Informatyka, studia STACJONARNE, I st. semestr VI

Komputerowe systemy rozpoznawania **2020/2021**

Prowadzący: dr inż. Marcin Kacprowicz poniedziałek, 12:00

Data oddania: _____ Ocena: _____

Maciej Lewandowski 224357
Kamil Dike 224282

Projekt 1. Klasyfikacja dokumentów tekstowych

Opis projektu ma formę artykułu naukowego lub raportu z zadania badawczego/doświadczalnego/obliczeniowego (wg indywidualnych potrzeb związanych np. z pracą inżynierską/naukową/zawodową).

Wybrane sekcje (rozdziały sprawozdania) są uzupełniane wg wymagań w opisie Projektu 1. i Harmonogramie ZAJĘĆ na WI-KAMP KSR jako efekty zadań w poszczególnych tygodniach.

1. Cel projektu

Celem zadania jest stworzenie systemu klasyfikującego teksty w zależności od kraju który jest opisywany przez tekst. System został oparty o metodę k-NN. Ponad to została przeanalizowana skuteczność działania programu w odniesieniu do nietraktowanego wektora cech.

2. Klasyfikacja nadzorowana metodą k-NN

Metoda k-NN służy do klasyfikacji obiektów. Opiera się na założeniu podobieństwa obiektów blisko położonych w przestrzeni cech. Jak podaje założenia dla algorytmu:
$numclass$ - liczba rozpoznawanych klas
dim - wymiar przestrzeni cech
num - liczba obiektów ciągu uczącego
$sampl[1...num][1...dim+1]$ - ciąg uczący
rec - identyfikator rozpoznanego obiektu

$obj[1...dim]$ - rozpoznawany obiekt
$dist(sampl[k], obj)$ - funkcja podająca odległość między i-tym elementem ciągu uczącego a rozpoznawanym obiektem
k - zmienna określająca ilość uwzględnianych sąsiadów
$tab[1...num][1...2]$ - tablica odległości
$sort(tab)$ - funkcja sortująca tablicę
$fun[1..numclass]$ - tabela wartości funkcji przynależności
$pointmax(fun)$ - funkcja wskazująca numer klasy, dla której wartość przynależności jest maksymalna. Algorytm według składa się z następujących kroków:

```
1.  wyzeruj tablicę fun
2.  wykonaj pętlę s od i=1 do num
3.  w pętli s przyporządkuj elementowi tablicy tab[i][1]
    wynik wywołania funkcji dist(sampl[i],obj)
4.  w pętli s przyporządkuj elementowi tablicy tab[i][2]
    element tablicy sampl[i][dim+1]
5.  zakończ pętlę s
6.  wykonaj sortowanie tablicy sort(tab)
7.  rozpocznij pętlę q od i=1 do i=k
8.  w pętli q przyporządkuj elementowi tablicy fun[tab[i][2]]
    element tablicy fun[tab[i][2]]+1
9.  zakończ pętlę q
10. przyporządkuj zmiennej rec wynik funkcji pointmax(fun)
```

Algorytm jako dane wejściowe pobiera obiekt do klasyfikacji obj oraz zmienną przechowującą informacje o klasie do którego zostanie przyporządkowany rec.Odległość dwóch obiektów określana jest poprzez określoną metrykę. Porównywane będą wektory cech reprezentujące obiekty.

2.1. Ekstrakcja cech, wektory cech

Na potrzeby reprezentacji obiektów poprzez wektory cech wybrano cechy:

1. Liczba słów w dokumencie
$$v_1 = \hat{A} \tag{1}$$
,gdzie
A oznacza artykuł taki, że $A = [s_1, s_2, s_3, ..., s_T]$
s_i oznacza i- te słowo w artykule
$\hat{A}$ oznacza moc zbioru A
2. Wartość logiczna z logiki trój-wartościowej określająca dominujący rodzaj jednostek występujących w tekście. Wartość cechy 1 oznacza że dominują w artykule jednostki układu SI. Wartość cechy 0 oznacza że w artykule dominują jednostki układu Imperialnego. Wartość cechy 1/2 oznacza że w artykule nie dominują jednostki układu SI anie jednostki układu imperialnego.
$$v_2 = l(A) \tag{2}$$
,gdzie
$l : \mathcal{A} \to \{0, \frac{1}{2}, 1\}$, l funkcja przyporządkowywuje artykułowi wartość lo-

giczną 0, 1/2 albo 1 w zależności od ilości wystąpień jednostek danego typu(si/imperialne).
$\mathcal{A}$ oznacza zbiór wszystkich możliwych wektorów reprezentujących artykuły.

3. Najczęściej występujący miesiąc

$$v_3 = m(A) \tag{3}$$

,gdzie
$m : \mathcal{A} \to \{0, 1, 2, ..., 12\}$, m funkcja przyporządkowywująca artykułowi wartość całkowitą od 0 do 12, w zależności od ilości wystąpień danego miesiąca w zbiorze A.

4. Najczęściej występujący typ spółki/firmy

$$v_4 = f(max(k(A, G_S))) \tag{4}$$

,gdzie
$\mathcal{G}$ zbiór wszystkich możliwych wektorów słów kluczowych
$G_S = [x_1, x_2, x_3, ..., x_j]$ wektor słów kluczowych rodzajów spółek
x_i oznacza i-te słowo kluczowe
$\mathcal{H}$ zbiór wszystkich możliwych wektorów częstości występowania słów kluczowych
H wektor częstości występowania słów kluczowych
$f : \mathcal{H} \to \mathcal{G}$, f jest funkcją przyporządkowującą zbiór częstości do zbioru słów kluczowych
$k : \mathcal{A}, \mathcal{G} \to \mathcal{H}$, k jest funkcją zwracającą wektor częstości dla zapewnionego artykułu oraz wektora słów kluczowych

