Checkpoint 3

Warsztaty z technik uczenia maszynowego

Jakub Rymuza Karol Nowiński

26 maja 2022

1 Przygotowanie danych

Po "wyczyszczeniu" danych w poprzedniej checkpoint'cie, kolejnym krokiem mającym przygotować dane do efektywnego wykorzystania przez modele były: tokenizacja, stemizacja, lematyzacja oraz wektoryzacja.

1.1 Tokenizacja

Krok ten polega na podziale tekstu (ciągu słów) na tablicę pojedynczych słów. Wykorzystano dwie metody tokenizacji:

- Regexp Tokenization - ten typ tokenizacji przy podziale na słowa, wyrzuca wszelkie znaki interpunkcyjne.
- Treebank Tokenization - ten typ tokenizacji zachowuje wszystkie znaki interpunkcyjne.

1.2 Stemizacja

Stemizacja (ang. steeming) usuwa ze słów końcówki, zachowujący jedynie tzw. temat wyrazu (ang. stem). Temat wyrazu to część wspólna wszystkich wyrazów z danej rodziny. Na przykład tematem słowa "residents" jest "resid", a słowa "asked" jest "ask".

1.3 Lematyzacja

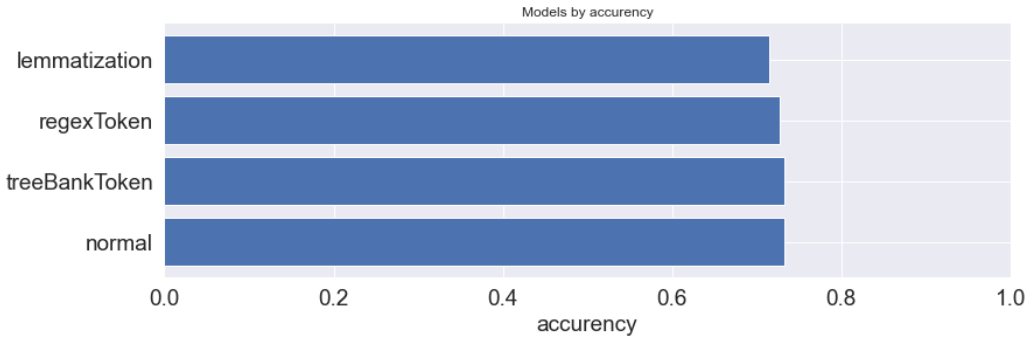
Lematyzacja (ang. lemmatization) przekształca słowa do podstawowej, "słownikowej" formy. Na przykład "is" oraz "are" są przekszłacane do "be", zaś "cars" do "car".

1.4 Wektoryzacja

Wektoryzacja to technika przekształcająca ciąg słów (tokenów) na tablicę liczb, którą wykorzystują modele. Wybrany typ wektoryzacji to wektoryzacja typu CountVectorizer - zlicza on po prostu ilość tokenów danego typu i zapisuje w rzadkiej macierzy. Innym typem wektoryzacji jest wektoryzacja TF-IDF, która bierze pod uwagę to jak rzadki jest dany token w tekście. Ta metoda jednak nie przyniosła lepszych wyników, a nawet je pogorszyło. Mogło to wynikać z wcześniejszych faz przygotowania danych (przede wszystkim usunięcie tzw. stop words).

1.5 Wyniki

Poniższy wykres przedstawia porównanie różnych metod tokenizacji, stemizacji oraz lematyzacji po użyciu ich na modelu klasyfikatora drzewa decyzyjnego. Jak widać zyski z zastosowania tych metod są dla naszych danych znikome, a wręcz mogą nieco pogarszać wyniki. Z tego względu w dalszej części będziemy rozważać wyniki dla których nie zastowano lematyzacji ani stemizacji.

