

El Yazılarının Evrişimli Sinir Ağları Kullanılarak Cinsiyet ve Yazımda Kullanılan El Bakımından Sınıflandırılması

Mustafa Serhat USLU¹, Burak TOKMAK²

1- 161180061
uslu.mserrhat2@gmail.com

2- 161180057
buraktokmak07@gmail.com

ÖZETÇE

Cinsiyet ve yazımda kullanılan elin sınıflandırılması gibi demografik el yazısı temelli sınıflandırma problemleri adli biyometri gibi disiplinlerdeki uygulamalarda önem taşır. Bu çalışma üç ayrı yapay sinir ağıнын; cinsiyet, yazımda kullanılan el ve bu problemlerin birleşimi için uyumunu inceler. Bu çalışma için kullanılan veri seti beyaz tahta üzerine yazılmış el yazılarının dijital ortama geçirilmesiyle elde edilmiş “IAM On-Line Handwriting Database” den elde edilmiştir. Çözüm sunulan problem doğal bir sınıflandırma zorluğu taşır, çözüme bağlı niteliklerin ayıklanması oldukça zordur. Çözümümüzde elle üretilmiş niteliklerle kıyaslandığında daha yüksek başarı oranlarıyla kendini ispatlamış olan evrişimli sinir ağları kullanılır. İleri sürülen çözümlerde özgün ağ mimari yapıları kullanılmıştır. Elde edilenler karşılaştırıldığında, iki demografik problem için kurulan ayrı ağ yapıları problemlerin birleşimi için kullanılan ağ yapısından daha yüksek başarı göstermiştir.

Anahtar Kelimeler: Evrişimli Yapay Sinir Ağları, Demografik Sınıflandırma, Biyometri

Using Convolutional Neural Networks to Predict Gender and Handedness from On-Line Handwriting

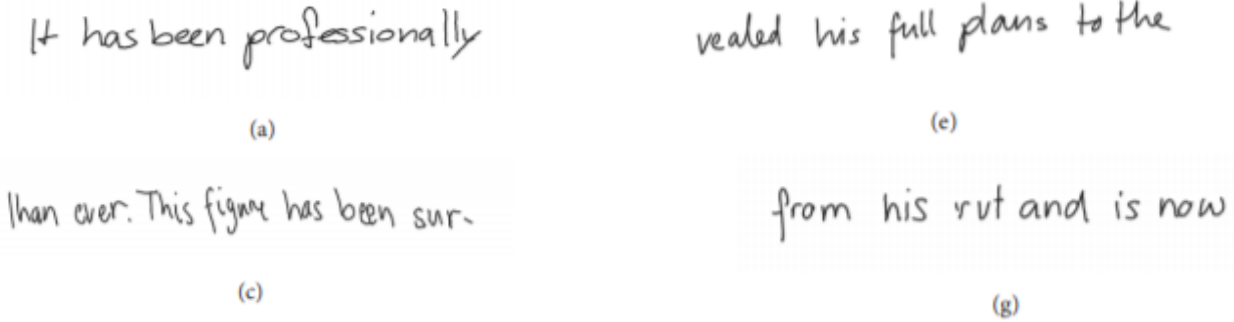
Demographic handwriting based classification problems, such as gender and handedness classifications, present important applications in disciplines like forensic biometrics. This work describes three different neural networks used for three different demographic problems: gender, handedness, and combined gender and handedness classifications. The data used for this study was gathered on a whiteboard then digitized. The data was acquired from “IAM On-Line Handwriting Database”. The considered problems present a high difficulty when extracting relevant features for discriminating the involved subclasses. Our solution is based on convolutional neural networks since these models had proven better capabilities to extract good features when compared to hand-crafted ones. Moreover, the proposed solutions have been designed using a unique network configuration for the three considered demographic problems. When the results of the neural nets constructed for this study are compared, it's seen that the separate networks created for each individual classification problems present higher success than the network constructed for the combined gender and handedness classifications.

