

Назва: Python SaaS для прогнозування повітряних тривог в регіоні

Мета: Розробка Python SaaS для прогнозування повітряних тривог в регіоні на основі різних факторів, таких як погода, новини та ситуація в інших регіонах.

Опис

Сервіс складається з чотирьох основних модулів (див. Рис. 1):

1. Модуль погоди (Weather Module):

- Weather API Integration Sub-Module отримує прогнози погоди на наступні 12 годин за допомогою наданого API (<https://www.visualcrossing.com/weather-api>) – див. Рис. 2
- Історичні прогнози погоди
- Data Preprocessing Sub-Module виконує попередню обробку даних про погоду для прогнозування

2. Модуль повітряних тривог (Air Alert Module):

- Air Alert API Integration Sub-Module дані про повітряні тривоги в інших регіонах за допомогою API (<https://api.ukrainealarm.com/>)
- Статистика повітряних тривог (<https://air-alarms.in.ua/>)
- Data Preprocessing Sub-Module виконує попередню обробку даних про повітряні тривоги для прогнозування

3. Модуль новин (News Module):

- ISW Data Dowloading Sub-Module збирає звіти про воєнні події з веб-сайту Інституту вивчення війни (<https://www.understandingwar.org/>) – див. Рис. 3
- Data Preprocessing Sub-Module виконує попередню обробку даних із завантажених звітів для прогнозування – див. Рис. 4

4. Модуль прогнозування (Air Alert Prediction Module):

- Використовуючи дані, отримані з модулів погоди, повітряних тривог і новин, прогнозує потенційні повітряні тривоги в регіоні
- Для прогнозування використовуються алгоритми машинного навчання, зокрема, обробка природної мови (Natural Language Processing).

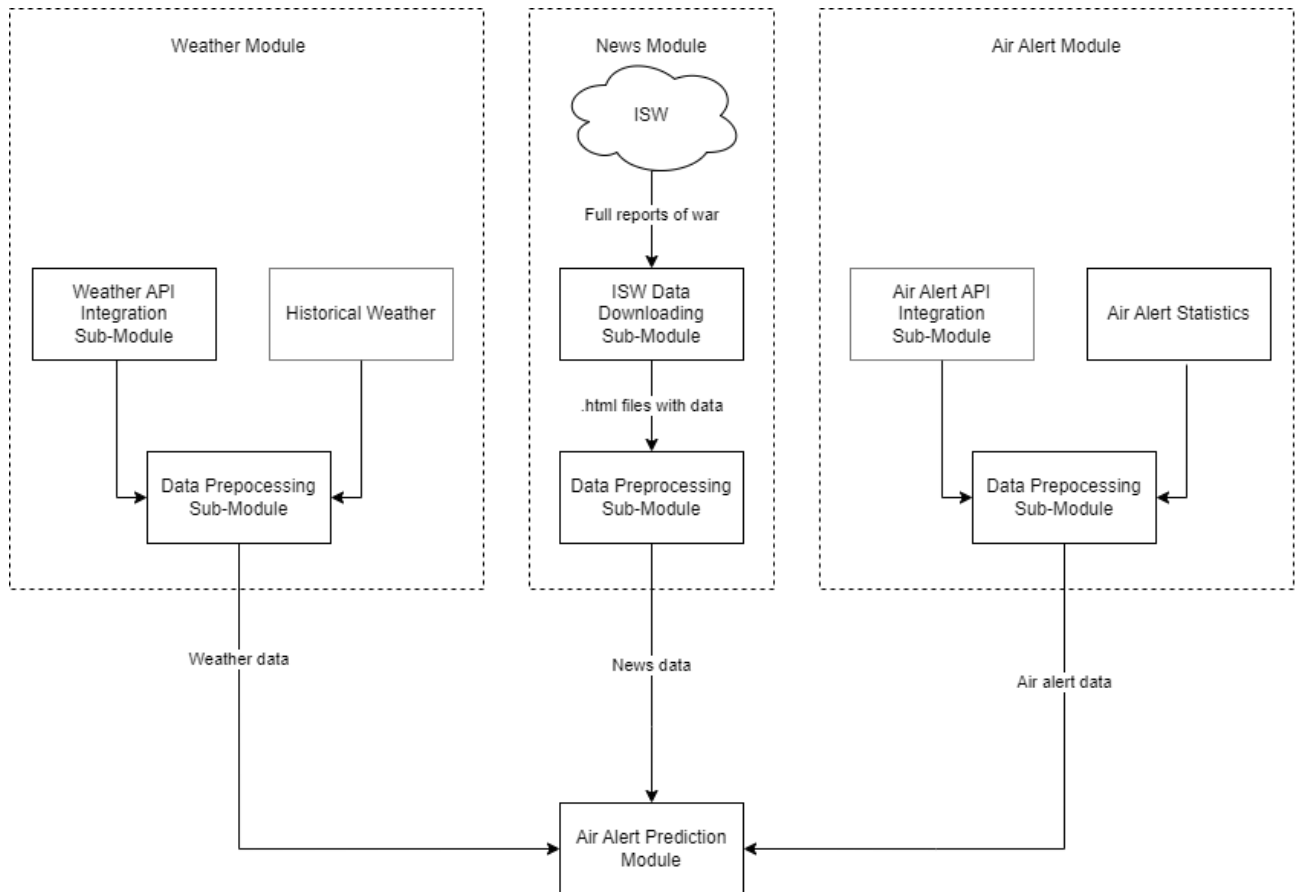


Рис. 1. Діаграма SaaS для прогнозування повітряних тривог в регіоні

Сервіс розроблений так, щоб його можна було масштабувати та легко підтримувати. Він буде розміщений на хмарній платформі та доступний через API. Користувачі зможуть переглядати і отримувати сповіщення про прогнозовані повітряні тривоги в обраному регіоні.

Сервіс буде реалізовано з використанням Python із фреймворком Flask. Для прогнозування використовуються алгоритми машинного навчання (обробка природної мови) з використанням Python-бібліотеки NLTK. Сервіс буде розгорнутий на хмарній платформі AWS, щоб забезпечити масштабованість і високу доступність.


```
1 with open("model/tfidf_transformer_v1.pkl", 'wb') as handle:
2     pickle.dump(tfidf_transformer, handle)
```

```
1 df_idf = pd.DataFrame(tfidf_transformer.idf_, index=cv.get_feature_names_out(), columns=["idf_weights"])
2 df_idf.sort_values(by=['idf_weights'])
```

	idf_weights
advanc	1.021429
luhansk	1.021429
area	1.025046
staff	1.028676
combat	1.028676
...	...
envoy	5.546835
t0803	5.546835
envis	5.546835
nexu	5.546835
nazism	5.546835

5745 rows × 1 columns

```
1 tf_idf_vector = tfidf_transformer.transform(word_count_vector)
```

```
1 tf_idf_vector
```

<282x5745 sparse matrix of type '<class 'numpy.float64'>'
with 154221 stored elements in Compressed Sparse Row format>

Рис. 4. Скріншот прикладу попередньої обробки даних для звітів про воєнні події

Team #6: Касьяненко Максим, Найда Андрій, Каліка Степан

GitHub: https://github.com/Ksenichiro/Naukma_api_project