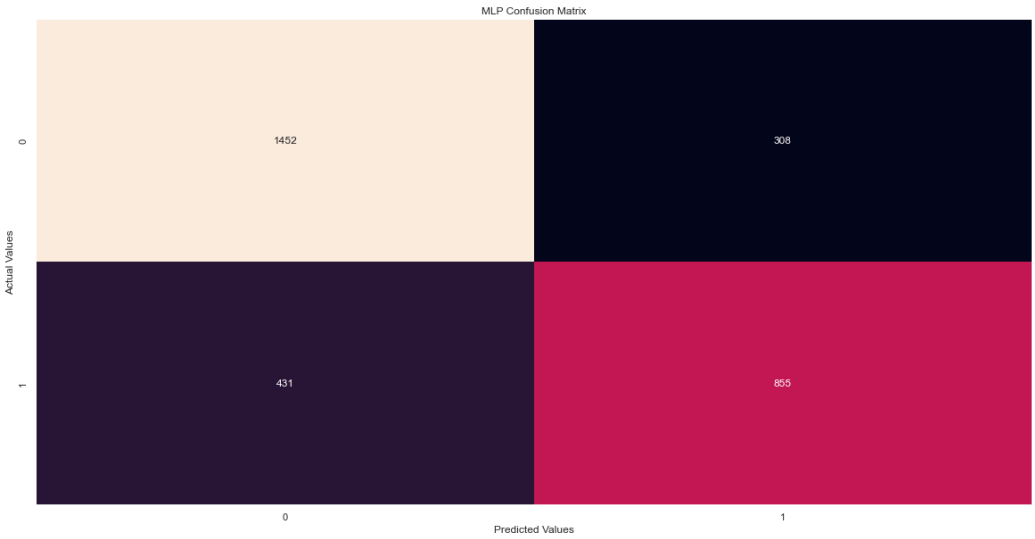


Rysunek 1: Porównanie metod tokenizacji, stemizacji oraz lematyzacji

2 Modele

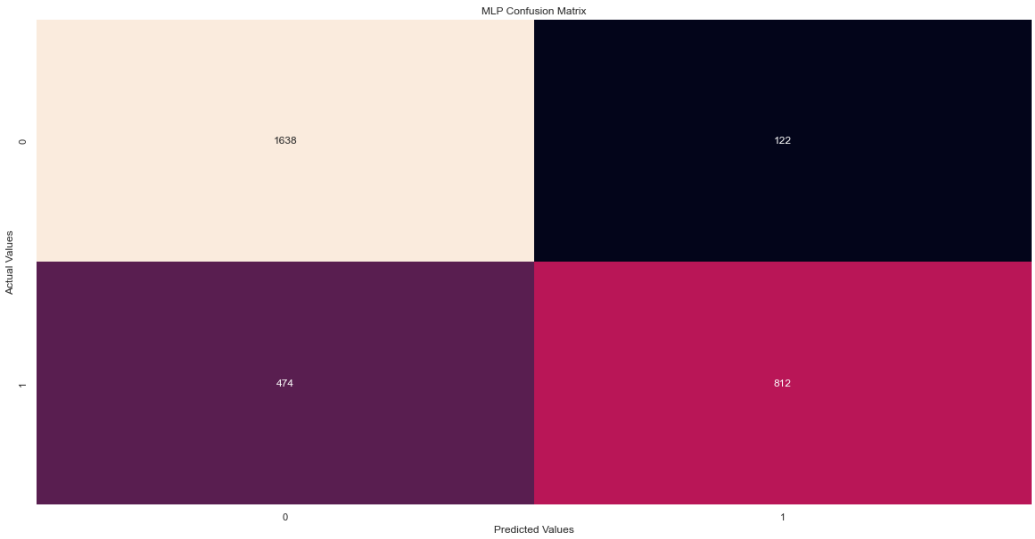
Po przygotowaniu danych, użyto różnych modeli w celu sprawdzeniu, który z nich jest najlepszy. Użyte modele wraz z dokładnością (accuracy) oraz tablicami pomyłek (confusion matrices):

- Klasyfikator drzewa decyzyjnego (decision tree classifier) - dokładność - 75.73%

