

Implementacja naiwego klasyfikatora Bayesa

Ćwiczenie z przedmiotu WSI

Yan Korzun

4 lutego 2026

1 Cel ćwiczenia

Celem ćwiczenia była implementacja algorytmu naiwego klasyfikatora Bayesa oraz zbadanie jego jakości na rzeczywistym zbiorze danych tekstowych. Jako zbiór danych wykorzystano *SMS Spam Collection Dataset*, który zawiera wiadomości SMS oznaczone jako spam lub wiadomości poprawne (ham).

2 Opis zbioru danych

Zbiór danych pochodzi z serwisu Kaggle i zawiera 5574 wiadomości SMS. Każda próbka składa się z:

- etykiety klasy (`spam` lub `ham`),
- treści wiadomości tekstowej.

Dane zostały wczytane przy użyciu biblioteki `pandas`. Następnie podzielono je na zbiór treningowy (80%) oraz testowy (20%) z zachowaniem proporcji klas.

3 Przetwarzanie danych

Przed treningiem klasyfikatora przeprowadzono wstępne przetwarzanie tekstu:

- zamiana wszystkich liter na małe,
- usunięcie znaków specjalnych,
- podział na słowa,
- tworzenie słownika z liczbą występowania słów.

Każda wiadomość została zamieniona na zbiór słów, a następnie wykorzystana do obliczenia statystyk potrzebnych do działania algorytmu Bayesa.

4 Opis algorytmu

Zaimplementowany algorytm jest wariantem naiwnego klasyfikatora Bayesa typu Multinomial. Zakłada on warunkową niezależność cech (słów) w obrębie danej klasy.

Prawdopodobieństwo przynależności wiadomości do danej klasy obliczane jest według wzoru:

$$P(C | W) \propto P(C) \prod_{i=1}^n P(w_i | C)$$

W celu uniknięcia problemów numerycznych zastosowano logarytmy:

$$\log P(C | W) = \log P(C) + \sum_{i=1}^n \log P(w_i | C)$$

Dodatkowo użyto wygładzania Laplace'a:

$$P(w | C) = \frac{n_{w,C} + 1}{N_C + |V|}$$

gdzie:

- $n_{w,C}$ – liczba wystąpień słowa w w klasie C ,
- N_C – liczba wszystkich słów w klasie C ,
- $|V|$ – rozmiar słownika.

5 Eksperymenty

Model został wytrenowany na zbiorze treningowym, a następnie przetestowany na zbiorze testowym. Do oceny jakości wykorzystano następujące miary:

- Accuracy,
- Precision (dla klasy spam),
- Recall (dla klasy spam),
- F1-score (średnia geometryczna)

6 Wyniki

Uzyskane wyniki klasyfikacji przedstawiono poniżej:

- Accuracy: 0.97
- Precision (spam): 0.88
- Recall (spam): 0.94
- F1-score (spam): 0.91

Wysoka wartość miary Recall wskazuje, że klasyfikator skutecznie wykrywa większość wiadomości spam. Jednocześnie stosunkowo wysoka Precision oznacza, że liczba fałszywych alarmów pozostaje na akceptowalnym poziomie.

7 Wnioski

Zaimplementowany naiwny klasyfikator Bayesa osiągnął bardzo dobre wyniki mimo swojej prostoty. Algorytm skutecznie rozróżniał wiadomości spam od wiadomości poprawnych, co potwierdzają uzyskane miary jakości.

Przeprowadzone ćwiczenie pokazało, że naiwny Bayes jest efektywnym i szybkim algorytmem do klasyfikacji tekstu, szczególnie w zadaniach takich jak filtrowanie spamu, natomiast w jeżeli jest potrzebna wysoka precyzja w dużych projektach, np. w gmail.com, należy stosować bardziej zaawansowane algorytmy.