Fakultet tehničkih nauka
Novi Sad, Trg Dositeja Obradovića 6

Telco Customer Churn Classification

Mentor:

Danilo Kaćanski

Student:

Filip Goldberger

Novi sad, 29.09.2025.

Sadržaj

Uvod

Mašinsko učenje je oblast veštačke inteligencije koja se bavi razvojem algoritama i modela koji omogućavaju računarima da uče iz podataka i donose odluke bez eksplicitnog programiranja. Ova tehnologija je postala ključna jer omogućava automatizaciju, analizu velikih količina podataka i predviđanje budućih događaja.

Proces mašinskog učenja obuhvata prikupljanje i pripremu podataka, izbor odgovarajućeg algoritma, treniranje modela, evaluacija performanski i implementaciju rešenja.

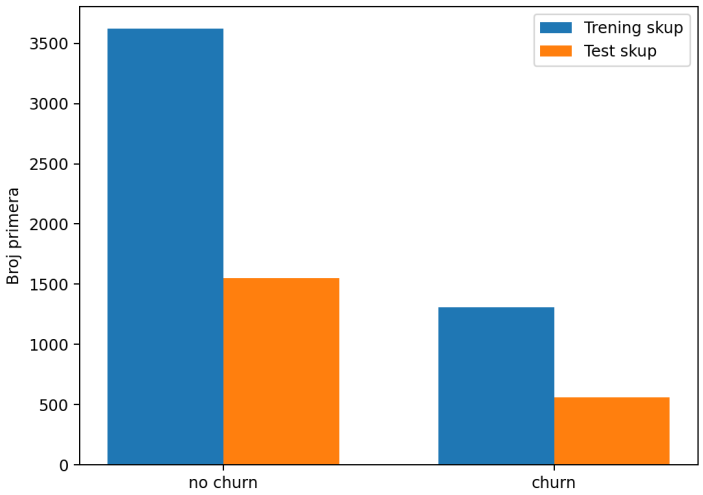
U ovom radu biće prikazan postupak izrade klasifikacionog modela, korišćeni algoritmi, evaluacija rezultata kroz ključne metrike (tačnost, preciznost, odziv, F1-skor) i diskusija o radu modela na konkretnom problemu.

Opis i priprema podataka

Za izradu klasifikacionog modela korišćen je skup podataka koji sadrži informacija o korisnicima telekomunikacionih usluga sa cilje predikcije da li će korisnik odustati od usluge (eng. *Churn*) ili ne (eng. *No Churn*). Svaki red u skupu podataka predstavlja jednog korisnika, dok kolone predstavljaju različite atribute korisnika: trajanje ugovora, tip ugovora, način plaćanja, ukupni troškovi, broj dodatnih usluga i druge relevantne informacije.

Podaci su prošli kroz sledeće korake pripreme:

- Učitavanje podataka iz „telco_data.csv“ datoteke
- Popunjavanje praznih polja u učitanom fajlu
- Čišćenje podataka: uklonjene su kolone sa premalo ili irelevantnim informacijama
- Enkodovanje podataka: kategorijske promenljive (npr. tip ugovora, način plaćanja) su pretvorene u numerički format korišćenjem ***LabelEncoder()***
- Skaliranje numeričkih vrednosti: numerički atributi poput ukupnih troskova su skalirani radi bolje konvergencije modela korišćenjem logaritamske funkcije i ***StandardScaler()***
- Podela na trening i test skup: podaci su podeljeni na trening (70%) i test (30%) skup, pri čemu je podela izvršena nasumično, ali stratifikovano kako bi u trening i test ušli sve moguće ciljne vrednosti.



