# Dokumentacja techniczna projektu z przedmiotu: “Architektury rozwiązań i wdrożeń SI”

Opis projektu i jego funkcjonalności

W ramach tego projektu tworzony jest model AI, który może być wykorzystany w ocenie jakości wody. Aby nauczyć model wykorzystywane są dane pomiarowe pochodzące z z platformy Kaggle (https://www.kaggle.com/datasets/adityakadiwal/water-potability/). Kaggle to platforma zajmująca się nauką o danych oraz internetowa społeczność analityków danych i praktyków zajmujących się uczeniem maszynowym w ramach Google LLC. Kaggle umożliwia użytkownikom wyszukiwanie i publikowanie zbiorów danych, eksplorowanie i budowanie modeli w internetowym środowisku nauki o danych, współpracę z innymi badaczami danych i inżynierami uczenia maszynowego oraz udział w konkursach w celu rozwiązywania wyzwań związanych z nauką o danych. W ramach tej pracy wykorzystywany następujący zakres danych:

1) Wartość pH:
   pH jest ważnym parametrem w ocenie równowagi kwasowo-zasadowej wody. Jest także wskaźnikiem kwaśnego lub zasadowego stanu wody. WHO (World Health Organization) zaleca maksymalny dopuszczalny limit pH dla wody pitnej od 6,5 do 8,5. Obecne zakresy badań wynosiły 6,52–6,83 i mieszczą się w zakresie standardów WHO.

2) Twardość:
   Za twardość odpowiadają głównie sole wapnia i magnezu. Sole te rozpuszczają się w osadach geologicznych, przez które przepływa woda. Długość kontaktu wody z materiałem powodującym twardość pomaga określić stopień twardości wody surowej.

3) Substancje stałe (całkowita ilość rozpuszczonych substancji stałych – TDS):
   Woda ma zdolność rozpuszczania szerokiej gamy nieorganicznych i niektórych organicznych minerałów lub soli, takich jak potas, wapń, sód, wodorowęglany, chlorki, magnez, siarczany itp. Minerały te powodują niepożądany smak i odcień wody. Jest to ważny parametr dotyczący wykorzystania wody. Woda o wysokiej wartości TDS wskazuje, że jest to woda silnie zmineralizowana. Pożądany limit dla TDS wynosi 500 mg/l, a maksymalny limit to 1000 mg/l, przepisywany do celów pitnych.

4) Chloraminy:
   Chlor i chloramina to główne środki dezynfekcyjne stosowane w publicznych systemach wodociągowych. Chloraminy powstają najczęściej, gdy amoniak jest dodawany do chloru w celu uzdatniania wody pitnej. Poziomy chloru do 4 miligramów na litr (mg/l lub 4 części na milion (ppm)) są uważane za bezpieczne w wodzie pitnej.

5) Siarczany:
   Siarczany to substancje występujące naturalnie w minerałach, glebie i skałach. Występują w otaczającym powietrzu, wodach gruntowych, roślinach i żywności. Głównym komercyjnym źródłem siarczanów jest przemysł chemiczny. Stężenie siarczanów w wodzie morskiej wynosi około 2700 miligramów na litr (mg/l). W większości źródeł słodkiej wody waha się od 3 do 30 mg/l, chociaż w niektórych lokalizacjach geograficznych stwierdza się znacznie wyższe stężenia (1000 mg/l).

6) Przewodność:
   Czysta woda nie jest dobrym przewodnikiem prądu elektrycznego, a raczej dobrym izolatorem. Wzrost stężenia jonów poprawia przewodność elektryczną wody. Ogólnie rzecz biorąc, ilość rozpuszczonych substancji stałych w wodzie określa przewodność elektryczną. Przewodność elektryczna (EC) w rzeczywistości mierzy proces jonowy roztworu, który umożliwia mu przewodzenie prądu. Według standardów WHO wartość EC nie powinna przekraczać 400 µS/cm.