5. Najczęściej występująca w tekście nazwa giełdy

$$v_5 = f(max(k(A, G_G))) \tag{5}$$

,gdzie
$G_g = [x_1, x_2, x_3, ..., x_j]$ wektor słów kluczowych nazw giełd

6. Najczęściej występująca nazwa morza lub oceanu

$$v_6 = f(max(k(A, G_M))) \tag{6}$$

,gdzie
$G_M = [x_1, x_2, x_3, ..., x_j]$ wektor słów kluczowych nazw mórz i oceanów

7. Względna ilość słów o długości do 4 znaków

$$v_7 = \frac{c(A, 0, 4)}{v_1} \tag{7}$$

,gdzie
$c : \mathcal{A}, N, M \to P$ c jest funkcją zliczającą ilość słów o długości od n do m znaków
$N = \{n : n \in \mathrm{N} \wedge n > 0\}$
$M = \{m : m \in \mathrm{N} \wedge m > n\}$
$P = \{p : p \in \mathrm{N}\}$

8. Względna ilość słów o długości od 4 do 8 znaków

$$v_8 = \frac{c(A, 4, 8)}{v_1} \tag{8}$$

9. Względna ilość słów o długości od 8 znaków

$$v_9 = \frac{c(A, 8, \infty)}{v_1} \tag{9}$$

10. Najczęściej występujący rok w artykule

$$v_{10} = yr(A) \tag{10}$$

,gdzie
$yr : \mathcal{A} \to \mathcal{P}$, yr to funkcja zwracająca najczęściej występującą datę w tekście

11. Ilość cen w tekście

$$v_{11} = dl(A) \tag{11}$$

,gdzie
$dl : \mathcal{A} \to \mathcal{P}$, yr to funkcja zwracająca najczęściej występujący rok w tekście

12. Liczba unikalnych słów

$$v_{12} = us(A) \tag{12}$$

,gdzie
$us : \mathcal{A} \to \mathcal{P}$, yr to funkcja zwracająca ilość różnych słów w tekście

2.2. Miary jakości klasyfikacji

Celem miar jakości klasyfikacji jest zbadanie dokonanej klasyfikacji. Ze względu na brak miary idealnej posłużymy się paroma następującymi miarami:

1. accuracy
2. precision
3. recall
4. F1

Do wyznaczenia miar jakości klasyfikacji korzystamy z tablicy pomyłek. Spis oznaczeń:
TP - prawdziwie pozytywna klasyfikacja
FP - fałszywie pozytywna klasyfikacja
FN - fałszywie negatywna klasyfikacja
TN - prawdziwie negatywna klasyfikacja

TP *Przykład dla TP* Jeśli obiekt A w rzeczywistości należy do klasy $\mathcal{A}$, i zostanie sklasyfikowany do klasy $\mathcal{A}$ wówczas klasyfikacja jest uznawana za prawdziwie pozytywną TP.

FP *Przykład dla FP* Jeśli obiekt A w rzeczywistości nie należy do klasy $\mathcal{A}$, i zostanie sklasyfikowany do klasy $\mathcal{A}$ wówczas klasyfikacja jest uznawana za fałszywie pozytywną FP.

FN *Przykład dla FN* Jeśli obiekt A w rzeczywistości należy do klasy $\mathcal{A}$, i nie zostanie sklasyfikowany do klasy $\mathcal{A}$ wówczas klasyfikacja jest uznawana za fałszywie negatywną FN.

TN *Przykład dla TN* Jeśli obiekt A w rzeczywistości nie należy do klasy $\mathcal{A}$, i nie zostanie sklasyfikowany do klasy $\mathcal{A}$ wówczas klasyfikacja jest uznawana za prawdziwie negatywną TN.

2.2.1. Accuracy

Dokładność określa sprawność klasyfikatora. Miara ta jest wspólna dla wszystkich klas. Dokładność wyraża się wzorem:

$$ACC = \frac{\Sigma TP}{\Sigma populacja} \tag{13}$$

2.2.2. Precision

Precyzja jest miarą liczoną dla danej klasy. Miara ta określa precyzje rozpoznawania w obrębie konkretnej klasy. Precyzja wyraża się wzorem:

$$PPV = \frac{\Sigma TP}{\Sigma TP + \Sigma FP} \tag{14}$$

2.2.3. Recall

Recall jest miarą liczoną dla danej klasy. Miara ta określa ilość rozpoznanych elementów danej klasy. Czułość wyraża się wzorem:

$$TPR = \frac{\Sigma TP}{\Sigma TP + \Sigma FN} \tag{15}$$

2.2.4. F1

F1 jest miarą liczoną dla danej klasy. Liczona jest na podstawie miar precision oraz recall jako ich średnia harmoniczna. Miarę F1 wyraża się wzorem:

$$TPR = 2 * \frac{PPV * TPR}{PPV + TPR} \tag{16}$$

3. Klasyfikacja z użyciem metryk i miar podobieństwa tekstów

Program umożliwia wybór trzech metryk do porównywania wektorów cech: metryki miejskiej, amplitudy kosinusowej oraz odległości euklidesowej. Metryka miejska

$$\rho_C(X, Y) = \Sigma^n{}_{i=1}|x_i - y_i| \tag{17}$$

Amplituda kosinusowa

$$r_{ca}(V_1, V_2) = \frac{|\Sigma^n{}_{i=1} v_{1i} \cdot v_{2i}|}{\sqrt{\Sigma^n{}_{i=1} v_{1i}^2 \cdot \Sigma^n{}_{i=1} v_{2i}^2}} \tag{18}$$

Odległość euklidesowa

$$\rho_E(X, Y) = \sqrt{\Sigma^n{}_{i=1}(x_i - y_i)^2} \tag{19}$$

Jako metrykę tekstową do ekstrakcji cech tekstowych zastosowano metodę *trigramów*, opisaną poniższym równaniem.

$$sim_3(s_1, s_2) = \frac{1}{N-2}\Sigma^{N-2}{}_{i=1}h(i) \tag{20}$$

, gdzie
$N-2$ - ilość możliwych trój-elementowych podciągów.
$h(i) = 1$ - jeśli trój-elementowy podciąg zaczynający się od i-tej pozycji w s_1 występuje przynajmniej raz w s_2, w innym przypadki $h(i) = 0$.