Keywords: Convolutional Neural Networks, Demographic Classification, Biometry

1. GİRİŞ

El yazısı tanıma problemi; kişiler arası varyasyonlar, el yazılarının çeşitli nitelikleri, farklı kalem türlerinin kullanımı veya gürültülü arka plana sahip kağıt kullanımından dolayı çok sayıda değişkene bağlı olan bir problemdir [1]. Srihari, yaptığı çalışmada el yazılarının bireyselliğini inceledi ve belirledi [2]. El yazısı tanıma problemleri “Offline” ve “Online” olarak ikiye ayrılmıştır: [1]. “Offline” el yazıları, daha önce kağıda yazılan ve sonrasında sayısal ortama dökülen el yazısı metinlerinden oluşur. “Online” el yazıları ise bir tür elektronik sayısallaştırma cihazı (tablet vb.) kullanılarak yazılmış metinleri tanımayı amaçlamaktadır. El yazısı kullanılarak bireylerin farklı demografik kategorilere otomatik olarak sınıflandırılması [3-5] Adli Biyometri, Psikoloji, İnsan-Bilgisayar Etkileşimi veya Biyometrik Güvenlik gibi alanlarda ilginç uygulamalar sunmaktadır [6, 7]. Örneğin, bir suç mahallinde anonim bir el yazısı parçası bulunduğunda ve yazarın “sol elli bir kadın” olduğunu otomatik olarak tespit etmek mümkündür, bu durum araştırılacak şüpheli grubunu azaltma potansiyeline sahiptir. Ayrıca Psikoloji alanında da, yazarın el yazısı ve bazı kişilik özellikleri arasındaki korelasyonların kurulması durumunda, el yazısı stili üzerine yapılan araştırmalardan yararlanılabilir. İnsan-Bilgisayar Etkileşimi alanında da, bir kullanıcının cinsiyeti ve/veya elliliği otomatik olarak tahmin edilebilirse, bilgisayar uygulamaları daha kişiselleştirilmiş bir etkileşim sunabilir (örneğin, cinsiyet yönelimli reklamcılık). Biyometrik Güvenlik de el yazısı tahmininden yararlanma potansiyeline sahiptir, bilgisayar sistemlerine erişirken güvenliği artırmak için diğer biyometrik yöntemlerle birleştirilebilir. Bu el yazısı temelli demografik tahmin problemleri arasında cinsiyet, el, yaş aralıkları ve uyruk bulunmaktadır [8]. Bahsedilen denetimli (supervised) öğrenme problemleri ikili (binary) veya çok sınıflı olarak değerlendirilebilir. İkili sınıflandırma problemleri arasında el

yazılarından cinsiyet (erkek-kadın) tahmini ve yazımda kullanılan el tahmini (sağ-sol) en yaygın üzerine çalışma yapılanlar arasında yer alır. Çok sınıflı problemler arasında, yaş grupları tespiti, uyruk tespiti vb. problemler yer alır. Tüm bu sorunların bir özelliği, cinsiyet sınıflandırması durumunda olduğu gibi sınıflar arası veri dağılımı dengelenmiş olabilir (%50 kadın - %50 erkek) veya yazımda kullanılan el sınıflaması durumunda olduğu gibi sınıflar arası dengesiz bir dağılım bulunabilir (%90 sağ eli - %10 sol eli). Genel olarak, bu demografik sınıflandırma sorunları, insanlar için de çok karmaşıktır, çünkü hangi el yazısı özelliklerinin ilgili sınıfları uygun şekilde karakterize ettiğini bulmak oldukça zordur. Kadınsı yazının erkeksi olandan daha yuvarlak ve daha temiz olduğu kabul edilsede, erkeksi yazımın “kadınsı” görünümüne sahip olabileceği veya tam tersi olduğu durumlar da vardır. Şekil 1, “sağ elini kullanan bir erkek”, “sol elini kullanan bir erkek”, “sağ elini kullanan bir kadın” ve “sol elini kullanan bir kadın” tarafından yazılmış farklı el yazısı satırlarını göstermektedir.



Şekil 1: “Iam” veri setiden alınmış İngilizce el yazısı örnekleri. (a) sağ eli bir erkeği, (c) sol eli bir erkeği, (e) sağ eli bir kadını, (g) sol eli bir kadını temsil etmektedir.

1.1. Proje Amacı

Bu çalışma üç ayrı yapay sinir ağı yapısıyla; cinsiyet, yazımda kullanılan el ve bu problemlerin birleşimi için uygun olabilecek çözüm yolları sunmayı amaçlar.

Projeden almayı beklediğimiz yarar grafoloji biliminin sanal ortama taşınmasını destekleyerek bu alandaki sanallaşma akımına katkıda bulunmak ve buna ek olarak suç önlenmesi örneğin tehdit notları vb. durumlarda şüpheli listesinin daraltılması, ya da el yazısı ile herhangi bir müesseseye/ürüne değerlendirme notu bırakan müşterilerin cinsiyet dağılımının çıkarılmasını sağlamak ve bu sayede müşteri dağılım yüzdesinde çoğunluğa sahip olan kesime yönelik kampanya yapmak ya da ürüne o kitlenin ihtiyaçlarını karşılamaya yönelik güncellemeler yapılmasını sağlamaktır.

1.2. Alanda Yapılmış Çalışmalar

Raporun bu kısmında el yazısından yazının sahibini cinsiyetine, sağlak ya da solak olma durumuna ve hem cinsiyetine hemde sağlak ya da solak olma durumlarına göre sınıflandıran geçmişte yapılmış benzer çalışmalardan bahsedilecektir. Bluche [9], un 2016 yılında “Long Short-Term Memory (LSTM)” adlı sinir ağı sisteminin modifiye edilmiş bir versiyonunu kullanan ve tüm haldeki paragrafların işlenmesi ve tanınmasını gerçekleştiren bir sistem ortaya sunmuştur. Ancak bu sistem metindeki sözcük sayısını kısıtlaması yüzünden sadece kısıtlı sayıda sözcüklerin kullanıldığı metinlerde iyi tanıma oranları elde edebilmiştir.