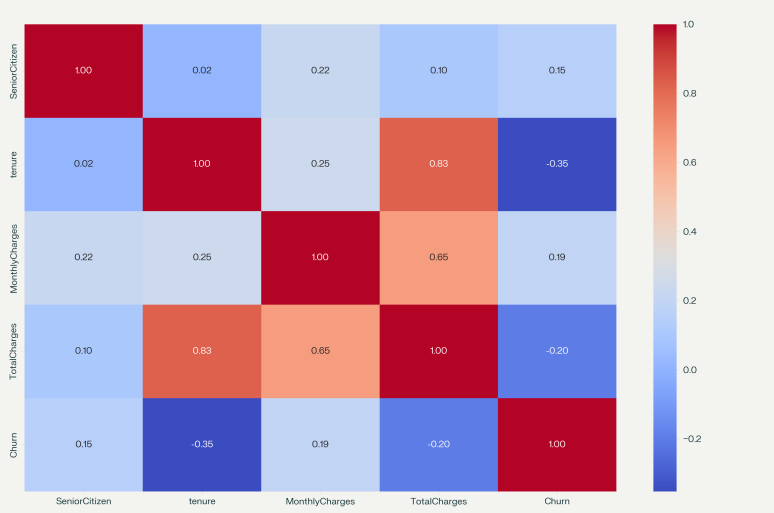
Slika 1: Raspodela klasa po skupovima

Eksplorativna analiza podataka

Nakon pripreme podataka i podele na trening i test skup urađena je eksplorativna analiza podataka (eng. *EDA*). Cilj ove analize je dublje razumevanje karakteristika skupa podataka i pronalaženje veza izmedju različitih atributa i ciljne promenljive ***Churn*** (odlazak korisnika).

Korelaciona matrica

Prvi korak u analizi bio je ispitivanje korelacione matrice između numeričkih atributa. Matrica nam vizuelno prikazuje jačinu i smer veza između atributa kao što su ***tenure*** (broj meseci korišćenja usluge), ***MonthlyCharges*** (mesečni troškovi), ***TotalCharges*** (ukupni troškovi).



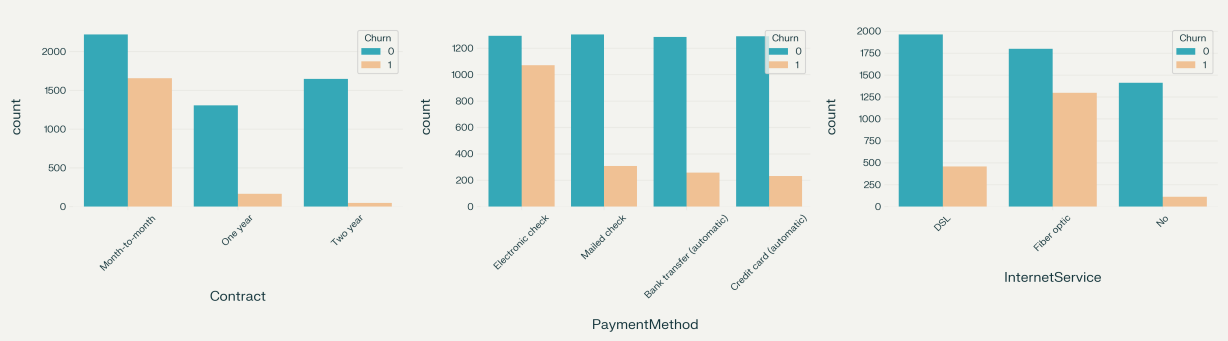
Slika 2: Korelaciona matrica

Iz matrice se mogu izvući sledeci zaključci:

- ***tenure*** i ***TotalCharges***: veoma jaka pozitivna korelacija (0.83), što je i logično jer korisnici koji duže koriste usluge imaju veće ukupne troškove
- ***tenure*** i ***Churn***: postoji negativna korelacija (-0.35), što govori da korisnici koji su duže verni kompaniji ređe otkazuju usluge
- ***MonthlyCharges*** i ***Churn***: blaga pozitivna korelacija (0.19), što govori da korisnici sa višim mesečnim računima imaju veću verovatnoću odlaska.

Analiza kategoričkih atributa

Analizirana je distribucija odliva prema tipu ugovora, načinu plaćanja i internet usluzi.



Slika 3: Kategorička distribucija po tipu ugovora, načinu plaćanja i internet usluzi

Iz grafikona se jasno vide sledeći trendovi:

- Tip ugovora (***Contract***): korisnici sa mesečnim ugovorom (***Month-to-month***) imaju veću stopu odliva u poređenju sa korisnicima koji imaju jednogodišnje (***One Yera***) ili dvogodišnje (***Two Year***) ugovore.
- Način plaćanja (***PaymentMethod***)Ć najveći procenat odliva je kod korisnika koji plaćaju putem elektronskog čeka (***Electronic check***)
- Internet usluga (***InternetService***): korisnici sa ***Fiber optic*** internetom imaju značajno veću stopu odliva u odnosu na korisnike sa ***DSL*** konekcijom

Odabir i treniranje modela

Nakon što su podaci pripremljeni i analizirani, sledeći korak je odabir, treniranje i optimizacija modela za predikciju odliva korisnika. S obzirom da je problem klasifikacione prirode odabrani su ansambl (***ensemble***) algoritmi koji su se dobro pokazali na tabelarnim podacima: ***Random Forest*** i ***Gradient Boosting***.