3.0.1. Wstępne wyniki miary accuracy

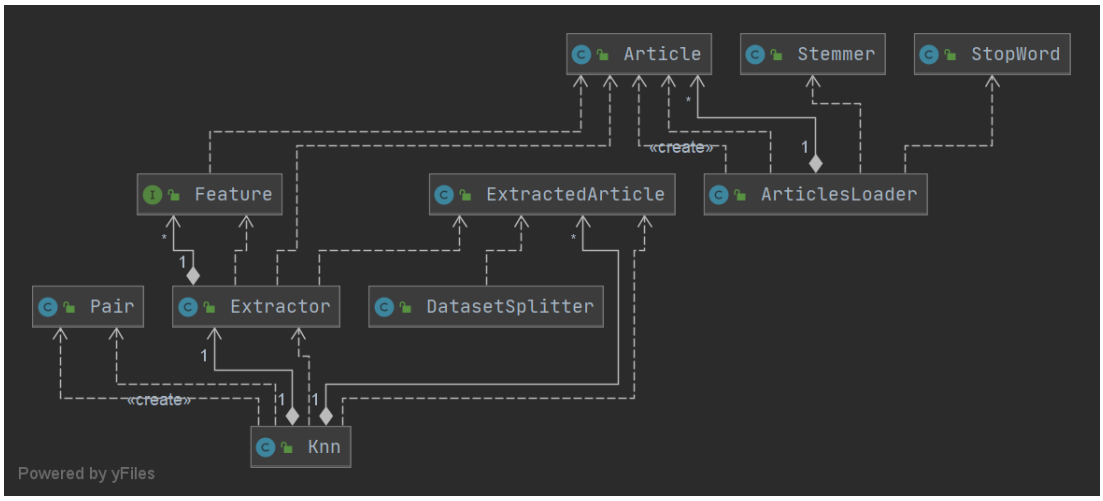
Uruchomiono program w czterech konfiguracjach, we wszystkich zastosowane metrykę euklidesową:
Dla zbioru uczącego stanowiącego $60proc$ oraz dla 17 sąsiadów $ACC = 0.7666$.
Dla zbioru uczącego stanowiącego $60proc$ oraz dla 3 sąsiadów $ACC = 0.7693$.
Dla zbioru uczącego stanowiącego $30proc$ oraz dla 17 sąsiadów $ACC = 0.7913$.
Dla zbioru uczącego stanowiącego $30proc$ oraz dla 3 sąsiadów $ACC = 0.7379$.

4. Budowa aplikacji

4.1. Diagramy UML

4.1.1. Struktura aplikacji

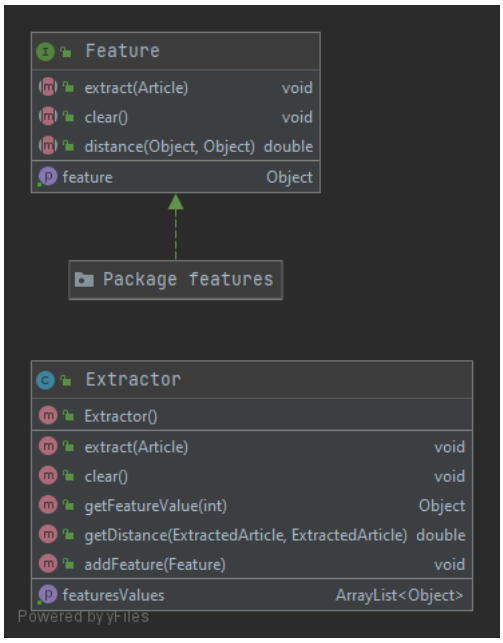
Aplikacja złożona jest z komponentów: extractor, features, knn, main, model, parser, utils. Struktura aplikacji została przedstawiona na rysunku 4.1.1 na stronie 6.



Rysunek 1. Struktura aplikacji

4.1.2. extractor

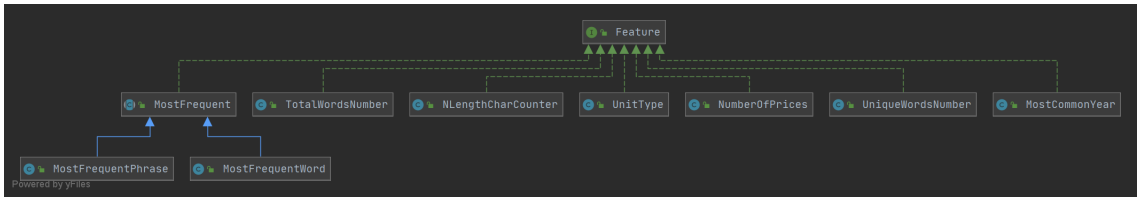
Pakiet ekstraktor udostępnia narzędzia umożliwiające ekstrakcję cech z tekstów. Pakiet ekstraktor zaprezentowano na rysunku 4.1.2 na stronie 7.



