

# SPRAWOZDANIE

## IUM etap II

### 1. Ocena modeli:

- model bazowy:

Jako model bazowy użyliśmy linearSVC. Dał on następujące wyniki:

```
Accuracy: 0.5608552631578947
Precision: 0.5459940652818991
Recall: 0.6174496644295302
F1 score: 0.5795275590551181
```

Jak widać te wyniki nie są zadowalające. Nauczony model osiąga wyniki niewiele lepsze niż losowo przypisane wartości.

- model właściwy:

W tym modelu użyliśmy las losowy RandomForestClassifier, który dał następujące wyniki:

```
Accuracy: 0.8585526315789473
Precision: 0.8173652694610778
Recall: 0.9161073825503355
F1 score: 0.8639240506329113
```

Po zestawieniu ich z naszymi założeniami:

Accuracy: 90%

Precision: 50%

Recall: 85%

F1 score: 63%

Widać, że wyniki są bardzo zadowalające. Jedynym wynikiem niższym od założonego to Accuracy, lecz pozostałe mocno przerosły nasze oczekiwania i z tego powodu zdecydowaliśmy się zaakceptować to rozwiązanie.

Aby upewnić się, że nasze dane są spreparowane w odpowiedni sposób sprawdziliśmy czy usunięcie którejkolwiek z kolumn poprawi wyniki lub pozostawi je bez zmian. Jednak okazało się, że usunięcie dowolnej kolumny pogarszało rezultat.

## 2. Aplikacja:

Aplikacja składa się z pięciu plików:



preprocess\_data.py



models.py



main\_test.py



home\_page.py



ai\_models.py

- Preprocess\_data.py – przygotowanie danych dla modelu.
- Models.py – wzornik dla danych, jeśli zostaną podane złe dane (np. dane ze złym typem danych) to aplikacja je odrzuci.
- Main\_test.py – obsługa klienta, ładowanie plików w formacie .json i wysyłanie danych oraz otrzymywanie predykcji od serwera.
- Home\_page.py – obsługa żądań od klienta, komunikacja z modułem obsługującym modele i modułem odpowiadającym za przygotowanie danych oraz przekazywanie odpowiedzi do klienta.
- Ai\_models.py – wczytywanie modeli i obsługa generowania predykcji.

Dostępne żądania w pliku home\_page.py:

### 1. Żądania typu get:

- "/get-users" - zwraca wszystkich zapisanych użytkowników
- "/get-sessions" - zwraca wszystkie zapisane sesje
- "/get-deliveries" - zwraca wszystkie zapisane dostawy
- "/get-products" - zwraca wszystkie zapisane produkty
- "/get-predictions" - zwraca wszystkie zapisane predykcje
- "/get-basic-prediction/{session\_id}" - zwraca predykcję bazowego modelu dla danego id sesji
- "/get-target-prediction/{session\_id}" - zwraca predykcję docelowego modelu dla danego id sesji
- "/get-both-models-prediction/{session\_id}" - zwraca predykcje obu modeli i dla danego id sesji
- "/save-all-data-and-predictions" - zapisuje wszystkie dane i predykcje

### 2. Żądania typu post:

- "/add-user" - dodaje użytkownika do bazy danych serwera
- "/add-delivery" - dodaje dostawę do bazy danych serwera
- "/add-product" - dodaje produkt do bazy danych serwera
- "/add-session-row" - dodaje jeden wiersz sesji do bazy danych serwera

### 3. Jak uruchomić serwer i aplikację:

Przygotowujemy środowisko trzema komendami uruchamianymi w folderze głównym:

- python -m venv venv
- ./venv/scripts/activate
- pip install -r requirements.txt

W folderze „app” wywołujemy komendę: `uvicorn home_page:app`

Czekamy na:

```
(venv) PS D:\nauka\sem5\ium\projekt\app> uvicorn home_page:app
INFO:      Started server process [18368]
INFO:      Waiting for application startup.
INFO:      Application startup complete.
INFO:      Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
```

uruchamiającą serwer. Aby zwrócić predykcje należy wpisać komendę:

`main_test.py`. Aplikacja automatycznie sprawdzi poprawność danych i wyświetli wynik z obydwóch modeli:

```
(venv) PS D:\nauka\sem5\ium\projekt\app> python main_test.py
Basic model prediction:
b'{"predictions for425":1}'
Target model prediction:
b'{"predictions for425":1}'
All predictions:
{'session_id': 129, 'target_prediction': 1, 'basic_prediction': 1}
{'session_id': 9645, 'target_prediction': 1, 'basic_prediction': 1}
{'session_id': 425, 'target_prediction': 1, 'basic_prediction': 1}
```