1. ***Random Forest***: kreira veći broj stabala odlučivanja na različitim podskupovima podataka i spaja njihove predikcije. Otporan je na preobučavanje (***overfitting***) i može da prepozna važnost atributa
2. ***Gradient Boosting***Č Sličan kao i Random Forest, kombinuje više stabala odlučivanja, ali ih trenira sekvencijalno gde svako novo stablo ispravlja greške prethodnog. Ima veoma visoke performanse

Pre finalne evaluacije potrebno je pronaći optimalne hiperparametre za svaki model i osigurati da su performanse modela pouzdane.

- Podešavanje hiperparametara: korišćen je ***RandomizedSearchCV()*** kako bi se isprobale različite kombinacije hiperparametara (npr. broj stabala, maksimalna dubina stabla, stopa učenja) i odabrala ona koja daje najbolje rezultate na validacionom skupu

```
DEFAULT_PARAMS_RF = {
    "n_estimators": [50, 75, 125, 150, 200, 300],
    "max_depth": [None, 5],
    "min_samples_split": [2, 5, 10, 20],
    "class_weight": ["balanced"]
}

DEFAULT_PARAMS_GB = {
    "n_estimators": [50, 75, 125, 150, 200, 300],
    "learning_rate": [0.01, 0.1, 0.2],
    "max_depth": [3, 5, 10]
}
```

Slika 4: Skup hiperparametara za trening

- Unakrsna validacija (eng. *Cross-Validation*): da bi se izbegla zavisnost performanski od jedne specifične podele na trening i test skup, primenjena je unakrsna validacija (***cv=5***). Na ovaj način model je treniran i testiran više puta na različitim delovima trening skupa, čime se dobija bolja procena njegovih performansi

```
def _train_model(X_train=None, y_train=None, model_class=None, model_name=""):

    if model_class == LogisticRegression:
        grid = RandomizedSearchCV(model_class(), param_distributions=DEFAULT_PARAMS_LR,
                                  n_iter=20, cv=5, scoring='f1', random_state=42, verbose=1)
    elif model_class == RandomForestClassifier:
        grid = RandomizedSearchCV(model_class(), param_distributions=DEFAULT_PARAMS_RF,
                                   n_iter=20, cv=5, scoring='f1', random_state=42, verbose=1)
    elif model_class == GradientBoostingClassifier:
        grid = RandomizedSearchCV(model_class(), param_distributions=DEFAULT_PARAMS_GB,
                                  n_iter=20, cv=5, scoring='f1', random_state=42, verbose=1)
```

Slika 5: Treniranje modela sa default parametrima

Evaluacija podataka

Za procenu performansi korišćene su sledeće metrike:

- Tačnost (***Accuracy***): procenat ukupno ispravno klasifikovanih instanci
- Preciznost (***Precision***): od svih korisnika koje je model predvideo da će otići, koliki procenat je zaista otišao. Visoka preciznost znači da model ne pravi mnogo lažno pozitivnih grešaka
- Odziv (***Recall***): od svih korisnika koji su zaista otišli, koliki procenat je model uspeo da identifikuje. Visok odziv znači da model uspešno „hvata“ slučajeve odliva
- F1-skor (***F1-score***): harmonijska sredina preciznosti i odziva. Koristan je kada je potrebno balansirati između preciznosti i odziva

```
Results for Forest Classifier:
Accuracy: 0.7283483199242783
Confusion Matrix:
 [[1079  473]
 [ 101  460]]
Classification Report:
              precision    recall  f1-score   support

           0       0.91      0.70      0.79      1552
           1       0.49      0.82      0.62       561

    accuracy                           0.73      2113
   macro avg       0.70      0.76      0.70      2113
weighted avg       0.80      0.73      0.74      2113

Results for Gradient Boosting:
Accuracy: 0.7927117841930904
Confusion Matrix:
 [[1401  151]
 [ 287  274]]
Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.90      0.86      1552
           1       0.64      0.49      0.56       561

    accuracy                           0.79      2113
   macro avg       0.74      0.70      0.71      2113
weighted avg       0.78      0.79      0.78      2113
```