7) Węgiel organiczny:
   Całkowity węgiel organiczny (TOC) w wodach źródłowych pochodzi z rozkładającej się naturalnej materii organicznej (NOM), a także ze źródeł syntetycznych. TOC jest miarą całkowitej ilości węgla w związkach organicznych w czystej wodzie. Według US EPA < 2 mg/Litr jako TOC w wodzie uzdatnionej/pitnej i < 4 mg/Litr w wodzie pochodzącej z ujęcia przed uzdatnieniem.

8) Trihalometany (THM):
   THM to substancje chemiczne, które można znaleźć w wodzie uzdatnionej chlorem. Stężenie THM w wodzie pitnej różni się w zależności od poziomu substancji organicznych w wodzie, ilości chloru wymaganego do uzdatniania wody i temperatury uzdatnianej wody. Poziomy THM do 80 ppm są uważane za bezpieczne w wodzie pitnej.

9) Zmętnienie:
   Mętność wody zależy od ilości substancji stałych obecnych w stanie zawieszonym. Jest to miara miara nieprzezroczystości cieczy i służy do określenia jakości odprowadzanych ścieków pod względem zawartości materii koloidalnej. Średnia wartość zmętnienia uzyskana dla Wondo Genet Campus (0,98 NTU) jest niższa niż zalecana przez WHO wartość 5,00 NTU.

10) Zdatność do spożycia:
    Wskazuje, czy woda jest bezpieczna do spożycia przez ludzi, gdzie 1 oznacza zdatną do picia, a 0 oznacza niezdatną do picia.

Dane te są dzielone na dane treningowe i testowe w proporcji 80:20. Dane treningowe są używane do uczenia modeli, a dane testowe do weryfikowania poprawności działania modelu. Ponadto istnieje możliwość generowania danych syntetycznych przy pomocy biblioteki Pythona Synthetic Data Vault (SDV) w oparciu o dane pomiarowe, które w połączeniu z danymi pomiarowymi są wykorzystywane do retrenowania modeli, po podziale na dane treningowe i testowe, analogicznie jak w przypadku trenowania modeli.

Dane o modelach są zbierane przy pomocy platformy Weights & Biases (https://wandb.ai/site), a informacje o jakości danych syntetycznych są przetwarzane za pomocą biblioteki SDV.

Aplikacja łączy się z wandb API i dla każdego trenowanego modelu zapisuje na serwer dane o:

- precyzji (Precision): jest to stosunek liczby poprawnie zaklasyfikowanych pozytywnych przypadków do łącznej liczby przypadków zaklasyfikowanych jako pozytywne;
- dokładności (Accuracy): jest to stosunek liczby poprawnie zaklasyfikowanych przypadków (zarówno pozytywnych, jak i negatywnych) do łącznej liczby przypadków;
- odzysku (Recall): jest to stosunek liczby poprawnie zaklasyfikowanych pozytywnych przypadków do łącznej liczby faktycznie istniejących pozytywnych przypadków;
- wskaźniku F1 (F1 Score): jest to harmoniczna średnia precyzji i odzysku;
- macierzy pomyłek (Confusion Matrix): prezentuje liczbę poprawnych i błędnych klasyfikacji w formie tabeli

Ponadto można sprawdzić konfigurację oraz metryki systemu dla każdego wytrenowanego modelu.

