Zero2End ML Stage 2: “ETA Prediction” | Özge Usta

Öne Çıkan Maddeler

Bu konuşma, **ETA (Estimated Time of Arrival) Prediction** (Tahmini Varış Süresi) problemini ele almaktadır. Özellikle **Uber**, **Lyft**, **Didi**, **Martı** ve **Trendyol Go** gibi **Ride-sharing** ve teslimat firmalarının karşılaştığı bu sorun, kullanıcı deneyimini doğrudan etkileyen kritik bir unsurdur. Firmalar, **Google Maps API** gibi dış kaynakların getirdiği yüksek **maliyet** ve **latency** (gecikme) sorunları nedeniyle kendi modellerini geliştirmeye yönelmektedir. Amaç, milisaniyeler içinde doğru ve ekonomik ETA tahminleri sunmaktır.

Model geliştirme sürecinde, ham veri analiziyle başlanıp **Long Tail (Uzun Kuyruk) Dağılımı** ve **Outlier Detection** gibi temel veri kalitesi zorlukları giderilmiştir. **Feature Engineering** aşamasında, zaman, mesafe (**Haversine Distance**, **Bearing**, **Manhattan Distance**) gibi geleneksel özelliklerin yanı sıra, özellikle **Uber** tarafından geliştirilen **H3 (Hexagonal Hierarchical Spatial Index)** kütüphanesinin kullanımı öne çıkmıştır. H3 ile harita altıgenlere bölünerek, her bölgenin geçmiş yoğunlukları ve hızları gibi **uzamsal özellikler** (spatial features) türetilmiş, bu da **Tree-based** modellerin başarısını önemli ölçüde artırmıştır.

Model performansı değerlendirilirken sadece genel hata metrikleri yerine, veriyi farklı segmentlere (kısa, orta, uzun mesafeler) ayırarak her segmentteki hata oranlarının **MSE (Mean Squared Error)** ile ayrı ayrı incelenmesi, modelin **Long Tail** bölgesindeki yeteneğini anlamak için kritik bir yaklaşım olmuştur. Bu proje, pratik iş problemlerinde **veri temizliği**, **feature engineering** ve uygun **model seçimi**nin önemini vurgularken, büyük firmaların **Deep Learning** ve **Attention** mekanizmaları gibi ileri tekniklerle daha karmaşık ETA çözümleri geliştirdiğini de göstermiştir.

1. Problem Tanımı ve Motivasyon

1. **Sektör ve Kapsam: ETA (Estimated Time of Arrival)** tahmini, **Ride-sharing** (Uber, Lyft, Martı) ve teslimat (Trendyol Go, Getir) sektörlerinin temel problemidir.
2. **Maliyet Baskısı:** Firmaların kendi modellerini geliştirmesinin ana nedeni, **Google Maps API** maliyetlerinin yüksek olmasıdır.
3. **Hız (Latency) İhtiyacı:** Sürücü ve yolcu eşleşmesinin **milisaniyeler** içinde yapılması gerektiği için dış servislere bağımlılık azaltılmalıdır.
4. **Büyük Oyuncular:** Uber, Didi gibi firmalar bu problem için çok gelişmiş **Deep Learning** mimarileri ve akademik makaleler üretmektedir.

2. Veri Zorlukları ve Temizliği

1. **Long Tail Dağılımı:** ETA verisi normal dağılmaz; veri sola dayalıdır ancak sağa doğru uzayan bir **Long Tail (Uzun Kuyruk)** vardır, bu da uç değerlerin tahminini zorlaştırır.
2. **Aykırı Değer (Outlier) Analizi:** 1 saniyelik veya 900 saatlik sürüşler gibi mantıksız verilerin **temizlenmesi** gerekir.
3. **Label (Etiket) Gürültüsü:** Sürücülerin uygulamayı geç kapatması veya **fraud** (sahtecilik) girişimleri veriyi kirletebilir.
4. **Rota Belirsizliği:** Tahmin edilen rota ile sürücünün izlediği **gerçek rota** arasındaki farklar (trafik, kazalar) sapmalara neden olur.

3. Feature Engineering (Öznitelik Mühendisliği)

1. **Temel Değişkenler:** Zaman (saat, gün, hafta sonu) ve ham lokasyon verileri başlangıç noktasıdır.
2. **Mesafe ve Yön:** İki nokta arasındaki **Haversine Distance** (kuş uçuşu mesafe) ve **Bearing** (yön açısı) hesaplamaları modele yön verir.
3. **H3 Kütüphanesi:** Uber'in geliştirdiği **H3 (Hexagonal Hierarchical Spatial Index)** ile haritayı altıgenlere bölmek ve **bölgesel yoğunluk/hız** özelliklerini türetmek model başarısını en çok artıran yöntemdir.
4. **Alternatif Yöntemler:** Koordinatları dönüştürmek için **PCA** veya kümeleme için **K-Means** denenebilir ancak H3 genelde daha iyi sonuç verir.

4. Modelleme ve Değerlendirme

1. **Model Seçimi:** Çok büyük verilerde **Deep Learning** kullanılsa da, bu projede hızlı ve etkili olduğu için **Tree-based** (Ağaç tabanlı) modeller tercih edilmiştir.
2. **Hata Fonksiyonu:** Büyük sapmaları (Long Tail kısmını) daha çok cezalandırmak için **MSE (Mean Squared Error)** kullanılması önerilir.
3. **Segmentasyon Analizi:** Model başarısını sadece ortalamaya göre değil, **Low, Middle, High** süre segmentlerine ayırarak incelemek, modelin nerede hata yaptığını gösterir.

5. Canlı Sistem (Production) ve Operasyonel Bakış

1. **Dış Veri Kaynakları: Hava durumu** verisi Kaggle yarışmalarında skoru artırsa da, **production** ortamında API gecikmeleri ve maliyet nedeniyle her zaman tercih edilmeyebilir.
2. **Coğrafi Dinamikler:** Her şehrin (Örn: **İstanbul Boğazı** vs. Konya düzlüğü) coğrafi yapısı farklıdır, modelin bu dinamikleri öğrenmesi gerekir.
3. **İlgili Alanlar:** Bu problem, Endüstri Mühendisliğindeki **Operations Research** ve **Traveling Salesman Problem (Gezgin Satıcı)** konularıyla yakından ilişkilidir.