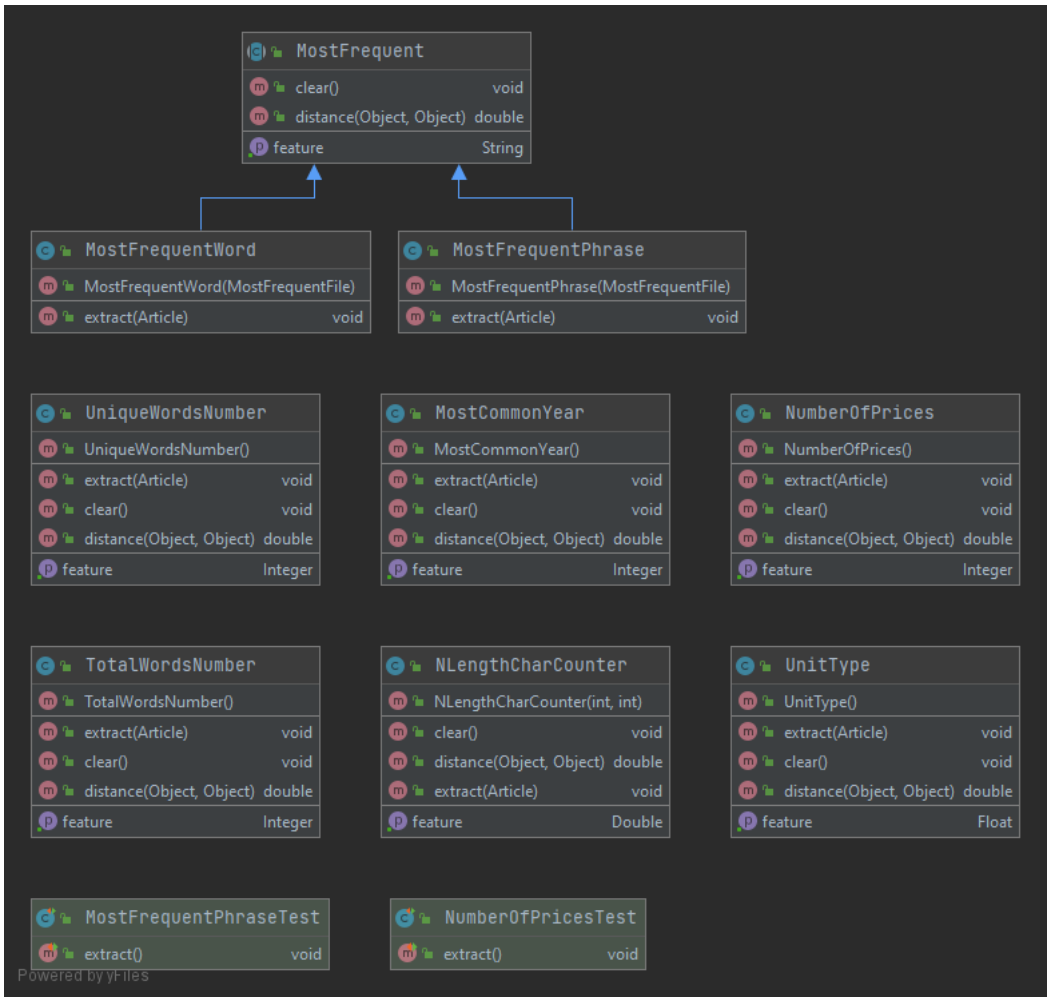
Rysunek 2. Pakiet extrator

4.1.3. features

Pakiet features udostępnia abstrakcję cechy oraz modeluje logikę konkretnych cech. Pakiet features zaprezentowano na rysunku 4.1.3 oraz 4.1.3 na stronie 7.



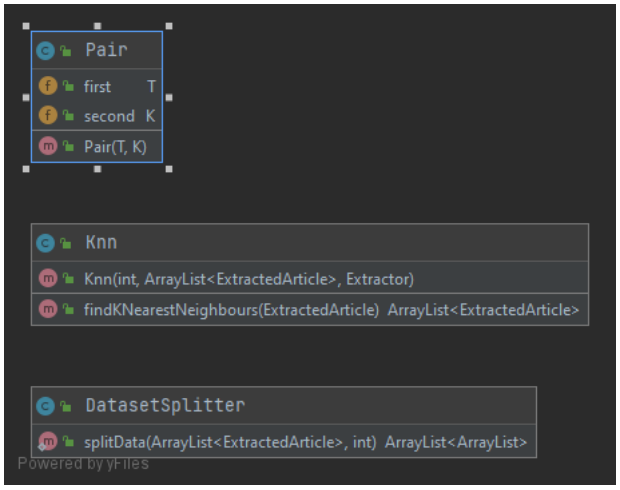
Rysunek 3. Struktura pakietu features



Rysunek 4. Pakiet features

4.1.4. knn

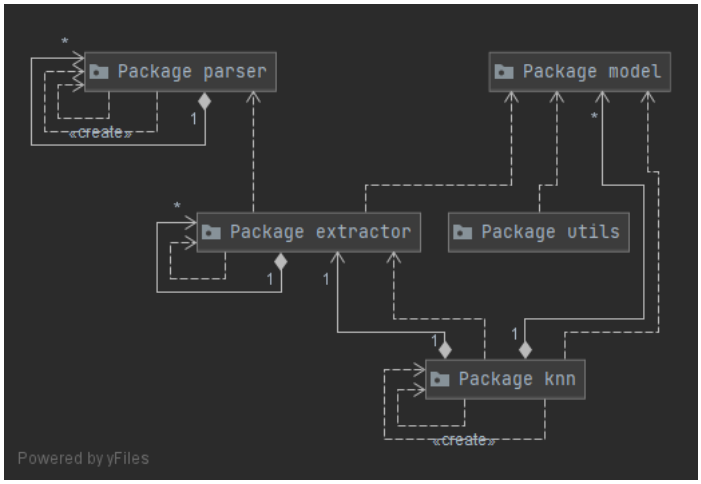
Pakiet knn udostępnia algorytm $k - NN$. Pakiet knn zaprezentowano na rysunku 4.1.4 na stronie 8.



Rysunek 5. Pakiet knn

4.1.5. main

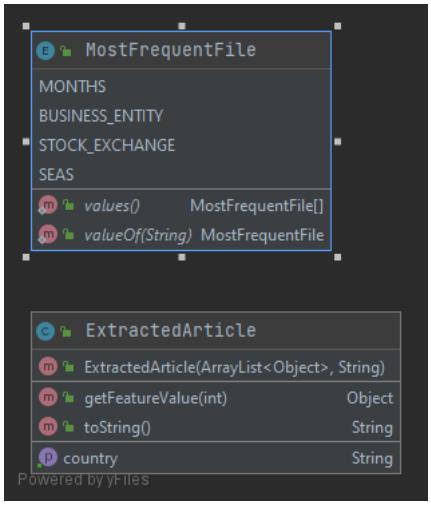
Pakiet main stanowi wejście, oraz implementuje logikę CLI. Pakiet main zaprezentowano na rysunku 4.1.5 na stronie 9.



