 dla kategorii places, zbiór treningowy 20%, zbiór testowy 80%



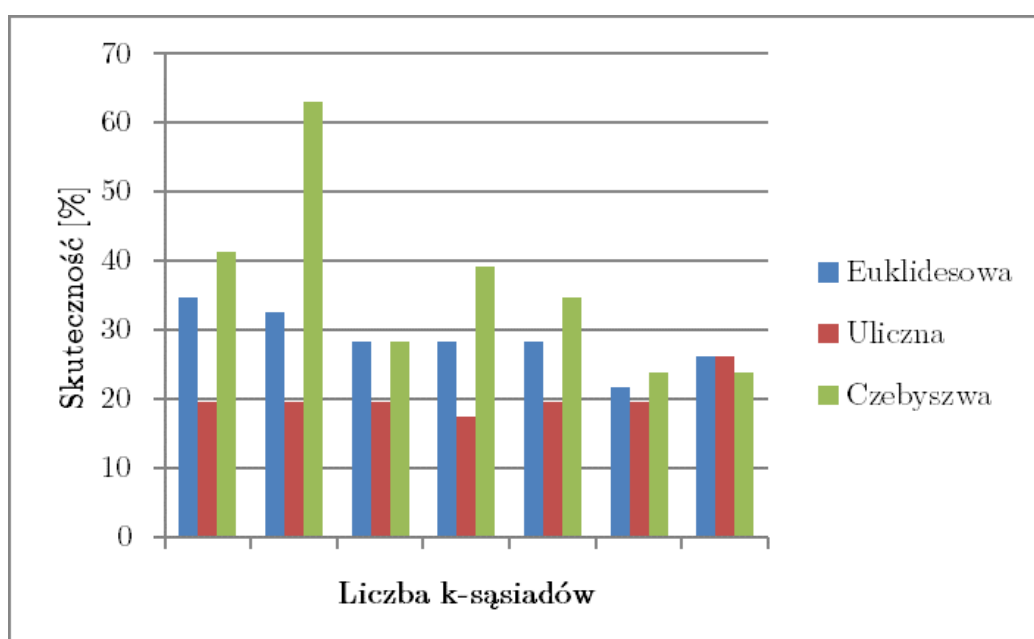
Rysunek 7. Dane z tabel 1-9 dla kategorii places, zbiór treningowy 60%, zbiór testowy 40%



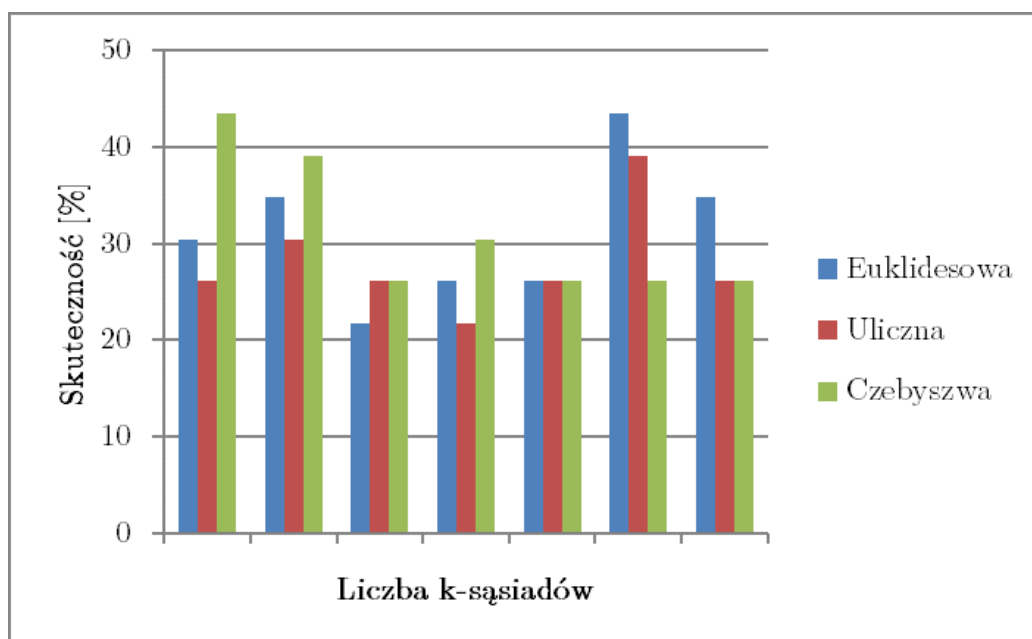
Rysunek 8. Dane z tabel 1-9 dla kategorii places, zbiór treningowy 80%, zbiór testowy 20%