#### 4. Testy A/B:

Dla podanych danych:

```
app > test_data > test_users.json
1 {"user_id": 270, "name": "Elżbieta Broś", "city": "Gdynia", "street": "plac Makowa 49"}
2 {"user_id": 102, "name": "Klara Sałuda", "city": "Szczecin", "street": "ulica Dębowa 64/65"}
3 {"user_id": 107, "name": "Mariusz Sypek", "city": "Wrocław", "street": "ul. Tysiąclecia 24/85"}
```

Zawartość pliku test\_users.json

```
app > test_data > test_sessions.json
1 {"session_id": 129, "timestamp": "2021-05-21T12:56:12", "user_id": 102, "product_id": 1287, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
2 {"session_id": 129, "timestamp": "2021-05-21T12:56:54", "user_id": 102, "product_id": 1288, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
3 {"session_id": 129, "timestamp": "2021-05-21T13:01:43", "user_id": 102, "product_id": 1291, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
4 {"session_id": 129, "timestamp": "2021-05-21T13:04:42", "user_id": 102, "product_id": 1289, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
5 {"session_id": 129, "timestamp": "2021-05-21T13:08:50", "user_id": 102, "product_id": 1284, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
6 {"session_id": 129, "timestamp": "2021-05-21T13:12:34", "user_id": 102, "product_id": 1292, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
7 {"session_id": 129, "timestamp": "2021-05-21T13:15:43", "user_id": 102, "product_id": 1285, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
8 {"session_id": 129, "timestamp": "2021-05-21T13:20:02", "user_id": 102, "product_id": 1290, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
9 {"session_id": 129, "timestamp": "2021-05-21T13:22:35", "user_id": 102, "product_id": 1293, "event_type": "VIEW_PRODUCT", "offered_discount": 0, "purchase_id": null}
10 {"session_id": 129, "timestamp": "2021-05-21T13:23:18", "user_id": 102, "product_id": 1293, "event_type": "BUY_PRODUCT", "offered_discount": 0, "purchase_id": 20004}
11 {"session_id": 9645, "timestamp": "2021-06-16T04:19:14", "user_id": 270, "product_id": 1283, "event_type": "VIEW_PRODUCT", "offered_discount": 15, "purchase_id": null}
12 {"session_id": 9645, "timestamp": "2021-06-16T04:23:28", "user_id": 270, "product_id": 1283, "event_type": "BUY_PRODUCT", "offered_discount": 15, "purchase_id": 25745}
13 {"session_id": 425, "timestamp": "2021-11-16T22:06:56", "user_id": 107, "product_id": 1318, "event_type": "VIEW_PRODUCT", "offered_discount": 15, "purchase_id": null}
14 {"session_id": 425, "timestamp": "2021-11-16T22:08:37", "user_id": 107, "product_id": 1318, "event_type": "BUY_PRODUCT", "offered_discount": 15, "purchase_id": 20187}
```

Zawartość pliku test\_sessions.json

```
app > test_data > test_products.json
1 {"product_id": 1287, "product_name": "Philips SDV6226", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 129.0}
2 {"product_id": 1288, "product_name": "Philips SDV5100", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 119.0}
3 {"product_id": 1291, "product_name": "Philips SDV6224", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 168.79}
4 {"product_id": 1289, "product_name": "Philips SDV5225", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 129.0}
5 {"product_id": 1284, "product_name": "Opticum AX-900", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 49.0}
6 {"product_id": 1292, "product_name": "Philips SDV8622", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 189.0}
7 {"product_id": 1285, "product_name": "Opticum AX-800", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 82.99}
8 {"product_id": 1290, "product_name": "Philips SDV5120", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 169.0}
9 {"product_id": 1293, "product_name": "Hama 44292", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 44.99}
10 {"product_id": 1277, "product_name": "Apple iPad mini 64GB 4G", "category_path": "Komputery;Tablety i akcesoria;Tablety", "price": 2317.02}
11 {"product_id": 1276, "product_name": "Apple iPad mini 64GB", "category_path": "Komputery;Tablety i akcesoria;Tablety", "price": 1816.97}
12 {"product_id": 1318, "product_name": "Plantronics Savi W710", "category_path": "Sprzęt RTV;Audio;Słuchawki", "price": 553.0}
13 {"product_id": 1283, "product_name": "Okulary 3D PHILIPS PTA436/00", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Okulary 3D", "price": 99.99}
```