Slika 6: Rezultat modela treniranim po odzivu

```
Results for Forest Classifier:
Accuracy: 0.7761476573592049
Confusion Matrix:
 [[1317  235]
 [ 238  323]]
Classification Report:
              precision    recall  f1-score   support

           0       0.85      0.85      0.85      1552
           1       0.58      0.58      0.58       561

    accuracy                           0.78      2113
   macro avg       0.71      0.71      0.71      2113
weighted avg       0.78      0.78      0.78      2113

Results for Gradient Boosting:
Accuracy: 0.795551348793185
Confusion Matrix:
 [[1405  147]
 [ 285  276]]
Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.91      0.87      1552
           1       0.65      0.49      0.56       561

    accuracy                           0.80      2113
   macro avg       0.74      0.70      0.71      2113
weighted avg       0.78      0.80      0.79      2113
```

Slika 7: Rezultat modela treniranim po F1-skoru

Matrica konfuzije

Pored numeričkih metrika se za vizuelnu analizu grešaka modela koristi i matrica konfuzije (***Confusion Matrix***). Ona prikazuje četiri ključne vrednosti:

- True Negatives (TN): korisnici koji nisu otišli i model je to tačno predvideo
- False Positives (FP): korisnici koji nisu otišli, ali je model predvideo da hoće
- False Negatives (FN): korisnici koji su otišli, ali je model predvideo da neće

Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

- True Positives (TP): korisnici koji su otišli i model je to tačno predvideo

Diskusija rezultata

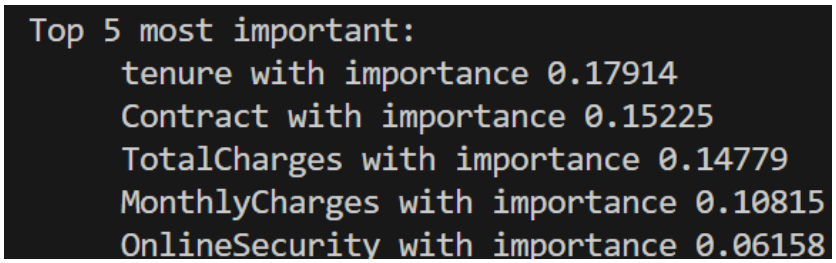
Oba modela pokazuju dobre rezultate sa ukupnom tačnošću preko 75%. ***Gradient Boosting*** model se pokazao neznatno boljim u svim ključnim metrikama:

- Tačnost je relativno visoka, ali s obzirom na nebalansiranost klasa (više korisnika ostaje nego što odlazi), ona sama po sebi nije dovoljna za procenu
- Preciznost od 0.55 pokazuje da su pola predikcija odliva bile tačne
- Odziv od 0.72 pokazuje da je model uspeo da identifikuje tri četvrtine stvarnih odlazaka. Ovo je ključna metrika za biznis, jer cilj jeste sprečiti odliv korisnika pa je važno što više njih detektovati

Na osnovu metrika može se zakljuciti da ***Gradient Boosting*** daje dovoljno dobre rezultate za ovaj problem.

Odabir najbitnijih atributa

Da bi se odredila važnost svakog atributa korišćena je ***feature_importances_*** funkcionalnost koja je ugrađena u modele kao što su *Random Forest* i *Gradient Boosting*. Ova metode meri koliko svaki atribut doprinosi odluci modela. Što je veća vrednost to je atribut bio značajniji za donošenje odluke.



Slika 8: Top pet najuticajnijih atributa modela

Zaključak

U okviru ovog projekta uspešno je treniran model mašinskog učenja za predikciju odliva korisnika u telekomunikacionoj kompaniji. Cilj je bio kreirati precizan klasifikacioni model i indentifikovati ključne faktore koji utiču na odluku korisnika da otkažu uslugu.

PS. Neko bi možda bio zadovoljan rezultatom modela mada iskreno mislim da bi moglo bolje ali nisam imao vremena da previše testiram