Rysunek 6. Pakiet main

4.1.6. model

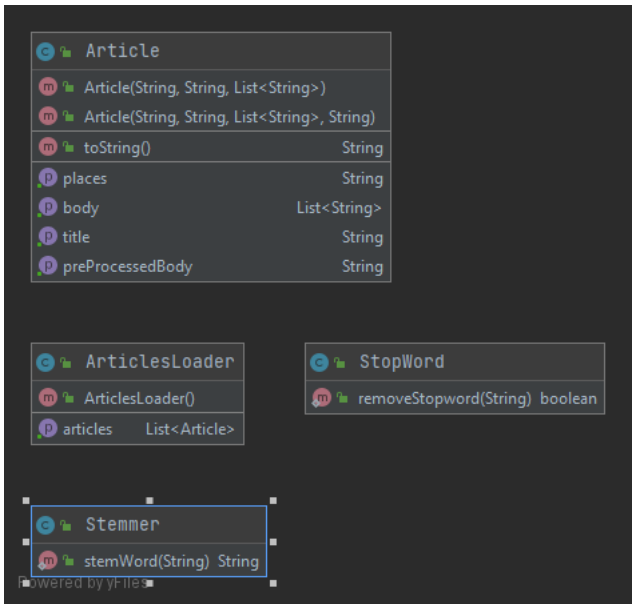
Pakiet model dostarcza model danych dla artykułu reprezentowanego jako wektor cech oraz pozostałe wymagane modele danych. Pakiet model zaprezentowano na rysunku 4.1.6 na stronie 9.



Rysunek 7. Pakiet model

4.1.7. parser

Pakiet parser udostępnia metody odpowiedzialne za przygotowanie tekstu przed ekstrakcją cech. Pakiet parser zaprezentowano na rysunku 4.1.7 na stronie 10.



Rysunek 8. Pakiet parser

4.1.8. utils

Pakiet utils udostępnia narzędziowe metody wykorzystywane przez pozostałe pakiety. Pakiet utils zaprezentowano na rysunku 4.1.8 na stronie 10.

Rysunek 9. Pakiet utils

4.2. Prezentacja wyników, interfejs użytkownika

4.2.1. Interfejs użytkownika

Interfejs użytkownika jest w formie tekstowej. Poniżej pokazano przykładowe uruchomienie programu przy udziale zbioru treningowego 60*proc*, siedmiu sąsiadów, do klasyfikacji wykorzystano wszystkie kraje, oraz użyto wszystkich cech opisanych wyżej.

```
Okresl procent zbioru treningowego:
60
Okresl ilosc sasiadow KNN:
17
Wybierz interesujące kraje:
0. Wszystkie
1. USA
2. UK
3. Japan
4. Canada
5. West-Germany
```

```
6. France
Y. Przejdz dalej
0
Wybierz interesujące metryki:
0. Wszystkie
1. TotalWordsNumber
2. unitType
3. shortWords
4. middleWords
5. longWords
6. mostCommonYear
7. mostFrequentWordMonth
8. mostFrequentWordBusinessEntity
9. mostFrequentStockExchange
10. mostFrequentSea
11. numberOfPrices
12. uniqueWordsNumber
Y. Przejdz dalej
0
loading...
```

Wynik wygenerowany przez program będzie następujący:

```
Dla usa _________
Dokladnosc: 0.766563241467421
Precyzja: 0.789354044326577
Czulosc: 0.9643601018282805
F1: 0.8681249999999999


Dla uk _________
Dokladnosc: 0.766563241467421
Precyzja: 0.165
Czulosc: 0.0889487870619946
F1: 0.11558669001751312


Dla japan _________
Dokladnosc: 0.766563241467421
Precyzja: 0.5
Czulosc: 0.0
F1: 0.0


Dla canada _________
Dokladnosc: 0.766563241467421
Precyzja: 0.5
Czulosc: 0.0
F1: 0.0
```

```
Dla west-germany _________
Dokladnosc: 0.766563241467421
Precyzja: 0.5
Czulosc: 0.0
F1: 0.0


Dla france _________
Dokladnosc: 0.766563241467421
Precyzja: 0.5
Czulosc: 0.0
F1: 0.0
```

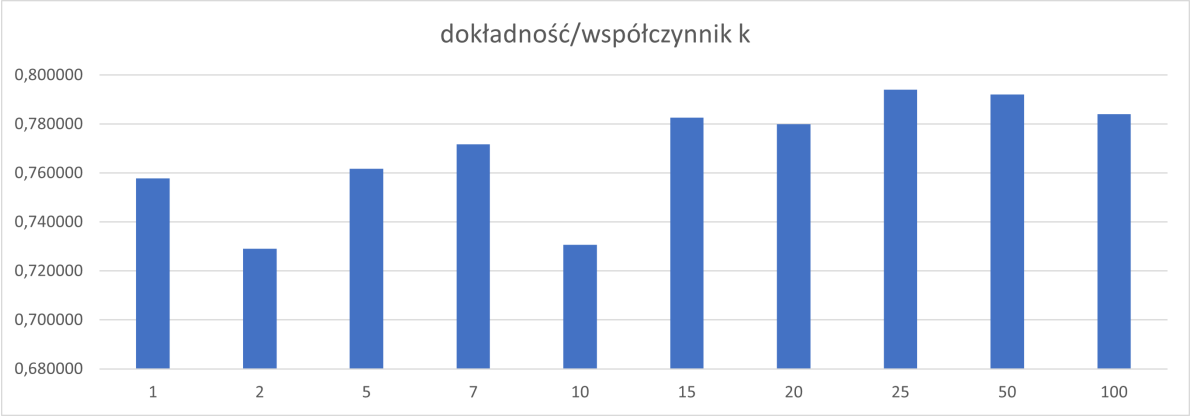
5. Wyniki klasyfikacji dla różnych parametrów wejściowych