“Off-line” el yazısından cinsiyet sınıflandırması yapan ilk sistemlerden biri 1996 yılında Hecker [10] tarafından ortaya atıldı. 96 erkek ve 96 kadından alınan el yazılarını kullanan ve otomatik olarak atanmış pixel yoğunluklarına sahip model sınıflandırmada %71,5 başarı oranı elde etmiştir.

2003 Koppel [11], British National Corpus (BNC) den elde ettikleri el yazısı dokümanlarını ve otomatik öğrenme algoritmalarını kullanarak bir model oluşturdu. Modelde kullanılan her doküman alakasız özelliklerin (feature) atılarak boyutları küçültülen bir özellik (feature) vektörü olarak ifade edilmiştir. Bu model cinsiyet sınıflandırmada %85 in üzerinde başarı elde etmiştir.

2004 yılında Tomai [12], CEDAR letter database’den aldığı offline karakterlerden elde ettiği özellikleri (features) K-Nearest Neighbor(KNN) algoritmasında kullanarak cinsiyet sınıflandırmasında %70’e varan başarı oranı elde etmiştir.

Liwicki [13], sırasıyla Destek Vektör Makinesi (SVM) ve “Gaussian Mixture Model (GMM)” kullanan iki tane online cinsiyet sınıflandırma yaklaşımı önerdi. Sonuçların değerlendirilmesi için yürütülen deneylerde IAM database kullanıldı ve SVM ile %62, GMM ile %67 başarı oranları elde edildi. Yine aynı yazar tarafından 2011 yılında tekrardan GMM ve IAM database kullanılarak yapılan çalışmada hem offline hem de online cinsiyet sınıflandırma problemlerinde %67,57 başarı oranı elde edilmiştir.

2014 yılında Al Maadeed ve Hassaine [8], araştırmalarını offline el yazısından cinsiyet tahmin etme üzerinde yoğunlaştırarak 2 ayrı yaklaşım kullandılar. Kullandıkları ilk yaklaşımda her birey aynı yazıyı yazarken kullandıkları ikinci yaklaşımda ise her birey birbirinden farklı yazılar yazdılar. Elde ettikleri özellikleri (features) Rassal Orman (RF) ve “Kernel Discriminant Analysis (KDA)” algoritmalarında kullanarak cinsiyet sınıflandırmasını yaptılar. Sistemin değerlendirilmesi aşamasında QUWI database den alınan Arapça, İngilizce ve her ikisinin de birleşiminden oluşan metinler kullanıldı. Elde edilen en iyi sonuçlar her iki dilinde birlikte kullanıldığı ve el ile yazılan metinlerin aynı olduğu veri grubunda elde edildi. Modelin bu şekilde RF ile %69,8 ve KDA ile %72,3 başarı elde etmesi sağlanmıştır.

Saran [14], a göre kişinin yazı yazarken hangi elini kullandığını kişinin yazıyı yazarken kalemli vuruş/dokunuş yönüne ve harflerin eğimlerine bakmak anlamak mümkündür. (Sol eli ile yazanlar sağ taraftan – sol tarafa doğru vuruş/dokunuş yapmakta ve harflerinin eğimi geriye doğru iken, sağ eli yazanlar ise bunların tam tersi özellikler üretiyor.)

2013 yılında Al – Maadeed [15], offline (el yazısı yazarken kaleminden/sensörlerden gelen dinamik bilgiyi kullanmadan) yöntem kullanarak yazı yazan kişinin yazıyı yazarken hangi elini kullandığını bulmaya çalışmıştır. Kişilerin vuruşlarından/dokunuşlarından çıkardıkları şekil ve eğrilik özelliklerini KNN sınıflandırma algoritmasında kullanarak QUWI database (Hem Arapça hemde İngilizce yazılar içeren bir database) üzerinde %71,5 başarı oranı elde etmişlerdir.

Bouadjenek [16], cinsiyet sınıflandırma için kullandıkları offline modelin aynısını (nitelik çıkarımı için Histogram of Oriented Gradients – HOG ve sınıflandırma algoritması olarak ise SVM) kullanıcının yazı yazarken hangi eli kullandıklarını tahmin etmek için de kullanmıştır. Bu model ile KHATT dataset ile %89,93 başarı oranı elde etmiştir.

Al Maadeed [17], yazı yazarken kullanılan eli bulmak için offline el yazısı ve “fuzzy logic” kullanan yeni bir yaklaşım ortaya atmıştır. 121 farklı kullanıcıdan topladığı elle yazılmış metinlerden şekiller ile ilgili yüksek miktarda nitelik çıkarmıştır. Modele temeli Bulanık Kavramsal Azaltmaya (Fuzzy Conceptual Reduction) dayanan ve “Lukasiewicz implication”ının uygulandığı boyut azaltma evresi de dahil edilmiştir. KNN algoritması kullanılarak yapılan sınıflandırmada model ortalama %83,43 başarı elde etmiştir.

