## **Yaptıklarımın Raporu**

Bu projede göçmen şikayetleri veri seti üzerinde doğal dil işleme (NLP) tekniklerini kullanarak analiz yaptım. Hem veriyi temizledim, hem de şikayetler arasındaki benzerliği ölçerek bazı görsel sonuçlar elde ettim. Tüm işlemleri adım adım aşağıda açıkladım:

### **1. Gerekli Kütüphaneleri Kurup İçe Aktardım**

Projeye başlamadan önce pandas, numpy, matplotlib, seaborn, nltk, gensim, sklearn gibi veri analizi ve NLP için gerekli kütüphaneleri kurdum ve içe aktardım. Ayrıca uyarı mesajlarını gizledim ki çıktı ekranı daha düzenli görünsün.

### **2. NLTK Verilerini İndirdim**

NLTK kütüphanesinden punkt, stopwords, wordnet gibi verileri indirdim. Bu sayede kelimeleri ayırmak, anlamsız olanları atmak ve köklerine indirmek mümkün hale geldi.

### **3. Veri Setini Okudum ve İnceledim**

CSV dosyasını pandas ile okudum. İçindeki boş satırları sildim. Veri setine genel olarak baktım, sütunların yapısını ve örnek satırları inceledim.

### **4. Metin Temizleme İşlemleri Yaptım**

Şikayet metinleri içinde geçen özel karakterler, noktalama işaretleri ve sayıları temizledim. Ardından küçük harfe çevirme, stopword'leri çıkarma ve lemmatizasyon/stemming işlemleri yaptım. Bu sayede metinler daha sade ve anlamlı hale geldi.

### **5. Tokenizasyon ve Kelime Frekansı Hesapladım**

Tüm metinleri tek parça olarak birleştirip kelime kelime ayırdım (tokenizasyon yaptım). Daha sonra hangi kelimeden kaç tane geçtiğini sayarak frekanslarını buldum. En sık kullanılan kelimeleri listeledim.

### **6. Zipf Yasası Grafiği Oluşturdum**

Kelimelerin frekanslarını log-log grafikte çizdim. Bu grafik Zipf Yasası’nı doğrulamak için kullanılır. En çok geçen kelimelerle sıralama arasındaki ilişkiyi burada gözlemledim.

### **7. TF-IDF Vektörleme Yaptım**

TF-IDF (Term Frequency - Inverse Document Frequency) yöntemiyle şikayet metinlerini sayısal vektörlere dönüştürdüm. Böylece her kelimenin hangi metinde ne kadar önemli olduğunu ölçtüm.

### **8. Word2Vec Modeli Eğittim**

Gensim kütüphanesi ile Word2Vec modelini kurdum ve eğittim. Bu sayede kelimeler anlamlarına göre birbirine yakın olan vektörlere dönüştü. Yani artık "ev" kelimesi ile "konut" kelimesi arasındaki benzerliği sayısal olarak gösterebiliyoruz.

### **9. Şikayetler Arası Benzerlik Hesapladım**

Cosine similarity yöntemi ile Word2Vec vektörleri üzerinden şikayetler arasındaki benzerliği hesapladım. Hangi şikayetlerin birbirine benzediğini bu şekilde ölçtüm.

### **10. Görselleştirme Yaptım**

Matplotlib ve Seaborn kütüphanelerini kullanarak frekans grafikleri, dağılım grafikleri ve benzerlik matrisleri oluşturdum. Bu görseller veriyi daha anlaşılır hale getirdi.

### **Sonuç**

Bu projede doğal dil işleme (NLP) tekniklerini kullanarak göçmen şikayetlerini detaylı bir şekilde analiz ettim. Kelime vektörleri oluşturup, benzerlikleri ölçerek veriden anlamlı bilgiler çıkardım. İleride bu veri sınıflandırma ya da kümeleme için kullanılabilir.