5.1. Wpływ ilości sąsiadów

Wykonano obliczenia dla dziesięciu wartości ilości sąsiadów z wykorzystaniem metryki euklidesowej oraz podziałem zbioru uczącego 60/40. Wykres na stronie 13 przedstawia wyniki obliczeń. Wyniki zawarto w tabeli na stronie 12. Wartości precyzji, f1, czułości w tabeli są średnią arytmetyczną dla każdej z klasy.

k	accuracy	precission	recall	f1
1	0.7578	0.3313	0.1678	0.1550
2	0.7281	0.3149	0.1725	0.1593
5	0.7618	0.4047	0.1671	0.1481
7	0.7717	0.5094	0.1728	0.1603
10	0.7306	0.5553	0.1742	0.1616
15	0.7826	0.4938	0.1744	0.1626
20	0.7799	0.5746	0.1769	0.1664
25	0.7941	0.5140	0.1737	0.1609
50	0.7921	0.5718	0.1724	0.1588
100	0.7841	0.4913	0.1678	0.1508

Tabela 1. Tabela przedstawiająca zależność dokładności od ilości sąsiadów



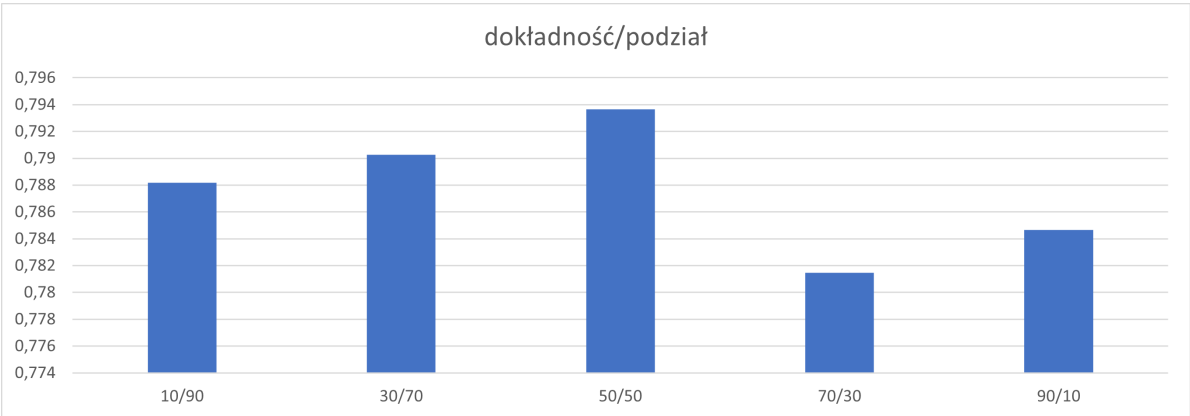
Rysunek 10. Wykres przedstawiający zależność dokładności od ilości sąsiadów

5.2. Wpływ proporcji podziału zbioru uczącego

Wykonano obliczenia dla pięciu podziału zbioru uczącego z wykorzystaniem metryki euklidesowej oraz ilości sąsiadów 25. Wykres na stronie 13 przedstawia wyniki obliczeń. Wyniki zawarto w tabeli na stronie 13.

proporcja podziału	accuracy	precission	recall	f1
10/90	0.7882	0.4742	0.1665	0.1476
30/70	0.7903	0.5236	0.1734	0.1608
50/50	0.7937	0.5570	0.1742	0.1619
70/30	0.7815	0.5645	0.1813	0.1741
90/10	0.7847	0.5082	0.1771	0.1674

Tabela 2. Tabela przedstawiająca zależność dokładności od podziału zbioru uczącego



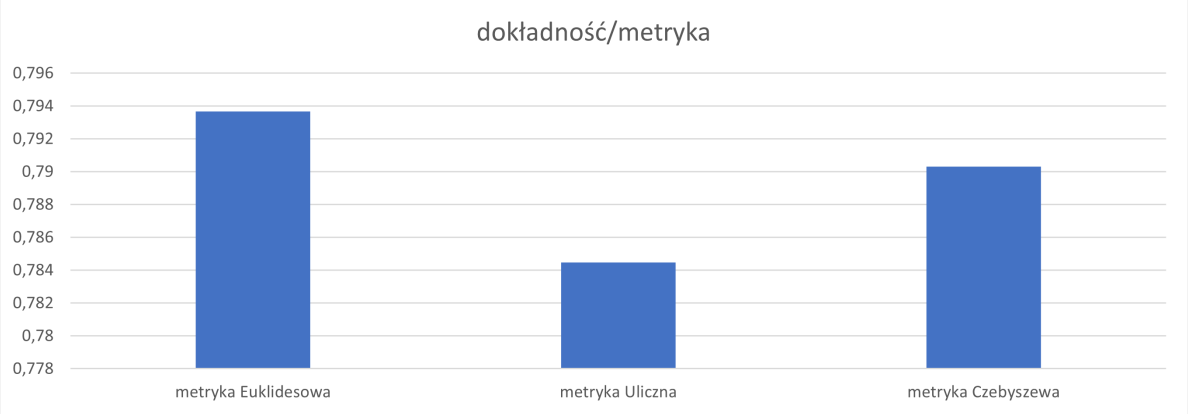
Rysunek 11. Wykres przedstawiający zależność dokładności od podziału zbioru uczącego

5.3. Wpływ metryki

Wykonano obliczenia dla trzech metryk z ilością sąsiadów $k = 25$ oraz podziale zbioru uczącego 60/40. Wykres na stronie 14 przedstawia wyniki obliczeń. Wyniki zawarto w tabeli na stronie 14.

metryka	accuracy	precission	recall	f1
Euklidesowa	0.7934	0.5512	0.1763	0.1660
Uliczna	0.7844	0.5139	0.1687	0.1534
Czebyszewa	0.7903	0.5479	0.1667	0.1469