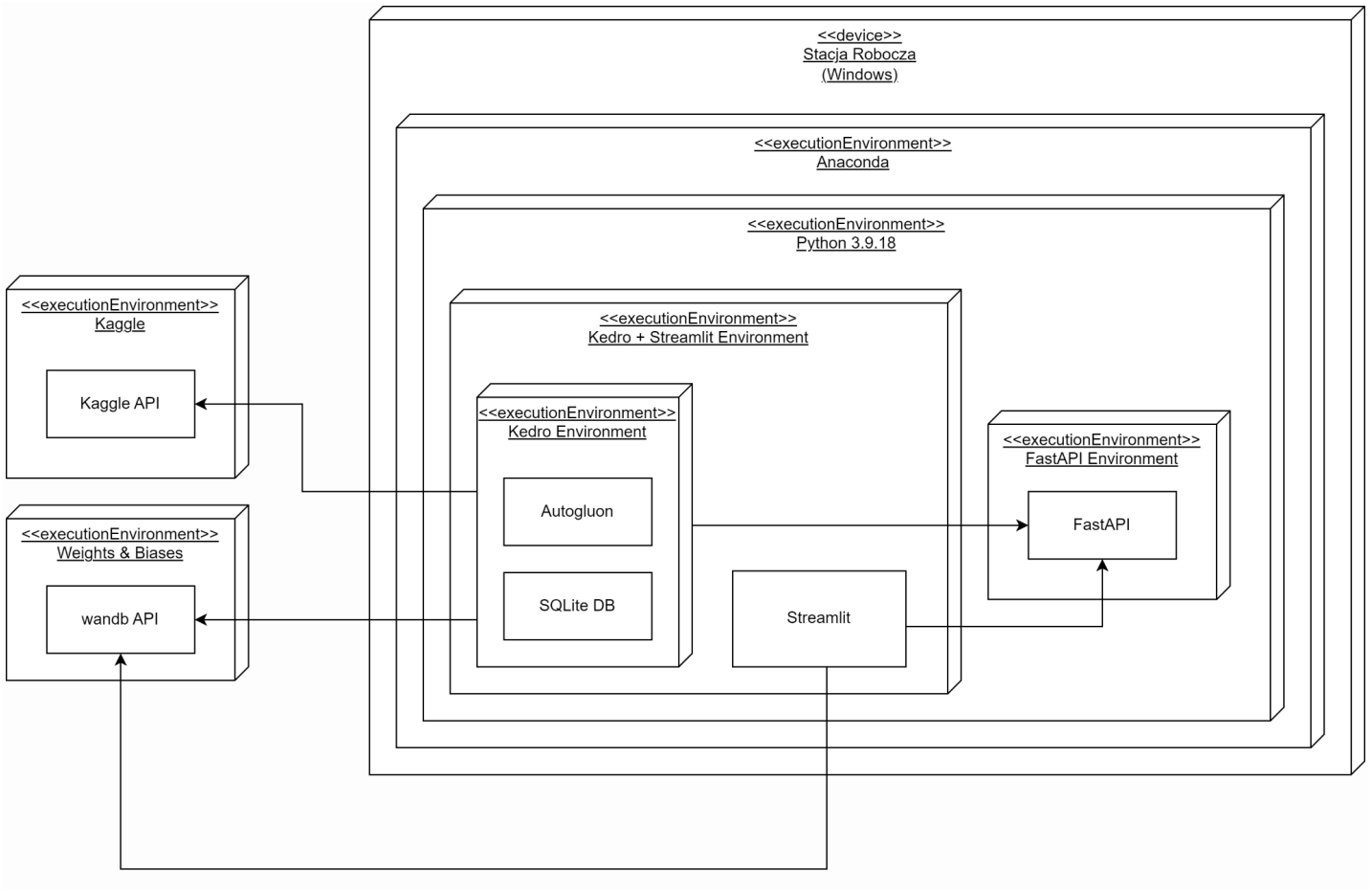
Jakość danych syntetycznych dla każdej kolumny jest sprawdzana poprzez:

- poprawność danych (data validity): czy klucze główne są unikalne i nie mają wartości null, czy wartość wygenerowanych danych nie przekracza maksymalnych i minimalnych wartości z danych pomiarowych i czy kategorie danych są ze sobą zgodne
- kształt kolumny (column shapes): statystyczne podobieństwo między danymi syntetycznymi i pomiarowymi
- trendy par kolumn (column pair trends): statystyczne podobieństwo między danymi syntetycznymi i pomiarowymi dla każdej pary kolumn, w tym korelacja między nimi
- wykresy częstości występowania wartości dla danych syntetycznych i pomiarowych dla każdej kolumny

Wykorzystane technologie

- Kedro do zarządzania danymi i pipeline'ami,
- sqlite do przechowywana danych,
- AutoGluon do automatycznego dostosowywania modeli,
- WandDB do śledzenia i wizualizacji trenowania modeli
- FastAPI do stworzenia API do interakcji z modelem,
- Streamlit do stworzenia prostego interfejsu użytkownika.