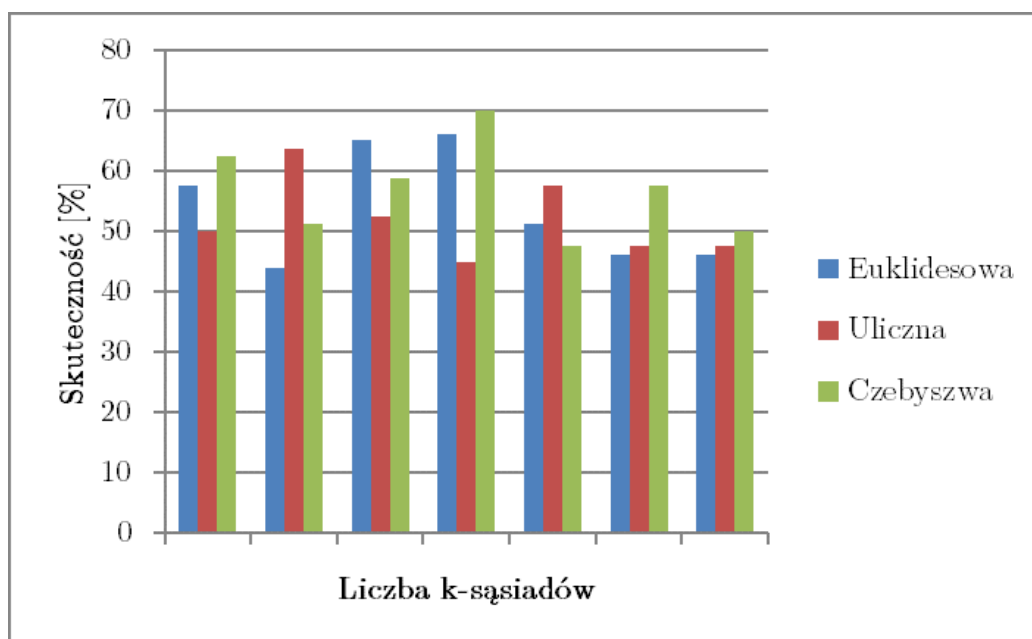
Rysunek 9. Dane z tabel 1-9 dla kategorii topics, zbiór treningowy 20%, zbiór testowy 80%



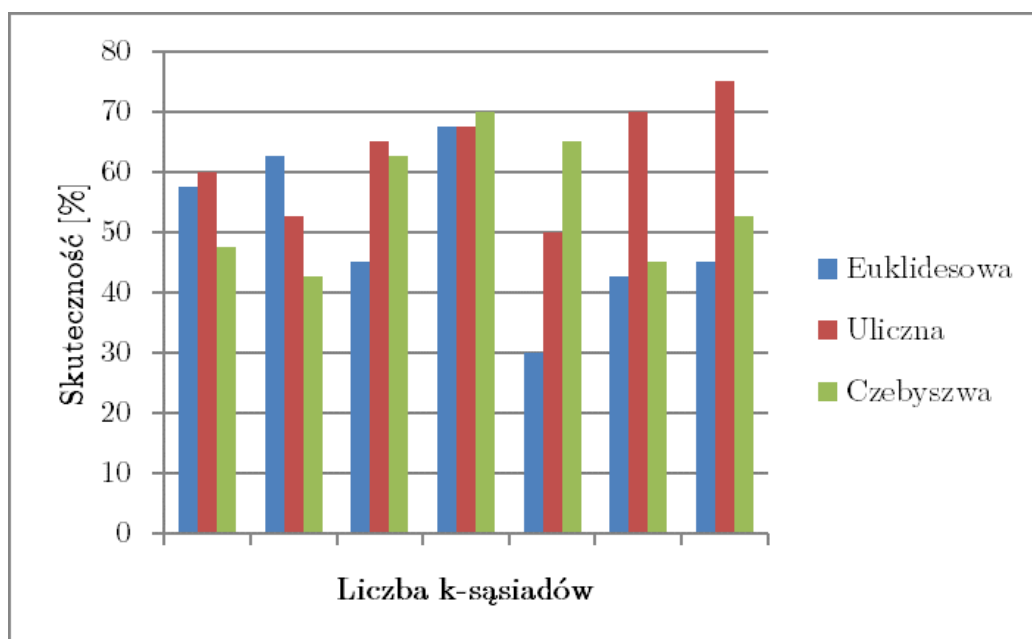
Rysunek 10. Dane z tabel 1-9 dla kategorii topics, zbiór treningowy 60%, zbiór testowy 40%