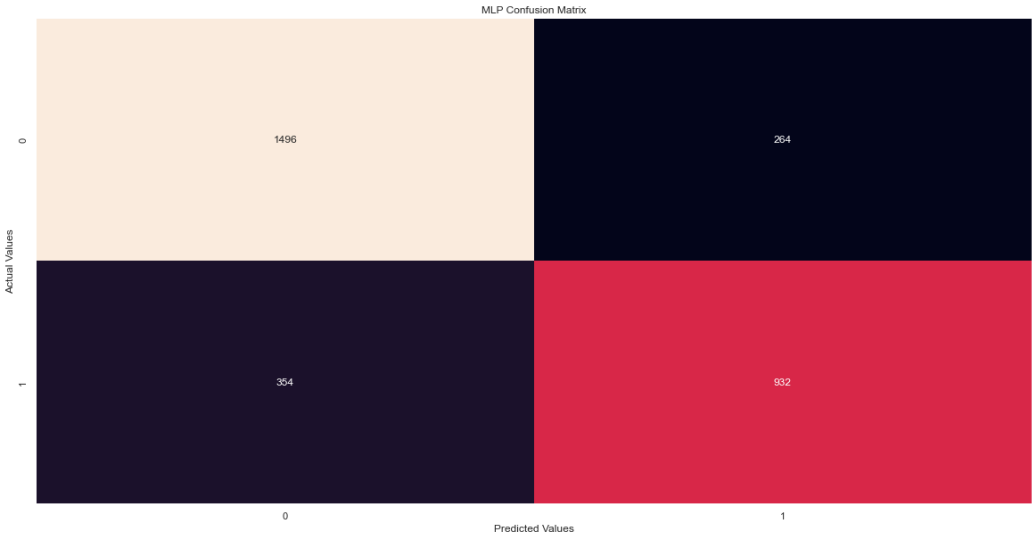


Rysunek 2: Tablica pomyłek dla klasyfikator drzewa decyzyjnego

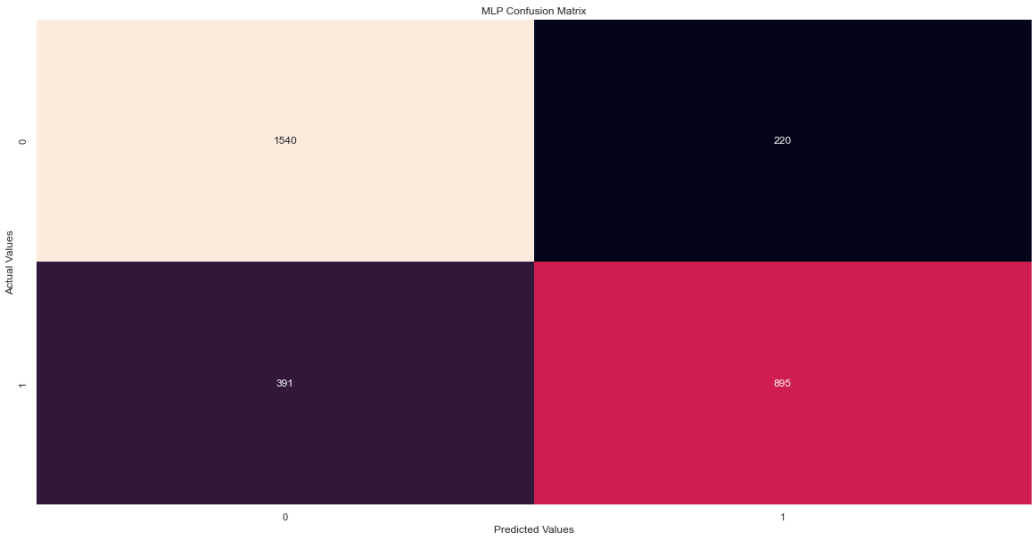
- Naiwny klasyfikator bayesowski (Naive Bayes classifier) - dokładność - 80.43%
- Wielomianowy naiwny klasyfikator bayesowski (multinomial naive Bayes classifier) - dokładność - 79.71%
- Klasyfikator regresji logistycznej (logistic regression classifier) - dokładność - 79.94%
- Klasyfikator regresji grzbietowej (ridge regression classifier) - dokładność - 77.61%
- Klasyfikator lasu losowego (random forest classifier) - dokładność - 78.92%



