**DERYA KESGİN - 02210224010**

**Osmanlıca Optik Karakter Tanıma (OCR) İçin Derin Sinir Ağları Kullanımı**

**1. Çalışmanın Amacı ve Önemi**

Osmanlıca, Osmanlı İmparatorluğu’nda **13. ve 20. yüzyıllar** arasında kullanılan Arap alfabesi tabanlı bir dildir. Günümüzde Osmanlı arşivlerinde milyonlarca Osmanlıca belge bulunmaktadır, ancak bunların büyük çoğunluğu henüz dijital metne dönüştürülmemiştir.

Bu çalışmanın temel amacı:  
-Osmanlıca matbu **nesih hattıyla yazılmış** belgeleri **optik karakter tanıma (OCR)** kullanarak dijital metne çevirmek  
- **Derin öğrenme tabanlı** bir model geliştirmek  
- Mevcut OCR araçları (Tesseract, Google Docs, Abby FineReader, Miletos) ile karşılaştırarak performans farklarını incelemek

**2. Kullanılan Derin Öğrenme Mimarisi**

Bu OCR sistemi, **Convolutional Neural Network (CNN) + Recurrent Neural Network (RNN) tabanlı bir CRNN (Convolutional Recurrent Neural Network) modeli** kullanılarak geliştirilmiştir.

Modelin temel bileşenleri:

**A. CNN Katmanı (Convolutional Neural Network)**

- Görüntüdeki karakterlerin **kenarlarını, gövdelerini ve genel yapılarını tanır**  
- Görüntüden **özellik çıkarımı (feature extraction)** yapar  
- Katmanlar giderek daha **soyut temsiller** oluşturur  
- **İlk katman:** Piksel seviyesinde özellikler çıkarır  
- **İkinci katman:** Harf gövdeleri belirlenir  
- **Üçüncü katman:** Harflerin kendisi tespit edilir  
- **Dördüncü katman:** Tüm satır düzeyinde analiz yapılır

**B. RNN (LSTM) Katmanı (Recurrent Neural Network)**

- Karakterlerin **bir sıra içinde nasıl yer aldığını** öğrenir  
- **İki yönlü (bidirectional) LSTM** kullanılarak, **hem önceki hem de sonraki harfleri dikkate alır**  
- Osmanlıca gibi **bağlamsal yazım kurallarına sahip dillerde yüksek doğruluk sağlar**  
- Zaman serisi içindeki **ilişkileri ve örüntüleri** yakalar

**C. CTC (Connectionist Temporal Classification) Katmanı**

- Modelin **sıralı veriyi anlamasını ve etiketlemesini sağlar**  
- Karakterlerin **doğru sıralamasını tahmin eder**  
 Bir satırın **doğru okunmasını sağlayan hata düzeltme mekanizması** içerir

-- **Sonuç:**  
Bu yapı, **CNN’in görsel analiz gücü ile LSTM’in sıralı veri işleme gücünü birleştirerek** Osmanlıca OCR için en iyi sonucu vermektedir.

**3. Eğitim ve Test Süreci**

**Üç farklı veri seti** oluşturulmuş ve model bu veri setleri ile eğitilmiştir:

**A. Eğitim Veri Setleri**

**Orijinal Veri Seti:** 1000 sayfa gerçek Osmanlıca belge  
**Sentetik Veri Seti:** **23.000 sayfa** yapay olarak üretilmiş Osmanlıca metin görüntüleri  
**Hibrit Veri Seti:** Orijinal ve sentetik verilerin birleşimi

**B. Test Veri Seti**

Test seti, **8 farklı Osmanlıca eserden alınmış 21 sayfalık belge** içerir.

**Ölçüm Metotları:**  
- **Ham Metin (Raw)**: OCR çıktısının doğrudan doğruluğu  
- **Normalize Edilmiş Metin (Normalized)**: OCR sonrası temel hata düzeltmeleri uygulanmış  
- **Bitişik Metin (Joined)**: Harf birleşim kuralları uygulanarak metnin doğruluğu test edilmiş

**4. Model Performans Karşılaştırmaları**

Model, **Google Docs, Tesseract (Arapça & Farsça), Abby FineReader, Miletos** gibi mevcut OCR araçlarıyla karşılaştırılmıştır.

| **Model** | **Ham (%)** | **Normalize (%)** | **Bitişik (%)** |
| --- | --- | --- | --- |
| **Hibrit Model (En İyi Sonuç)** | **88.86** | **96.12** | **97.37** |
| Orijinal Model | 87.73 | 94.87 | 96.16 |
| Sentetik Model | 73.16 | 77.64 | 78.10 |
| Google Docs OCR | 83.86 | 92.02 | 91.43 |
| Abby FineReader | 71.98 | 80.19 | 81.05 |
| Tesseract (Arapça) | 76.92 | 82.37 | 81.27 |
| Tesseract (Farsça) | 75.30 | 83.85 | 83.48 |
| Miletos OCR | 75.76 | 86.46 | 86.88 |

**Sonuç:** Hibrit model, **karakter, bağlı karakter ve kelime tanıma doğruluğunda en yüksek başarıyı göstermiştir.**

**5. Osmanlıca OCR’da Karşılaşılan Zorluklar**

- **Bağlantılı Harfler:** Osmanlıcada harfler **bitişik yazıldığından**, karakter bölütleme zorlaşmaktadır.  
- **Çoklu Harf Formları:** Harflerin **kelimenin başında, ortasında ve sonunda farklı yazım şekilleri vardır.**  
- **Benzer Harfler:** **ب (b) ve ت (t)** gibi harfler birbirine çok benzediği için OCR hata oranı artmaktadır.  
- **Noktalar ve Hareke İşaretleri:** Noktalar (•), harekeler ve diğer küçük semboller OCR sürecini zorlaştırmaktadır.  
- **Kelime Bölümleme Hataları:** Osmanlıcada kelimeler **bitişik veya ayrık yazılabildiğinden**, OCR'nin doğru bölümlendirme yapması gerekir.

**Bu zorlukları aşmak için:**  
- **Karakter, bağlı karakter ve kelime düzeyinde dil modelleri geliştirilmiştir.**  
- **Özel normalizasyon algoritmaları** ile OCR hataları azaltılmıştır.

**6. Sonuç ve Özet**

- **Geliştirilen model, Osmanlıca metinleri yüksek doğruluk oranıyla tanıyabilen derin öğrenme tabanlı bir OCR sistemidir.**  
- **CNN ve LSTM mimarisi birleştirilerek**, Osmanlıca gibi karmaşık bağlamsal özellikleri olan bir dil için en iyi OCR performansı sağlanmıştır.  
- **Model, mevcut OCR araçlarından daha başarılıdır ve Osmanlıca belgelerin dijitalleşmesi için önemli bir katkı sunmaktadır.**  
- **Bu model, Osmanlica.com adresinde OCR aracı olarak kullanıma sunulmuştur.**