

Program szkolenia:

Od Software Engineer do ML Practitioner: Modelowanie i Dane

Informacje:

Nazwa:	Od Software Engineer do ML Practitioner: Modelowanie i Dane
Kod:	ml-prac
Kategoria:	ML
Odbiorcy:	developerzy, architekci
Czas trwania:	1 dzień
Forma:	30% wykłady / 70% warsztaty

Szkolenie prowadzi uczestników przez kompletny proces budowy rozwiązań predykcyjnych w oparciu o dane tabelaryczne. Począwszy od właściwego zdefiniowania problemu biznesowego, przez przygotowanie danych i inżynierię cech, aż po trening oraz rzetelną ewaluację modeli (ze szczególnym naciskiem na XGBoost i modele drzewiaste). Pracujemy w podejściu data-first: najpierw weryfikujemy, czy dane „niosą” informację o problemie, a dopiero potem dobieramy algorytm.

W trakcie szkolenia uczestnicy nauczą się:

- **Dobierać podejście do problemu:** kiedy wystarczy klasyczny ML (np. XGBoost/Sklearn), a kiedy sens mają gotowe usługi AI/LLM – wraz z kryteriami decyzji i konsekwencjami biznesowymi.
- **Rozpoznawać typ zadania i dobrać narzędzia:** klasyfikacja, regresja, ranking, clustering – jak dopasować technikę do celu i danych.
- **Przekładać cel biznesowy na definicję sukcesu modelu:** target, metryka techniczna, progi decyzyjne i koszt błędów.
- **Przeprowadzać EDA i weryfikować hipotezy:** statystyka opisowa, korelacje, rozkłady, wykrywanie sygnału vs. szumu w danych.
- **Przygotowywać dane do modelowania:** imputacja braków, obsługa outlierów oraz porządkowanie jakości i reprezentatywności danych.
- **Projektować i budować cechy (feature engineering):** transformacje dat, kategorii i tekstu do postaci użytecznej dla algorytmów.
- **Prowadzić trening w sposób kontrolowany:** podział train/val/test, unikanie overfittingu i praktyczne podejście do walidacji.
- **Interpretować i komunikować wyniki:** macierz pomyłek, ROC, Precision/Recall – co mówią o jakości i „rodzaju” błędów modelu oraz jak podejmować decyzje wdrożeniowe na ich podstawie.

Zalety szkolenia:

- Zmiana mindsetu z deterministycznego "if-else" na probabilistyczny
- Praca z surowymi danymi i wyciąganie z nich wartości.
- Zrozumienie, jak "uczy się" maszyna i jak sprawdzić, czy robi to dobrze.

Szczegółowy program:

1. Strategiczne modelowanie problemów (ML vs AI)

1.1. Kiedy ML, a kiedy AI? Kryteria wyboru między klasycznym Machine Learningiem (XGBoost, Sklearn), a gotowymi usługami AI/LLM.

1.2. Typologia zadań w biznesie: Klasyfikacja, Regresja, Ranking, Grupowanie (Clustering) – jak dopasować narzędzie do problemu.

1.3. Definiowanie sukcesu: Jak przełożyć cel biznesowy na target modelu i metrykę technologiczną.

1.4. "Data-First approach": Dlaczego model jest drugorzędny względem jakości i reprezentatywności danych

2. Eksploracja danych i Inżynieria cech (Feature Engineering)

2.1. EDA (Exploratory Data Analysis): Statystyka opisowa, szukanie korelacji i wizualizacja rozkładów danych.

2.2. Stawianie i weryfikacja hipotez: Czy te dane faktycznie niosą informację o problemie, który chcemy rozwiązać?

2.3. Inżynieria cech: Przekształcanie danych surowych (daty, kategorie, teksty) w format zrozumiały dla algorytmów.

2.4. Czyszczenie danych: Obsługa brakujących wartości (imputacja) i outlierów (wartości odstających).

3. Budowa, Trening i Ewaluacja Modelu

3.1. XGBoost i modele drzewiaste: Dlaczego nadal dominują w danych tabelarycznych i jak działają "pod maską".

3.2. Proces uczenia: Podział na zbiór treningowy, walidacyjny i testowy. Zapobieganie przeuczeniu (Overfitting).

3.3. Interpretacja wyników: Macierz pomyłek, krzywa ROC, Precision/Recall – co nam mówią o błędach modelu.