1.3. Önerilen Çözüm Yolu ve Rapor Taslağı

Üç ayrı yapay sinir ağı yapısıyla; cinsiyet, yazımda kullanılan el ve bu problemlerin birleşimi için uygun olabilecek çözüm yolları sunulacaktır. Kurulması planlanan ilk iki yapı ikili (binary) problem çözümüne yönelik olup ikiye sınıflı olacaktır. Üçüncü ağ yapısı ise bu temel ikili sınıflandırma gruplarının alt sınıflara ayrılarak birleştirilmiş halini tahmin etmeyi amaçlayacaktır (sol eli kadın, sağ eli erkek vb.).

Kurulacak olan evrimsel sinir ağı mimarileri “LeNet-5” mimarisinden esinlenilmiş ve LeNet-5’e benzer bir yapıya sahip olacak şekilde tasarlanmıştır. El yazısı metni görsellerinin satırlara ayrılarak kullanılmasına karar verilmiştir. Problemin çözüm aşamasında göz önünde bulundurulması gereken en önemli faktör de veri setindeki sınıf dağılımlarındaki dengesizlik olmuştur. Veri seti incelendiğinde, cinsiyet sınıfların da yaklaşık olarak 1’e 2 oranında dengesizlik olduğu görülür, erkek sınıfının el yazısı fotoğrafları daha fazladır; kullanılan el sınıflarında ise yaklaşık olarak 1’e 10 oranında bir dengesiz dağılım vardır, sağ elle kullanılan yazılan metinler sol elle yazılanlara göre daha fazladır. Veri setindeki dengesizlik probleminin üstesinden gelmek için evrimsel sinir ağı kurulum aşamasında, ağdaki sınıfların başlangıç ağırlıkları verideki dengesizlik oranlarına göre değiştirilmiştir.

Rapor taslağı giriş bölümünden sonra; kullanılan araç ve yöntemler, gerçekleştirilen çalışma, elde edilenlerin değerlendirilmesi ve sonuç kısımlarını içerir.

2. KULLANILAN ARAÇ VE YÖNTEMLER

Projeyi hayata geçirmek için kullanılan temel yazılım dili altyapı desteği ve kullanım kolaylığından dolayı Python'dur. Python kullanılırken yararlanılması planlanan kütüphaneler ise: Pandas, NumPy, Matplotlib, Keras ve TensorFlow'dur.

- Keras, Python'da yazılmış açık kaynaklı bir sinir ağı kütüphanesidir. TensorFlow, Microsoft Cognitive Toolkit, R, Theano veya PlaidML'in üzerinde çalışabilir. Bizim durumumuzda burada TensorFlow devreye girer. Derin sinir ağları ile hızlı deneyler sağlamak için tasarlanmıştır dolayısıyla CNN yapısının kurulmasında temel rol oynayacaktır.
- TensorFlow, Google tarafından oluşturulan ve hızlı sayısal hesaplamalar yapmak için kullanılan bir Python kütüphanesidir. Doğrudan Derin Öğrenme modelleri oluşturmak için veya TensorFlow'un üzerine inşa edilen süreci basitleştiren sarmalayıcı kütüphaneler ile birlikte kullanılabilir bir kütüphanedir.
- Pandas, veri manipülasyonu ve analizi için Python programlama diline odaklı yazılmış bir yazılım kütüphanesidir. Özellikle, sayısal tabloları ve zaman serilerini değiştirmek için veri yapıları ve işlemleri sunar.
- NumPy, Python programlama dili için bir kütüphane olup, büyük, çok boyutlu diziler ve matrisler için destek sunar, bu diziler üzerinde işlem yapmak için geniş bir üst düzey matematiksel fonksiyon koleksiyonu içerir.
- Matplotlib, genel amaçlı GUI araç takımlarını kullanarak uygulamalara çizim yerleştirmek için nesne yönelimli bir API sağlar.

Çözümde verilerden değer üretmek için kullanılacak yöntem derin öğrenmenin bir dalı olan, elle üretilmiş niteliklerle kıyaslandığında daha yüksek başarı oranlarıyla kendini ispatlamış, Evrimsel Sinir Ağlarıdır. (Convolutional Neural Networks)

2.1) Derin öğrenme ve Evrimsel Sinir Ağları

Raporun bu bölümünde derin öğrenme ve Evrimsel Sinir Ağları hakkında bilgi verilecektir ancak derin öğrenmenin anlaşılabilmesi için önce makine öğrenmesi hakkında bilgi verilecektir.