Zawartość pliku test\_products.json

```
app > test_data > test_deliveries.json
1 {"purchase_id": 20004, "purchase_timestamp": "2021-05-21T13:23:18", "delivery_timestamp": "2021-05-25T13:27:01", "delivery_company": 360}
2 {"purchase_id": 20187, "purchase_timestamp": "2021-11-16T22:08:37", "delivery_timestamp": "2021-11-19T22:12:07", "delivery_company": 620}
3 {"purchase_id": 25745, "purchase_timestamp": "2021-06-16T04:23:28", "delivery_timestamp": "2021-06-18T04:27:03", "delivery_company": 516}
```

Zawartość pliku test\_deliveries.json

Model zapisuje dane w takiej postaci i z nich korzysta (model w „products” zapisuje tylko kupione produkty, a nie historię ich przeglądania):

```
app > server_data > users.json
1 {"user_id": 102, "name": "Klara Sałuda", "city": "Szczecin", "street": "ulica Dębowa 64/65"}
2 {"user_id": 270, "name": "Elżbieta Broś", "city": "Gdynia", "street": "plac Makowa 49"}
3 {"user_id": 107, "name": "Mariusz Sypek", "city": "Wrocław", "street": "ul. Tysiąclecia 24/85"}
```

Zawartość pliku users.json

```
app > server_data > sessions.json
1 {"session_id": 129, "timestamp": "2021-05-21T12:56:12", "user_id": 102, "product_id": 1287, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
2 {"session_id": 129, "timestamp": "2021-05-21T12:56:54", "user_id": 102, "product_id": 1288, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
3 {"session_id": 129, "timestamp": "2021-05-21T13:01:43", "user_id": 102, "product_id": 1291, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
4 {"session_id": 129, "timestamp": "2021-05-21T13:04:42", "user_id": 102, "product_id": 1289, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
5 {"session_id": 129, "timestamp": "2021-05-21T13:08:50", "user_id": 102, "product_id": 1284, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
6 {"session_id": 129, "timestamp": "2021-05-21T13:12:34", "user_id": 102, "product_id": 1292, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
7 {"session_id": 129, "timestamp": "2021-05-21T13:15:43", "user_id": 102, "product_id": 1285, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
8 {"session_id": 129, "timestamp": "2021-05-21T13:20:02", "user_id": 102, "product_id": 1290, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
9 {"session_id": 129, "timestamp": "2021-05-21T13:22:35", "user_id": 102, "product_id": 1293, "event_type": "VIEW_PRODUCT", "offered_discount": 102, "purchase_id": null}
10 {"session_id": 129, "timestamp": "2021-05-21T13:23:18", "user_id": 102, "product_id": 1293, "event_type": "BUY_PRODUCT", "offered_discount": 102, "purchase_id": 20004}
11 {"session_id": 9645, "timestamp": "2021-06-16T04:19:14", "user_id": 270, "product_id": 1283, "event_type": "VIEW_PRODUCT", "offered_discount": 270, "purchase_id": null}
12 {"session_id": 9645, "timestamp": "2021-06-16T04:23:28", "user_id": 270, "product_id": 1283, "event_type": "BUY_PRODUCT", "offered_discount": 270, "purchase_id": 25745}
13 {"session_id": 425, "timestamp": "2021-11-16T22:06:56", "user_id": 107, "product_id": 1318, "event_type": "VIEW_PRODUCT", "offered_discount": 107, "purchase_id": null}
14 {"session_id": 425, "timestamp": "2021-11-16T22:08:37", "user_id": 107, "product_id": 1318, "event_type": "BUY_PRODUCT", "offered_discount": 107, "purchase_id": 20187}
```

Zawartość pliku sessions.json

```
app > server_data > products.json
1 {"product_id": 1293, "product_name": "Hama 44292", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Anteny RTV", "price": 44}
2 {"product_id": 1283, "product_name": "Okulary 3D PHILIPS PTA436/00", "category_path": "Sprzęt RTV;Video;Telewizory i akcesoria;Okulary 3D", "price": 99}
3 {"product_id": 1318, "product_name": "Plantronics Savi W710", "category_path": "Sprzęt RTV;Audio;Słuchawki", "price": 553}
4
```

Zawartość pliku products.json

```
app > server_data > deliveries.json
1 {"purchase_id": 20004, "purchase_timestamp": "2021-05-21T13:23:18", "delivery_timestamp": "2021-05-25T13:27:01", "delivery_company": 360}
2 {"purchase_id": 25745, "purchase_timestamp": "2021-06-16T04:23:28", "delivery_timestamp": "2021-06-18T04:27:03", "delivery_company": 516}
3 {"purchase_id": 20187, "purchase_timestamp": "2021-11-16T22:08:37", "delivery_timestamp": "2021-11-19T22:12:07", "delivery_company": 620}
4
```

Zawartość pliku deliveries.json