WSI - ćwiczenie 7.
Modele bayesowskie

14 stycznia 2026

1 Sprawy organizacyjne

1. Ćwiczenie realizowwane jest samodzielnie.
2. Ćwiczenie wykonywane jest w języku Python.
3. Ćwiczenie powinno zostać oddane najpóźniej na 14. zajęciach. W ramach oddawania ćwiczenia należy zademonstrować prowadzącemu działanie kodu oraz utworzyć pull request (z kodem oraz raportem), który prowadzący będzie mógł komentować.
4. Raport powinien być w postaci pliku .pdf, .html albo być częścią notebooka jupyterowego. Powinien zawierać opis eksperymentów, uzyskane wyniki wraz z komentarzem oraz wnioski.
5. Na ocenę wpływa poprawność oraz jakość kodu i dokumentacja.
6. Można korzystać z pakietów do obliczeń numerycznych, takich jak *numpy*
7. Implemenetacja algorytmu powinna być ogólna. W szczególności powinna być możliwa do zastosowania dla dowolnego zbioru danych o dyskretnych wartościach atrybutów.
8. Można skorzystać z pakietów *pandas* i *scikit-learn* w celu załadowania zbioru danych oraz jego podziału na części.

2 Ćwiczenie

Celem ćwiczenia jest implementacja algorytmu naiwnego klasyfikatora Bayesa. Następnie należy wykorzystać stworzony algorytm do stworzenia i zbadania jakości klasyfikatorów dla zbioru danych SMS Spam Collection Dataset https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset. Klasą jest pole class.

2.1 Dla chętnych

1. Po wytrenowaniu modelu, wyekstrahuj i wyświetl listę 10 słów, które mają największe prawdopodobieństwo wystąpienia pod warunkiem, że wiadomość jest spamem oraz takich, które są najbardziej charakterystyczne dla zwykłych wiadomości (ham).

2. Zmodyfikuj proces przygotowania danych (tokenizację), tak aby model brał pod uwagę nie tylko pojedyncze słowa (unigramy), ale także pary słów występujących obok siebie (bigramy). Porównaj skuteczność modelu opartego na unigramach z modelem wykorzystującym unigramy + bigramy