Makine öğrenmesi kullanılan algoritmalar yardımıyla makinenin veriyi analiz etmesi, veriden öğrenmesi ve daha sonra yeni veriler hakkında kararlar vermesi ya da tahminlerde bulunmasıdır. Makine öğrenmesinin geleneksel programlamaya kıyasla farkı geleneksel programlamadaki gibi makinenin belirli bir görevi yerine getirmesi için elle tek tek girilmesi gerekli bir dizi talimatı yazmaktansa söz konusu makinenin çok sayıda veri kullanılarak eğitilmesi ve kullanılan algoritmaların makineye görevi yerine getirmesi için tek tek elle bir dizi talimat girilmesine gerek kalmadan o görevi nasıl yerine getirileceği öğrenmesini sağlamasıdır.

Derin öğrenme ise makine öğrenmesinin insan beyninin sinir ağlarının yapısı ve fonksiyonlarından esinlenilerek oluşturulmuş algoritmaları kullanan bir alt alanıdır. Sonuç olarak derin öğrenmede hayla verilerden öğrenen algoritmalar kullanılmaktadır ama bu sefer kullanılan algoritmaların yapısı beynin sinir ağlarının yapısı ve fonksiyonlarına dayanarak oluşturulmuştur. Derin öğrenme iki şekilde meydana gelebilir:

- Gözetimli Öğrenim (Supervised Learning)
- Gözetimsiz Öğrenim (Unsupervised Learning)

Gözetimli öğrenim derin öğrenme modelinin hâlihazırda etiketlenmiş veriler kullanılarak öğrenmesi ve bu etiketlenmiş verilerden bir sonuca varması ile son bulan öğrenimdir. Gözetimsiz öğrenim ise derin öğrenme modelinin etiketsiz veriler kullanarak öğrenmesi ve bu etiketsiz verilerden bir sonuca varmasıyla son bulan öğrenimdir.

Evrişimsel Sinir Ağlarında bulunan temel kavramlar işlenmeye başlanılmadan önce derin öğrenme modeli hakkında bilgi verilerek devam edilecektir. Derin öğrenmede model yerine aynı zamanda yapay sinir ağları kavramı da kullanılmaktadır. Raporun ilerleyen bölümlerinde bu iki kavram birbirlerinin yerine geçecek şekilde kullanılmaktadır.

Yapay sinir ağları beynin sinir ağlarından esinlenilmiş makine öğrenmesi sistemleridir. Yapay sinir ağlarının yapısını birbirine bağlı yapay nöron adı verilen birimler oluşturur. Bu nöronlar arasındaki her bir bağ bir nörondan diğerine veri iletebilir, sinyali alan nöron kendisine iletilen bu sinyali işler ve kendisine bağlı diğer nöronlara da bu sinyali iletir.

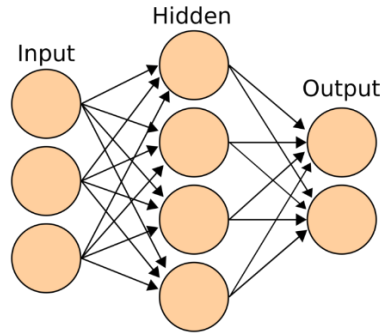
Nöronlar katmanlar halinde organize olmuşlardır. Bu katmanlar:

- Girdi (Input) Katmanı
- Gizli (Hidden) Katman(lar)
- Çıktı (Output) Katmanı

Olmak üzere 3'e ayrılır. Her bir katman kendi girdisi üzerinde birbirlerinden farkı dönüşümler uygulayabilir. Katmanlardaki veri akışı girdi katmanından çıktı katmanına doğru olur. Girdi ve çıktı katmanı arasında bulunan her katman gizli katman olarak adlandırılır. Yapay sinir ağlarında bu katmanlara ek olarak kullanılan başka katmanlarda bulunmaktadır. Bu katmanlar:

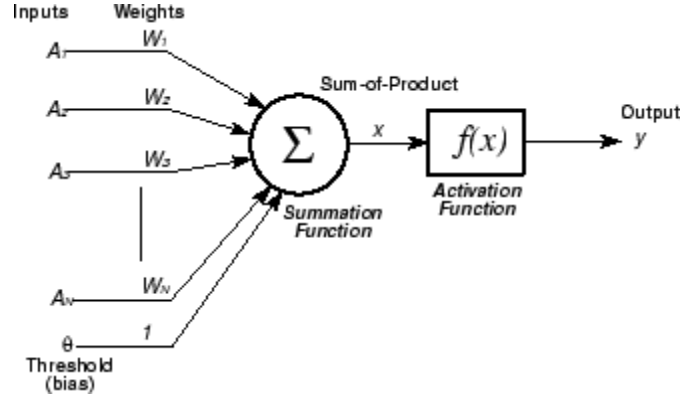
- Sık (Dense) Katman
- Evrişimsel (Convolutional) Katman
- Havuzlama (Pooling) Katmanı
- Tekrarlı (Recurrent) Katmanlar
- Normalizasyon (Normalization) Katmanı

Evrişimsel Sinir Ağlarının detayları anlatılan bölümde bu katmanlar detaylı bir şekilde açıklanacaktır.