Tabela 3. Tabela przedstawiająca zależność dokładności od metryki odległości



Rysunek 12. Wykres przedstawiający zależność dokładności od metryki odległości

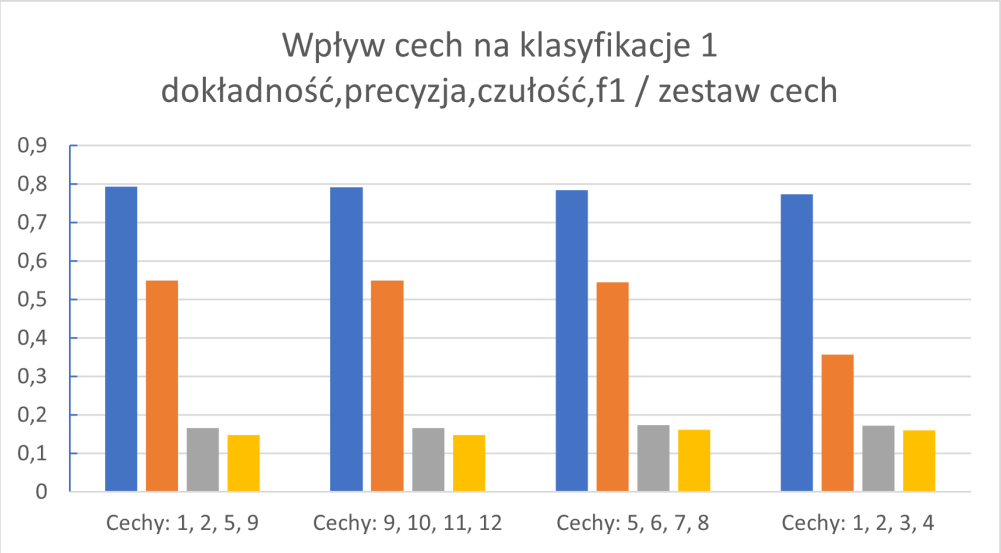
5.4. Wpływ poszczególnych cech

5.4.1. Pierwszy zestaw cech

Miary precyzji, czułości oraz f1 podano jako średnią ze wszystkich klas. Parametry wejściowe ustawiono jako: podział 60/40, $k = 25$ oraz metrykę euklidesową. Wyniki zawarto w tabeli na stronie 14. Wykres na stronie 15 przedstawia wyniki.

	cechy 1,2,5,9	cechy 9,10,11,12	cechy 5,6,7,8	cechy 1,2,3,4
dokładność	0.7934	0.7916	0.7848	0.7735
precyzja	0.5490	0.5486	0.5454	0.3568
czułość	0.1667	0.1667	0.1740	0.1714
f1	0.1475	0.1473	0.1609	0.1597

Tabela 4. Tabela przedstawiająca wyniki pierwszego zestawu cech. Parametry wejściowe ustawiono jako: podział 60/40, $k = 25$.



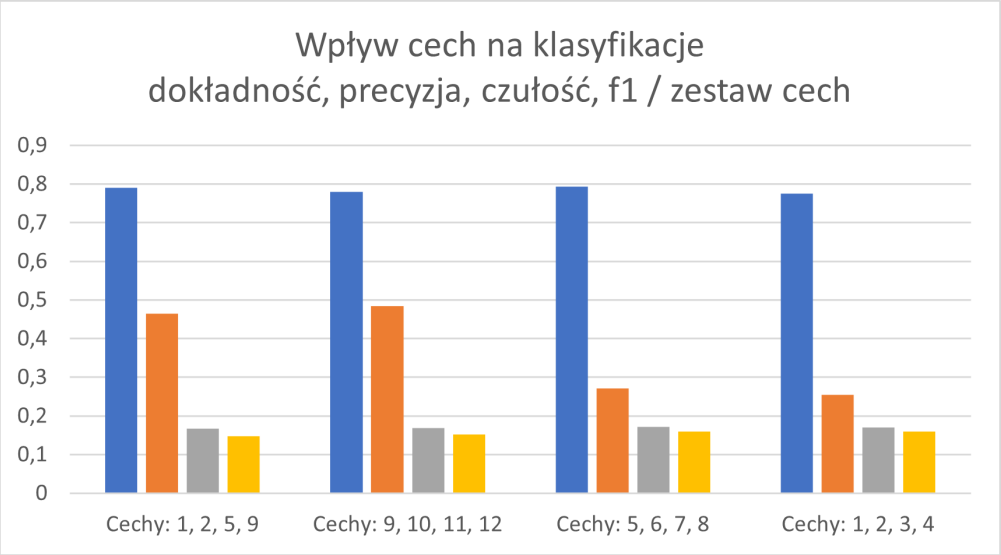
Rysunek 13. Wykres przedstawiający zależność dokładności od metryki odległości. Parametry wejściowe ustawiono jako: podział 60/40, $k = 25$.

5.4.2. Drugi zestaw cech

Miary precyzji, czułości oraz f1 podano jako średnią ze wszystkich klas. Parametry wejściowe ustawiono jako: podział 40/60, $k = 10$ oraz metrykę euklidesową. Wyniki zawarto w tabeli na stronie 15. Wykres na stronie 16 przedstawia wyniki.

	cechy 1,2,5,9	cechy 9,10,11,12	cechy 5,6,7,8	cechy 1,2,3,4
dokładność	0.7902	0.7790	0.7925	0.7750
precyzja	0.4650	0.4845	0.2712	0.2548
czułość	0.1667	0.1683	0.1724	0.1703
f1	0.1471	0.1521	0.1601	0.1590

Tabela 5. Tabela przedstawiająca wyniki drugiego zestawu cech. Parametry wejściowe ustawiono jako: podział 40/60, $k = 10$.



