

Inteligentne systemy internetowe

Przykładowe tematy projektów:

1. Aplikacja internetowa – klastrowanie danych różnymi metodami

Aplikacja internetowa bazująca na języku python, z wykorzystaniem biblioteki scikit-learn, shiny, umożliwiającą wczytanie danych w formacie csv i ich klastrowanie i wizualizację z algorytmami kmeans, dbscan i inne. W procesie klastrowania użytkownik może ustawiać odpowiednie hiperparametry algorytmów. Ponadto aplikacja musi mieć możliwość przypisania do klastrow nowego/nowych nieznanych przypadków, zarówno poprzez generowany formularz jak i przez wczytanie z pliku csv.

2. Aplikacja internetowa – korelacja danych różnymi miarami

Aplikacja internetowa bazująca na języku python, z wykorzystaniem biblioteki scikit-learn, shiny, umożliwiającą wczytanie danych w formacie csv i określenie korelacji między wartościami atrybutów wykorzystując różne miary w zależności od typu danych (Pearson, Spearman, Kendall, Point-Biserial) i wizualizację w odpowiedniej formie tych zależności (np. heatmapa). W procesie obliczania korelacji użytkownik może ustawiać odpowiednie wymagane parametry metod. Ponadto aplikacja musi mieć możliwość przypisania selekcji podzbioru najistotniejszych zmiennych na podstawie korelacji i zapisania podzbioru do pliku csv.

3. Aplikacja internetowa – klasyfikacja danych modelem drzew decyzji

Aplikacja internetowa bazująca na języku python, z wykorzystaniem biblioteki scikit-learn, shiny, umożliwiającą wczytanie danych w formacie csv i budowę modeli drzew decyzji z różnymi hiperparametrami i wizualizację w odpowiedniej formie tych modeli. Ponadto aplikacja musi mieć możliwość przypisania do modeli nowego/nowych nieznanych przypadków, zarówno poprzez generowany formularz jak i przez wczytanie z pliku csv.

4. Aplikacja internetowa – klasyfikacja danych modelem sieci neuronowych

Aplikacja internetowa bazująca na języku python, z wykorzystaniem biblioteki scikit-learn, shiny, umożliwiającą wczytanie danych w formacie csv i budowę modeli sieci neuronowych z różnymi hiperparametrami i wizualizację w odpowiedniej formie tych modeli. Ponadto aplikacja musi mieć możliwość przypisania do modeli nowego/nowych nieznanych przypadków, zarówno poprzez generowany formularz jak i przez wczytanie z pliku csv.

5. Aplikacja internetowa - System predykcji zdrowia (np. ryzyko choroby).

Opis: Aplikacja online, która na podstawie danych zdrowotnych użytkownika (wiek, BMI, ciśnienie, poziom cukru itp.) przewiduje prawdopodobieństwo zachorowania na wybraną chorobę.

Założenia funkcjonalne:

- a. Formularz do wprowadzania danych.
- b. Wizualizacja wyniku predykcji (np. wykres ryzyka).
- c. Możliwość zapisu wyników w bazie danych.

- d. Moduł interpretacji wyników (np. SHAP).

Technologie:

- e. Frontend: React / Bootstrap
- f. Backend: Flask / FastAPI
- g. ML: scikit-learn (Logistic Regression, Random Forest), SHAP
- h. Baza: SQLite lub PostgreSQL

Zakres ML

- i. Uczenie modelu predykcyjnego z danych medycznych (np. zbiór diabetes.csv).
- j. Porównanie kilku modeli i wybór najlepszego.
- k. Rozszerzenie: Dashboard z interpretacją wyników (SHAP / LIME).

6. Aplikacja internetowa - Inteligentny system rekomendacji artykułów.

Zaprojektowanie aplikacji webowej, która analizuje zachowania użytkowników (kliknięcia, czas czytania, tematyka) i proponuje im spersonalizowane rekomendacje artykułów lub newsów.

Założenia funkcjonalne:

- l. Rejestracja i logowanie użytkowników.
- m. Przeglądanie listy artykułów.
- n. Gromadzenie danych o aktywności użytkownika.
- o. Moduł rekomendacji treści na podstawie historii przeglądania.
- p. Interfejs do oceny trafności rekomendacji.

Technologie:

- q. Frontend: React / Vue.js
- r. Backend: Python (Flask / FastAPI)
- s. Baza danych: MongoDB lub PostgreSQL
- t. ML: scikit-learn (Collaborative Filtering, KNN, PCA)

Zakres ML

- u. Uczenie modelu rekomendacji na danych użytkowników:
- v. macierz użytkownik–artykuł,
- w. algorytmy podobieństwa (cosine similarity, matrix factorization).

7. Aplikacja internetowa – wizualizacja danych z zastosowaniem różnych metod

Aplikacja internetowa bazująca na języku python, z wykorzystaniem biblioteki scikit-learn, shiny, umożliwiającą wczytanie danych w formacie csv i wizualizację w odpowiedniej formie (różne typy wykresów).

8. Aplikacja Inteligentny agregator informacji - stworzenie portalu, który pobiera artykuły z wielu źródeł i klasyfikuje je tematycznie.

Funkcjonalności:

- web scraping (Scrapy, RSS),
- klasyfikacja artykułów (np. sport, nauka, technologia),
- wykrywanie duplikatów treści,
- dashboard z trendami tematów.

Technologie: Python (scikit-learn, BeautifulSoup), Flask, Bootstrap.