Şekil 2: Girdi, Gizli ve Çıktı katmanlarından oluşan basit bir yapay sinir ağı örneği

Girdi katmanında bulunan 3 düğüm (Node) veri setindeki her bir örneğe ait ayrı birer özelliği (feature) temsil etmektedir. Girdi ve Gizli katman arasındaki her bir bağlantı bir önceki katmandaki (burada bu katman Girdi katmanı) çıktıyı bir sonraki katmana girdi olarak iletir. Bağlantıların çıktı olarak alıcı katmana ilettikleri değerin ne olduğunu anlamak için önce bağlantıların sahip oldukları ağırlıklardan (Weights) söz edilmelidir. Bir nöronun diğerine olan her bir bağlantı kendisine ait rastgele atanmış bir ağırlık değerine sahiptir ve bu ağırlık değerleri 0 ile 1 arasında değişim gösterebilmektedir. Ağırlıkları aynı zamanda nöronlar arasındaki bağlantıların kuvveti olarak da düşünebiliriz. Girdi katmanı girdiyi ilk aldığı anda o girdi bir sonraki katmana nöronlar arasındaki bağlantılar aracılığıyla ve o bağlantılara atanmış olan ağırlık değerleriyle çarpılarak iletilecektir. Hedef nörona bağlı her bir bağlantının bu çarpım ile elde ettiği değerler toplanıp bu toplam değeri hedef nöronun aktivasyon fonksiyonuna (Activation Function) verilir. Aktivasyon fonksiyonu kendisine verilen bu toplam değerini 0 ile 1 arasındaki bir sayıya dönüştürür. Aktivasyon fonksiyonundan çıktı olarak aldığımız bu değer bir sonraki katmanda bulunan hedef nöronun Girdisidir. Bağlantıların ağırlıkları ve aktivasyon fonksiyonunun işleyişi hakkında tüm bu anlatılanların görsel ile ifade edilmiş hali Şekil 3. de gösterilmektedir.



Şekil 3: Nöronun temel yapısı

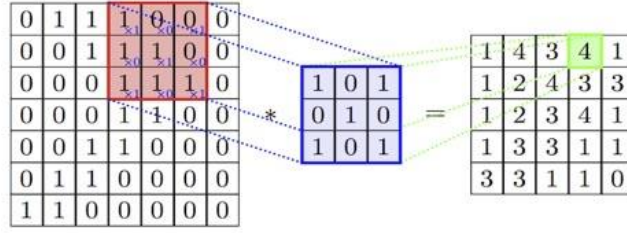
Yapay sinir ağları eğitilirken sistemin temel olarak yapmaya çalıştığı şey bir optimizasyon problemini çözmektir. Sistemin optimize etmeye çalıştığı şey işe bu bağlantılar arasındaki ağırlık değerleridir. Ağırlıkların nasıl optimize edileceği modelde kullanılan optimizasyon algoritmasına bağlı olarak değişir. Yapay sinir ağı modellerinde en çok kullanılan optimizasyon algoritması “Stochastic Gradient Descent” (SGD) algoritmasıdır. SGD algoritmasının amacı modelin kayıp fonksiyonunu (Loss Function) minimize etmektir. Minimize etmeye çalıştığımız bu loss değeri yapay sinir ağıımızın fotoğrafın tahmin ettiği etiket değeri ile fotoğrafın asıl etiket değeri arasındaki farktır. SGD algoritması nöronlar arasındaki bağlantıların ağırlıklarını bu loss fonksiyonunu sıfıra olabildiğince yaklaştıracak bir şekilde yeniden atamaya/güncellemeye çalışır. SGD’in yaptığı bu ağırlık güncellemeleri “back propagation” olarak adlandırılır.

Verinin modelden her bir geçişine devir (Epoch) denir. SGD her bir devir ile nöronlar arasındaki bağlantıların ağırlıkları üzerinde yapılan bu yeniden değer atama işleminin loss fonksiyonun değeri sıfıra mümkün olduğunca yaklaştırılıncaya kadar devam etmesi sonucu yapay sinir ağı öğrenmiş (nöronlar arasındaki bağlantıların ağırlıkları loss fonksiyonunu minimize eden değerlere ulaşmış) demektir.

Evrişimsel Sinir Ağları en çok resim analizinde kullanılmaktadırlar. Evrişimsel sinir ağları örüntü keşfedip bu örüntüleri anlamak ve bu örüntülerden bir anlam çıkarmakta uzmanlaşmış bir yapay sinir ağı olarak düşünülebilir. Evrişimsel sinir ağlarını çok katmanlı ileriye dönük bir yapay sinir ağı sınıfı olan Çok Katmanlı Algılayıcı (Multilayer Perceptron – MLP) dan ayıran şey evrişimsel katman olarak da adlandırılan gizli katmanlarıdır. Evrişimsel sinir ağlarında genellikle evrişimsel katmanlar dışında başka katmanlarda kullanılmaktadır ancak Evrişimsel Sinir Ağları’nın temelini bu evrişimsel katmanlar oluşturur.