## Architektura



## Struktura plików

```
ASI_GROUP
├──.idea
│   └──inspectionProfiles
├──api
│   ├──asi-kedro
│   ├──config
│   ├──endpoint
│   │   └──__pycache__
│   ├──model
│   │   └──__pycache__
│   ├──services
│   │   └──__pycache__
│   ├──temp
│   ├──templates
│   └──__pycache__
├──asi-kedro
│   ├──AutogluonModels
│   ├──conf
│   │   ├──base
│   │   └──local
│   ├──dags
│   ├──data
│   │   ├──01_raw
│   │   ├──03_primary
│   │   ├──05_model_input
│   │   ├──06_models
│   │   └──07_model_outputs
│   ├──docs
│   │   └──source
│   ├──sqlite
│   ├──src
│   │   ├──asi_kedro
│   │   │   ├──pipelines
│   │   │   │   ├──data_engineering
│   │   │   │   │   └──__pycache__
│   │   │   │   ├──data_science
│   │   │   │   │   └──__pycache__
│   │   │   │   ├──model_evaluation
│   │   │   │   │   └──__pycache__
│   │   │   │   ├──model_retraining
│   │   │   │   │   └──__pycache__
│   │   │   │   ├──synthetic_data_creation
│   │   │   │   │   └──__pycache__
│   │   │   │   └──__pycache__
│   │   │   └──__pycache__
│   │   └──tests
│   │       └──pipelines
│   │           ├──data_engineering
│   │           ├──data_science
│   │           ├──model_evaluation
│   │           ├──model_retraining
│   │           └──synthetic_data_creation
│   └──wandb
├──streamlit
│   ├──tools
│   │   └──__pycache__
│   └──__pycache__
├──visulation
│   └──tools
├──system_level
└──__pycache__
```

Instalacja

Rekomendowane środowisko do tworzenia wirtualnych środowisk to Anaconda.

1. Utwórz wirtualne środowisko dla Python 3.9.18 (np. kedro_env), z którego będą uruchamiana back-end i streamlit
    - Przejdź do folderu system_level
    - Zainstaluj wymagane moduły za pomocą komendy:
      pip install -r requirements_kedro.txt
2. Utwórz wirtualne środowisko dla Python 3.9.18 (np. fastAPI_env), z któego będą wystawiane endpointy z FastAPI
    - Przejdź do folderu system_level
    - Zainstaluj wymagane moduły za pomocą komendy:
      pip install -r requirements_fastapi.txt

Tworzenie i retrenowanie modeli

W środowisku kendro_env, z folderu ASI_GROUP/asi-kedro można utworzyć modele za pomocą komendy:
kedro run

Retrenowanie modeli odbywa się za pomocą komendy:
kedro run --pipeline=retrain

Dostępne pipeline’y:

- "de": data_engineering_pipeline pobiera dane pomiarowe i dzieli je na dane treningowe i testowe
- "ds": data_science_pipeline trenuje i testuje modele przy pomocy biblioteki Autogluon
- “me": model_evaluation_pipeline wylicza metryki dla modelu i uploaduje je do wandb
- “mr": model_retraining_pipeline dzieli połączone dane testowe i syntetyczne na dane treningowe i testowe, a następnie trenuje i testuje modele przy pomocy biblioteki Autogluon
- “sdc": synthetic_data_creation_pipeline na podstawie danych pomiarowych tworzy dane syntetyczne przy pomocy biblioteki SDV
- “__default__": data_engineering_pipeline + data_science_pipeline + model_evaluation_pipeline,
- “retrain": model_retraining_pipeline + model_evaluation_pipeline

Uruchomienie FastAPI i Streamlita

W środowisku fastAPI_env, z folderu ASI_GROUP należy uruchomić FastAPI za pomocą komendy:
uvicorn api.main:app --reload

W środowisku kedro_env, z folderu ASI_GROUP należy uruchomić Streamlit za pomocą komendy:
streamlit run streamlit/stream_app.py