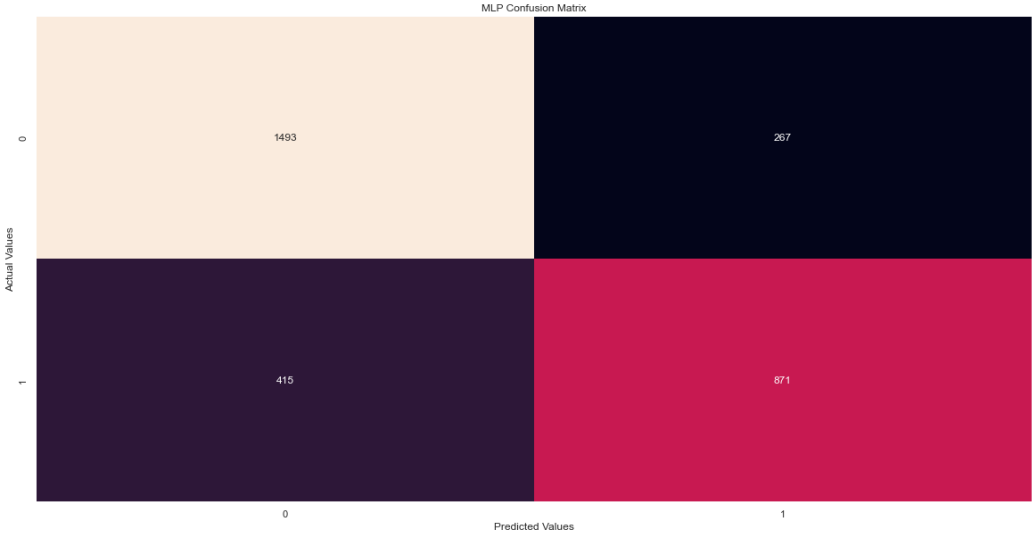
Rysunek 3: Tablica pomyłek dla naiwnego klasyfikatora bayesowskiego



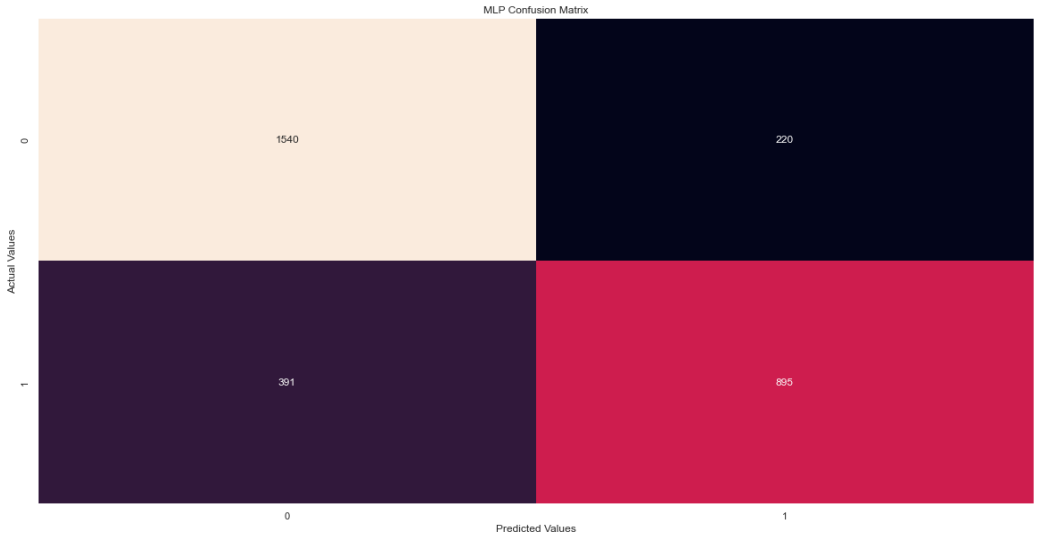
Rysunek 4: Tablica pomyłek dla wielomianowego naiwnego klasyfikatora bayesowskiego



Rysunek 5: Tablica pomyłek dla klasyfikatora regresji logistycznej



Rysunek 6: Tablica pomyłek dla klasyfikatora regresji grzbietowej



Rysunek 7: Tablica pomyłek dla klasyfikatora lasu losowego

Jak widać, najlepsze wyniki uzyskał naiwny klasyfikator bayesowski - powyżej 80%. Bardzo zbliżone wyniki uzyskał też klasyfikator regresjii logistycznej oraz wielomianowy naiwny klasyfikator bayesowski. Dokładność powyżej 80% można w uczeniu maszynowym można już traktować jako dość wysoki, zatem cel projektu można uznać za zrealizowany.