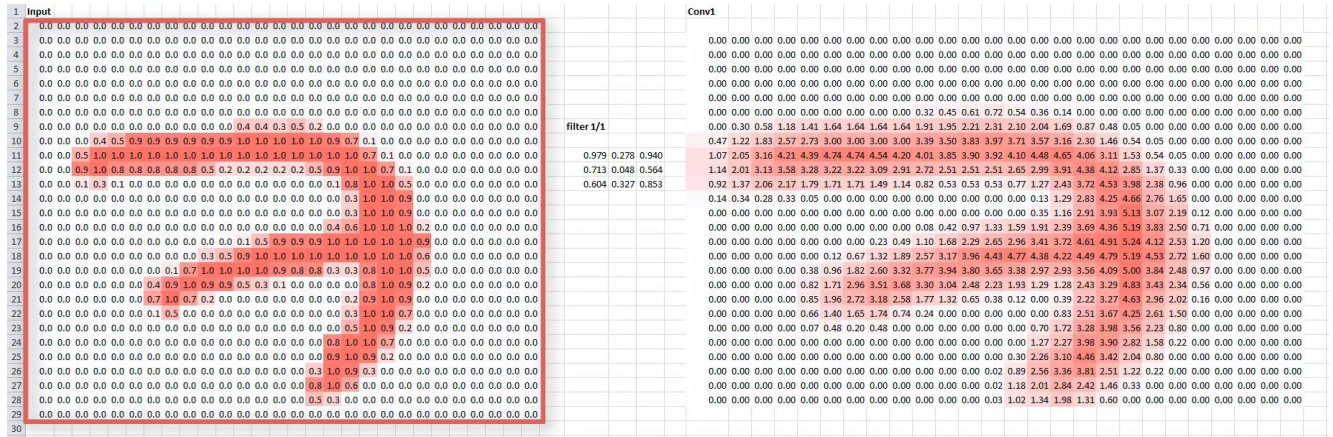
Tıpkı diğerleri gibi evrişimsel katman da bir girdi alır ve girdi üzerinde bir takım dönüşümler uygulayarak bu dönüşüme uğramış girdiyi çıktı olarak bir sonraki katmana iletir. Evrişimsel katmanlar örüntülerin farkına varabilmektedirler ve bu özelliklerini her evrişimsel katmanda kaç tane kullanılacağını belirtmemiz gereken filtrelerle borçludurlar. Örüntüleri tespit eden şey aslında bu filtrelerdir.

Filtrelerin fark ettikleri örüntülerden kasıt, verilen resimlerde bulunan birden çok kenar (Edge), şekil (Shape), doku (Texture), objeler (Objects) den herhangi biri olabilir. Filtrelerin tespit edebileceği örüntülerden biri olan kenarları tespit eden filtre kenar filtresi olarak adlandırılır, bazı filtreler köşelere ek olarak daire, kare gibi basit geometrik şekilleri de ayırt edebilirler. Bu basit ve bir bakımdan geometrik filtreler Evrişimsel Sinir Ağları’nın ön kısımlarında görülen şeylerdir, Evrişimsel Sinir Ağları’nın derinlerine gittikçe bu filtreler giderek daha komplike hale gelirler. Böylece daha derinlerdeki katmanlarda filtrelerimiz kenar ve basit şekilleri ayırt etmektense göz, kulak gibi belirli objeleri ayırt etmeye başlayabilir. Daha derindeki katmanların filtreleri ise köpek, kedi, kertenkele gibi daha da karmaşık objeleri ayırt edebilirler. Filtreler satır ve sütun sayılarına bizim karar verdiğimiz ve rastgele değerler atanmış küçük bir matris olarak düşünülebilirler. Şekil 4.’te 3x3 boyutlarında bir filtre görülmektedir.



Şekil 4: CNN’de 3x3 boyutlarında bir filtre

Şekil 4. de görülen 7x7 boyutundaki matris CNN’in girdi olarak aldığı resimdeki piksellerin 1 ve 0’lar ile ifade edilmiş halidir. Şekil 4. de görülen 3x3’lük filtre 7x7’lik matriste bulunan her bir 3x3’lük piksel bloğu üzerinde kayarak resim üzerindeki her bir 3x3’lük piksel bloğunu tarar. Bu kayma işlemine aynı zamanda Evriştirme de (Convolling) denilmektedir.



