

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 ze strony <http://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection> 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 - ilość 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 reprezentowały 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 po dokonaniu redukcji.

2. Druga najczęściej występująca waluta - wybieramy drugą najczęściej występującą walutę z tekstu, ponieważ uważamy, że pierwszą najczęściej występującą walutą będzie dolar ze względu na jego powszechne zastosowanie. Do pobierania nazw walut wykorzystujemy dołączony plik gdzie znajduje się 27 różnych walut wraz z kodami jakimi reprezentowane są w pobieranych tekstach. Przykładem jest kod dla waluty Dolaru Amerykańskiego - DLR. Cechę traktujemy jako cechę tekstową. Wartość będzie oznaczana poprzez symbol c_2 .
3. 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 .
4. Lokacja z tagu <Dateline>- jak wyżej. Lokację traktujemy jako cechę tekstową. Wartość będzie oznaczana poprzez symbol c_4 .
5. Tytuł z tagu <Title>- Każdy tekst w swoim body posiada tag <Title>. Tytuł traktujemy jako cechę tekstową. Wartość będzie oznaczana poprzez symbol c_5 .
6. Autor z tagu <Author>- Większość tekstów w swoim body posiada tag <Author>. Autora traktujemy jako cechę tekstową. Wartość będzie ozna-

czana poprzez symbol c_6 .

7. 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 .
8. Zbiór występujących słów kluczowych. Za słowa kluczowe przyjmujemy słowa znajdujące się w dołączonych plikach o rozszerzeniach .lc.txt. Cechę traktujemy jako cechę tekstową.

$$c_8 : c_8 \in N \cap t \quad (3)$$

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

9. Ilość wystąpień słów kluczowych - traktujemy jako cechę w postaci liczby całkowitej.

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

gdzie c_8 - zbiór występujących słów kluczowych

10. Nasycenie tekstu ilością słów kluczowych - traktujemy jako cechę w postaci liczby zmiennie przecinkowej.

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

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

11. Najczęściej występujące słowo kluczowe - wybieramy najczęściej występujące w analizowanym tekście słowo kluczowe. Cechę traktujemy jako cechę tekstową. Wartość będzie oznaczana poprzez symbol c_{11} .
12. 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:

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

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 błędów [3]. 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.

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

Tabela 1. Wzór tablicy pomyłek.

3. Klasyfikacja z użyciem metryk i miar podobieństwa tekstów

Wzory, znaczenia i opisy symboli zastosowanych metryk z przykładami. Wzory, opisy i znaczenia miar podobieństwa tekstów zastosowanych w obliczaniu metryk dla wektorów cech z przykładami dla każdej miary [2]. Oznaczenia jednolite w obrębie całego sprawozdania. Wstępne wyniki miary Accuracy dla próbnych klasyfikacji na ograniczonym zbiorze tekstów (podać parametry i kryteria wyboru wg punktów 3.-8. z opisu Projektu 1.).

Sekcja uzupełniona jako efekt zadania Tydzień 04 wg Harmonogramu Zajęć na WIKAMP KSR.

4. Budowa aplikacji

4.1. Diagramy UML

Diagramy UML i zwięzłe opisy: idei aplikacji, modułu ekstrakcji i modułu klasyfikatora.

Sekcja uzupełniona jako efekt zadania Tydzień 03 wg Harmonogramu Zajęć na WIKAMP KSR.

4.2. Prezentacja wyników, interfejs użytkownika

Krótki ilustrowany opis jak użytkownik może korzystać z aplikacji, w szczególności wprowadzać parametry klasyfikacji i odczytywać wyniki. Wersja JRE i inne wymogi niezbędne do uruchomienia aplikacji przez użytkownika na własnym komputerze.

Sekcja uzupełniona jako efekt zadania Tydzień 04 wg Harmonogramu Zajęć na WIKAMP KSR.

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] Tablica pomyłek:

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