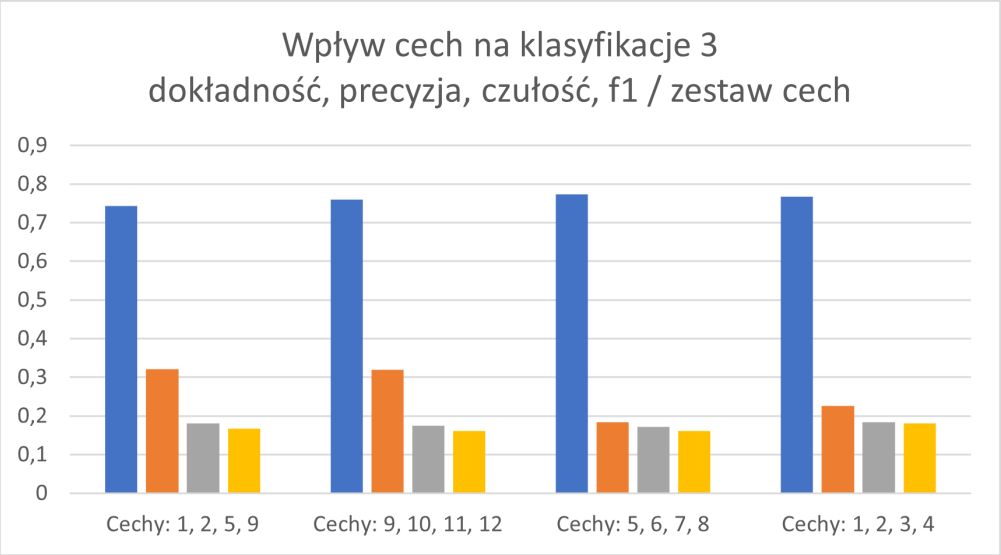
Rysunek 14. Wykres przedstawiający zależność dokładności od metryki odległości. Parametry wejściowe ustawiono jako: podział 40/60, $k = 10$.

5.4.3. Trzeci zestaw cech

Miary precyzji, czułości oraz f1 podano jako średnią ze wszystkich klas. Parametry wejściowe ustawiono jako: podział 20/80, $k = 3$ oraz metrykę euklidesową. Wyniki zawarto w tabeli na stronie 16. Wykres na stronie 17 przedstawia wyniki.

	cechy 1,2,5,9	cechy 9,10,11,12	cechy 5,6,7,8	cechy 1,2,3,4
dokładność	0.7439	0.7598	0.7737	0.7665
precyzja	0.3215	0.3193	0.1842	0.2257
czułość	0.1808	0.1743	0.1716	0.1831
f1	0.668 1	0.1618	0.1607	0.1809

Tabela 6. Tabela przedstawiająca wyniki drugiego zestawu cech. Parametry wejściowe ustawiono jako: podział 20/80, $k = 3$.



Rysunek 15. Wykres przedstawiający zależność dokładności od metryki odległości. Parametry wejściowe ustawiono jako: podział 20/80, $k = 3$.

6. Dyskusja, wnioski

6.1. Wpływ ilości sąsiadów

Szczytową wartość dokładności osiągnięto dla dwudziestu-pięciu sąsiadów. Wartość dokładności rosła wraz ze wzrostem ilości sąsiadów. Zarejestrowano przerwy w trendzie wzrostowym dla wartości dwóch oraz dziesięciu sąsiadów. Gdy ilość sąsiadów staje się bliska liczności grupy dokładność zaczyna spadać. Uogólniając wartość dokładności rośnie wraz ze wzrostem ilości sąsiadów do momentu w którym ilość sąsiadów jest bliska rozmiarowi danej grupy.

6.2. Wpływ proporcji podziału zbioru uczącego

Wartość dokładności rośnie w miarę zwiększania części zbioru uczącego. Trend wzrostowy załamuje się w okolicach pięćdziesięciu procent. Wartości dokładności są niższe dla podziałów ze zbiorem uczącym większym niż testowym. Uogólniając najwyższe wartości dokładności można osiągnąć przy stosunkowo równym podziale zbioru uczącego i testowego.

6.3. Wpływ metryki

Najwyższą wartość dokładności zanotowano dla metryki euklidesowej. Kolejna była metryka Czebyszewa. Najniższą wartość zanotowano dla metryki Ulicznej.

6.4. Wpływ poszczególnych cech

Największy spadek dla średniej wartości precyzji zanotowano dla drugiego zestawu cech. Testy z cechami 1,2,5,9 oraz 9,10,11,12 dały prawie dwukrotnie

większą precyzję niż testy dla cech 5,6,7,8 oraz 1,2,3,4. Na podstawie powyższego można wnioskować że cechy od 9-tej w górę, w szczególności cecha nr 9, mają większy wpływ na precyzję.

6.5. Wnioski ogólne z całego zadania

Problematyczne okazało się zastawianie danych dla całego eksperymentu ze względu na liczbę klas. Ostatecznie zdecydowano się zaprezentować wartości średnie ze wszystkich klas. Nie użyto średniej ważonej ponieważ średnia ważona z precyzji gdzie wagami jest liczność klas jest matematycznie równa dokładności.

Pierwotny wybór cech, w rzeczywistości był wyborem kierowanym intuicją. Dopiero po zbadaniu wpływu danej cechy na klasyfikację, możemy określić zasadność zastosowania danej cechy. Często cechy które wydawały się nam o istotnym znaczeniu, okazywały się mieć mały wpływ na ostateczną klasyfikację.

Kolejnym problemem uniemożliwiającym poprawną interpretację zadania okazał się podział zbioru artykułów na zaklasyfikowany i uczący. Po zdiagnozowaniu problemu wprowadziliśmy kontrolę poprawności podziału poprzez wymóg odpowiedniej ilości artykułów z każdego kraju.

Algorytm k - NN jest stosunkowo prostym i łatwym w implementacji algorytmem. Uzyskanie wysokiej dokładności w klasyfikacji polega na trafnym doborze cech oraz wymianie/poprawie tych cech gdy nie wnoszą one w klasyfikację dodatkowych informacji. Podczas stosowania algorytmu k - NN należy pamiętać aby celować w zbiór uczący podzielony w miarę równomiernie, oraz należy zadbać o to aby liczba sąsiadów nie zbliżała się do liczności najmniejszej z klas.

Jako dalszą pracę nad systemami rozpoznającymi i klasyfikującymi proponujemy system rekomendacji produktów w sklepie internetowym, który jest dostosowany adekwatnie do preferencji użytkownika.

7. Braki w realizacji projektu 1.

Zrealizowano wszystkie obowiązkowe elementy projektu.

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.