

Proje Önerisi

Proje İsmi: Web Uygulamaları için Kullanıcı Ayrılma (Churn) Tahmini ve Önleme Sistemi

Hazırlayan: Hasan Burak Songur

1. Giriş

Bu projenin amacı, bir web veya SaaS uygulamasını kullanan kişilerin geçmiş etkileşimlerini ve kullanıcı alışkanlıklarını analiz ederek, bir sonraki dönemde (örn. 30 gün içinde) hizmeti kullanmayı bırakma veya aboneliklerini iptal etme olasılıklarını tahmin etmektir.

Bu tahminler, şirketlerin ve ürün yöneticilerinin risk altındaki müşterileri proaktif olarak tespit etmeleri, onlara özel indirimler veya destek sunarak elde tutma (retention) stratejileri geliştirmeleri için pratik fayda sağlayabilir.

2. Yapay Zeka Çözümü

Proje, farklı modelleme yaklaşımlarını karşılaştırarak en iyi tahmin performansını (özellikle Precision ve Recall metrikleri açısından) elde etmeyi amaçlamaktadır.

Projede kullanılacak olan algoritmalar;

Basit Model: Kullanıcının statik özelliklerine (örn. üyelik süresi, abonelik paketi) veya basit kurallara dayalı bir Logistic Regression modeliyle başlanacak. Bu model aynı zamanda tahminlerin "nedenlerini" yorumlamak için de kullanılacaktır.

Geleneksel Makine Öğrenmesi: Kullanıcının son N gündeki özet istatistikleri (detaylı RFM ve davranışsal özellikler) gibi ek özellikleri kullanarak geleneksel ML modelleri (örneğin XGBoost) uygulanacaktır.

Derin Öğrenme (Deep Learning): Son olarak, ağaç tabanlı modellerden farklı bir yaklaşım olarak, türetilmiş zengin özellik setimizi kullanarak tam bağlantılı bir derin öğrenme modeli olan MLP (Multi-Layer Perceptron) geliştirilecektir. Bu, özellikler arasındaki karmaşık ve doğrusal olmayan ilişkileri yakalamak için farklı bir mimari denememizi sağlayacaktır.

Kullanılacak Teknolojiler: pandas, numpy, scikit-learn, XGBoost, TensorFlow/Keras (veya PyTorch), FastAPI, matplotlib, Kaggle (Veri Seti için), react

3. Projede Kullanılacak Olan Veri Seti

Veri seti neleri içerecek?

Projede kullanılacak veri, bir uygulamanın kullanıcı eylem loglarından (olay kayıtları) ve statik kullanıcı bilgilerinden oluşacak. Bunlar arasında user_id, event_time, event_type (örn: login, click, purchase), oturum süresi, abonelik tipi ve demografik bilgiler yer alacak.

Hedef Değişken (Churn Tanımı): Projedeki "churn" hedefi, Kaggle veri setinde bulunan net bir "abonelik iptal" (subscription_cancel veya benzeri) olayının gerçekleşip gerçekleşmemesi olarak tanımlanacaktır.

Nereden toplanacak?

Veriler, Kaggle üzerindeki hazır "Customer Churn" veya "Event Clickstream" veri setlerinden alınabilir (Örn: KKBBox Churn Prediction). Bu setler, gerçek dünyada bir backend event API'si ile toplanacak veriyi simüle edecektir.

4. Takım & Roller

Hasan Burak Songur:

Veri Toplama ve Temizleme: Kaggle'dan alınan ham log ve üye verilerinin birleştirilmesi, eksik verilerin doldurulması ve düzenlenmesi.

Özellik Mühendisliği: Tahmin için anlamlı yeni değişkenlerin (özelliklerin) oluşturulması.

XGBoost için: son_aktivite_gun_sayisi (Recency), son_30_gun_oturum_sayisi (Frequency), destek_bileti_acma_sayisi (Davranışsal) gibi detaylı özellikler.

MLP için: Verilerin derin öğrenme modeline uygun formata getirilmesi (örn: normalizasyon).

Model Geliştirme: Logistic Regression, XGBoost ve MLP modellerinin oluşturulması, eğitilmesi ve karşılaştırılması.

Backend: En iyi modeli (veya modelleri) FastAPI ile bir API servisi haline getirme (örn: POST /api/v1/predict_churn endpoint'i).

Frontend: React arayüzüyle (bir yönetici paneli olarak) kullanıcıların bir user_id girip o kullanıcının churn olasılığını ve Logistic Regression modelinden alınan yorumlanabilir "churn nedenlerini" görebilecekleri bir ekran geliştirme.

Analiz ve Raporlama: Sonuçların (Accuracy, Precision, Recall, F1-Score) değerlendirilmesi ve proje raporlarının hazırlanması.

5. Proje Aşamaları ve Zaman Çizelgesi

Hafta 1: Churn veri setlerinin araştırılması, Kaggle'dan en uygun (hem statik, hem sıralı, hem de net bir subscription_cancel hedefi içeren) setin seçilmesi ve indirilmesi.

Hafta 2: (Detaylandırıldı): Verilerin temizlenmesi, birleştirilmesi (data merging). XGBoost modeli için Özellik Mühendisliği:

Recency: son_aktivite_gun_sayisi, son_satin_alma_gun_sayisi

Frequency: toplam_oturum_sayisi_son_30_gun, haftalik_ortalama_oturum

Davranışsal: kullanılan_farkli_ozellik_sayisi, destek_bileti_acma_sayisi

Hafta 3: Logistic Regression ve XGBoost modellerinin kurulması, ilk sonuçların alınması.

Hafta 4: Model sonuçlarının (Logistic Regression, XGBoost) karşılaştırılması (Precision, Recall, F1-Score).

Yorumlanabilirlik Analizi: Logistic Regression modelinin katsayıları (model.coef_) incelenerek hangi özelliklerin churn'ü en çok tetiklediğinin (pozitif/negatif) listelenmesi.

İlk İlerleme Raporu'nun Hazırlanması: O ana kadar elde edilen bulgular, karşılaşılan zorluklar ve planın kalan kısmı için teyitleri içeren kısa bir raporun yazılması.

Hafta 5: MLP modeli için verinin uygun forma getirilmesi ve FastAPI tasarımına başlanması.

Hafta 6: MLP modelinin eğitilmesi. En iyi performanslı modelin (XGBoost veya MLP) ve yorumlama için Lojistik Regresyon modelinin (veya sadece katsayılarının) FastAPI ile entegre edilmesi.

Hafta 7: React yönetici paneli arayüzünün geliştirilmesine başlanması. Arayüzde: bir user_id giriş alanı, XGBoost/MLP'den gelen "Churn Skoru" ve Lojistik Regresyon'dan

gelen "Skoru Etkileyen İlk 3 Neden" alanı olacak.

Hafta 8: Frontend ve backend entegrasyonunun tamamlanması (React'in FastAPI'den churn skorunu ve yorumları çekmesi).

Hafta 9: Uçtan uca sistem testi, performans optimizasyonu ve arayüz iyileştirmeleri.

Hafta 10: Projenin belgelenmesi, sunum ve Final Raporu hazırlanması.