

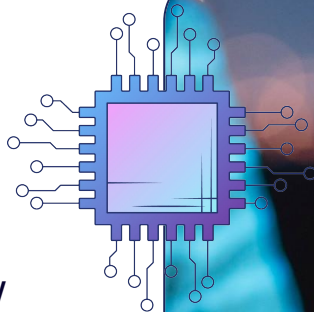


# **Porównanie algorytmów uczenia maszynowego dla predykcji wskaźnika rotacji klientów**

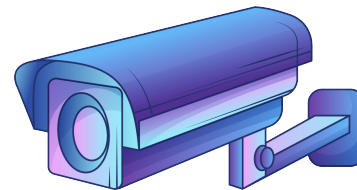
Mgr. Inż. Marzec Julia

## Cel pracy

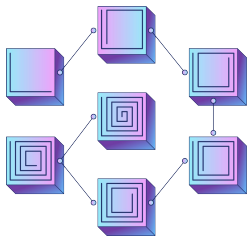
- Ocena skuteczności algorytmów uczenia maszynowego w predykcji rotacji klientów
- Porównanie różnych podejść
- Wyłonienie najlepszego modelu



# Zbiór danych i środowisko badawcze



- Telco Customer Churn z platformy Kaggle  
Każdy wiersz reprezentuje klienta, każda kolumna zawiera atrybuty klienta opisane w kolumnie Metadane.  
Surowe dane zawierają 7043 wiersze (klienci) i 21 kolumn (funkcje).
- Środowisko: Python, AutoGluon
- Google Colab



# Porównywane algorytmy i techniki

**01**    **LightGBMXT**

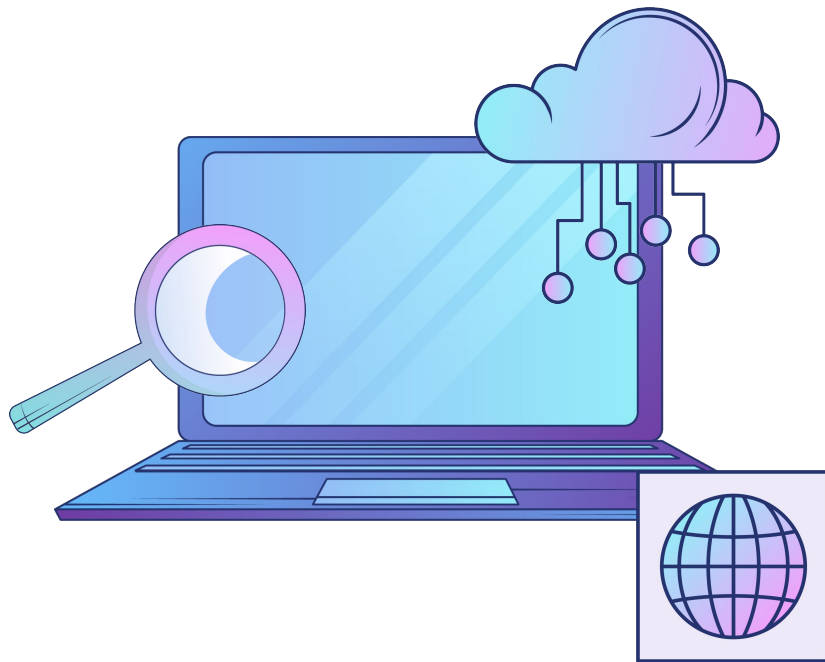
**02**    **XGBoost**

**03**    **CatBoost**

**04**    **CNN 1D**

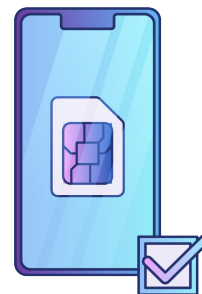
●    **SMOTE**

●    **scale\_pos\_weight**



# Wykorzystane metryki

- AUC-ROC
- Recall
- Accuracy
- Precision
- F1



## Wyniki model bazowy

Model	Precision	Recall	Accuracy	F1	AUROC
<b>LIGHTGBMXT</b>	0.62	0.53	0.79	0.57	0.83
<b>CatBoost</b>	0.67	0.51	0.80	0.58	0.84
<b>XGBoost</b>	0.62	0.53	0.79	0.57	0.83
<b>CNN</b>	0.62	0.60	0.79	0.60	0.83

Modele w scenariuszu bazowym wykazują wyższą precyzję, ale ich zdolność do wykrywania faktycznych odejść klientów jest ograniczona (niski recall). Oznacza to, że w takiej konfiguracji modele lepiej unikają fałszywych alarmów, ale mogą przeoczyć wielu klientów faktycznie zagrożonych odejściem.

## Wyniki model z ważeniem klas

Model	Precision	Recall	Accuracy	F1	AUROC
<b>LIGHTGBMXT</b>	0.48	0.78	0.71	0.60	0.83
<b>CatBoost</b>	0.51	0.80	0.74	0.63	0.84
<b>XGBoost</b>	0.52	0.77	0.75	0.62	0.84
<b>CNN</b>	0.50	0.79	0.74	0.61	0.84

Ważenie klasy churn pozwoliło uzyskać lepszą równowagę między recall a precision. Modele były w stanie wykrywać więcej przypadków odejść przy mniejszym spadku precyzji, co czyni tę metodę potencjalnie bardziej stabilną w zastosowaniach biznesowych.

# Wyniki model SMOTE

Model	Precision	Recall	Accuracy	F1	AUROC
<b>LIGHTGBMXT</b>	0.49	0.77	0.73	0.60	0.83
<b>CatBoost</b>	0.58	0.66	0.78	0.62	0.84
<b>XGBoost</b>	0.57	0.66	0.78	0.61	0.84
<b>CNN</b>	0.43	0.88	0.60	0.58	0.80

Zastosowanie SMOTE znacząco poprawiło skuteczność wykrywania klientów zagrożonych odejściem (wysoki recall), jednak kosztem większej liczby fałszywych alarmów (niższe precision). Taka strategia może być użyteczna tam, gdzie najważniejsze jest maksymalne wychwycenie potencjalnych odejść, nawet jeśli wymaga to dodatkowej weryfikacji.



# Wnioski



- CatBoost z ważeniem klas i SMOTE osiągnął najwyższą skuteczność
- CNN osiągnęło bardzo wysoki recall, aż 0.88, co oznacza, że prawie wszystkich klientów odchodzących udało się wychwycić. Jednak precision spadło do 0.43, co pokazuje dużą liczbę fałszywych alarmów, a accuracy ogólne wynosi tylko 60%
- Kluczowa metryka: recall, by maksymalnie wykrywać churn
- Modele bazowe miały wyższą precyzję, ale niższy recall
- W praktyce rekomenduję CatBoost z ważeniem klas (Precision: 0.51, Recall: 0.80)
- Predykcja churn wspiera działania retencyjne i ogranicza koszty
- Warto przetestować rozwiązanie na większym zbiorze

**Dziękuję za uwagę**