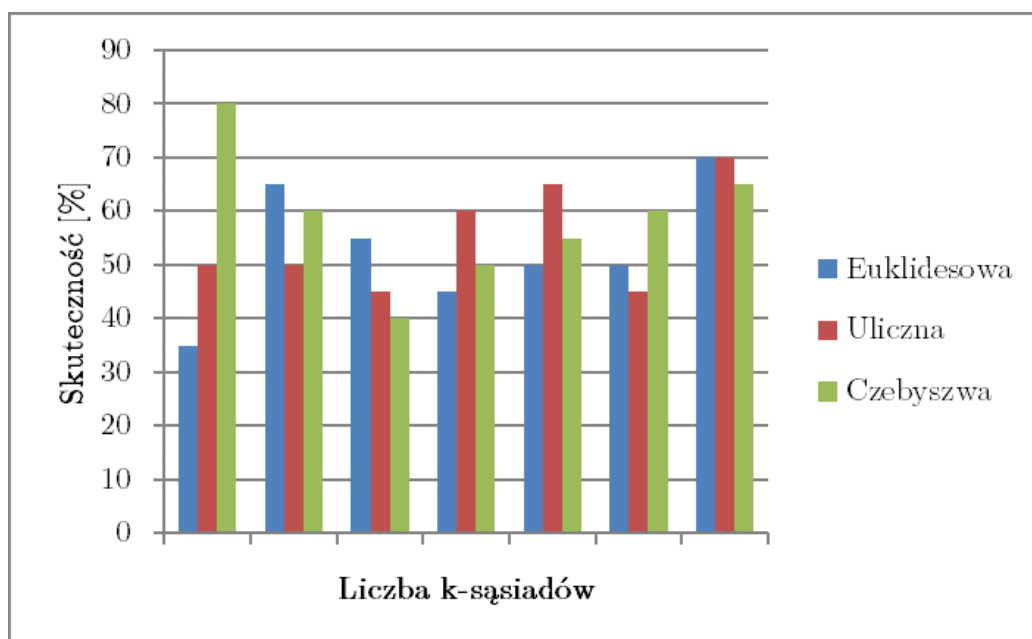
Rysunek 11. Dane z tabel 1-9 dla kategorii topics, zbiór treningowy 80%, zbiór testowy 20%



Rysunek 12. Dane z tabel 1-9 dla kategorii medium, zbiór treningowy 20%, zbiór testowy 80%



Rysunek 13. Dane z tabel 1-9 dla kategorii medium, zbiór treningowy 60%, zbiór testowy 40%



Rysunek 14. Dane z tabel 1-9 dla kategorii medium, zbiór treningowy 80%, zbiór testowy 20%

## 6. Dyskusja

Sekcja ta powinna zawierać dokładną interpretację uzyskanych wyników eksperymentów wraz ze szczegółowymi wnioskami z nich płynącymi. Najcenniejsze są, rzecz jasna, wnioski o charakterze uniwersalnym, które mogą być

istotne przy innych, podobnych zadaniach. Należy również omówić i wyjaśnić wszystkie napotkane problemy (jeśli takie były). Każdy wniosek powinien mieć poparcie we wcześniej przeprowadzonych eksperymentach (odwołania do konkretnych wyników). Jest to jedna z najważniejszych sekcji tego sprawozdania, gdyż prezentuje poziom zrozumienia badanego problemu.

## 7. Wnioski

W tej, przedostatniej, sekcji należy zamieścić podsumowanie najważniejszych wniosków z sekcji poprzedniej. Najlepiej jest je po prostu wypunktować. Znow, tak jak poprzednio, najistotniejsze są wnioski o charakterze uniwersalnym.

## Literatura

- [1] Methods for the linguistic summarization of data - applications of fuzzy sets and their extensions, Adam Niewiadomski, Akademicka Oficyna Wydawnicza EXIT, Warszawa 2008
- [2] <https://github.com/hklemp/dotnet-stop-words>
- [3] <http://snowball.tartarus.org/algorithms/english/stemmer.html>
- [4] <https://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection>
- [5] <https://github.com/jurczewski/KSR/blob/master/Zad1/Zad1/Data/ours.sgm>