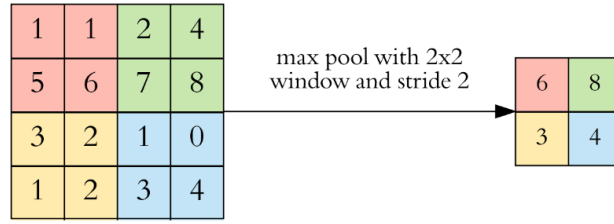
Şekil 5: Filtreleme işlemi sonrası oluşan boyut azalması

Filtre orijinal resmin üzerinde kayarak resmin tamamını taradığında filtreleme sonucu oluşan çıktı matrisinin boyutu filtreleme öncesi resmin boyutundan daha küçüktür. Boyuttaki bu düşüşü hesaplamak için kullanılan denklem, orijinal resim boyutunun $n \times n$ olduğu ve filtre boyutu $f \times f$ kabul edilirse:

$$\text{Çıktı Boyutu} = (n-f+1) \times (n-f+1) \quad (1)$$

Dolayısıyla bu modelde kullanılan resmin boyutunun her evrişimsel katmandan geçerken bir küçülmeye uğrayacağı anlamına geliyor ve bu istenmeyen bir durumdur. Özellikle hâlihazırda görece küçük boyutlu resimlerle model eğitilmeye başlanmış ise sadece birkaç evrişimsel katmandan geçtikten sonra resmin boyutu o kadar küçülebilir ki ortaya çıkan çıktı artık hiçbir anlam ifade etmez hale gelebilir. Bu durumun önüne geçmek için “Zero Padding” adlı yöntem kullanılmaktadır. Zero Padding orijinal girdi boyutunu korumamızı sağlayan bir tekniktir. Zero Padding de girdinin kenarlarına içindeki değerler 0 olan bir piksel matrisi ekleyerek orijinal resim girdisinin boyutu korunur. Bu sayede boyut küçülürken önemli olabilecek değerleri kaybetmek yerine sadece 0’lardan oluşan bu matrisler kaybolmuş olur.

Evrişimsel Sinir Ağları'nda kullanılan bir diğer katman (Filtre) ise "Max Pooling"dir. Max pooling Evrişimsel Sinir Ağları'nda genelde evrişimsel katmanların sonuna eklenir. Max pooling çalışma mantığı anlatılırken Şekil.6'dan yararlanılacaktır. Max pooling işleminde önce $n \times n$ boyutunda Max Pooling yaparken kullanılacak filtrenin boyutuna karşılık gelen bir matris tanımlarız. Max Pooling Şekil 6. üzerinden anlatılacağı için burada $n = 2$ 'dir. Sonra bu filtrenin atlama (Stride) miktarı belirtilir. Bu atlama miktarı ile filtremizin resim üzerinde kayarken kaç pixel boyu hareket edeceği belirtilir. Sonra boyutlarını 2×2 olarak belirlediğimiz filtremiz evrişimsel katmandan çıktı olarak gelen 4×4 matrisin ilk 2×2 'lik alanındaki değerler içinden en büyük değere sahip olan değeri kaydeder. Sonra 2×2 Max Pooling filtremiz daha önce belirttiğimiz kayma (Stride) miktarı kadar piksel yana ve sonrasında aşağıya kayarak 4×4 'lük matristeki her bir 2×2 'lik alandaki maksimum değerleri kaydeder. Max Pooling yapmaktaki amaç girdi boyutunu en önemli pikselleri koruyarak küçültmek ve bu küçülme sayesinde girdinin soyut bir formunu sisteme vererek "Over-Fitting" (ezberleme)'in önüne geçmektir. Max Pooling aynı zamanda girdideki parametre sayısını azaltarak sistemin yapması gereken işlem miktarını da düşürür.



Şekil 6: Max Pooling katmanının işleyiş mantığını temsili bir örnek

Bir Evrişimsel Sinir Ağı modelinde öğrenilebilir parametreleri veren denklem şu şekildedir:

$$\text{Öğrenilebilir Parametreler} = \text{Girdi Sınıf Ağırlıkları} \times \text{Çıktı Sınıf Ağırlıkları} + \text{Biases (Eğilimler)} \quad (2)$$

Bu denklemdeki eğilimler (Bias) önceden hesaplanmış ağırlıklı toplam değeri alıcı nöronun aktivasyon fonksiyonuna verilmeden önce ağırlıklı toplam değerine eklenir. Böylece eğilim (Bias) kullanılarak daha önce eşik değerinin (Threshold) altında kaldığı için ateşlenmemiş bir nöronu ateşleyebilir ya da daha önce ateşlenmiş bir nöronun ateşlenmemesini sağlayabiliriz. Sisteme eğilim (Bias) değerlerinin eklenmesi modelin veriyi öğrenirken ki esnekliğini arttırır.

Denklem (2)'ye geri dönecek olursak eğer bir önceki katman dense layer ise evrişimsel katmana gelen girdi (Input) sayısı önceki katmandaki düğüm (Node) sayısına eşittir. Eğer bir önceki katman evrişimsel ise evrişimsel katmana gelen girdi (Input) sayısı bir önceki evrişimsel katmanda bulunan filtre sayısına eşittir.

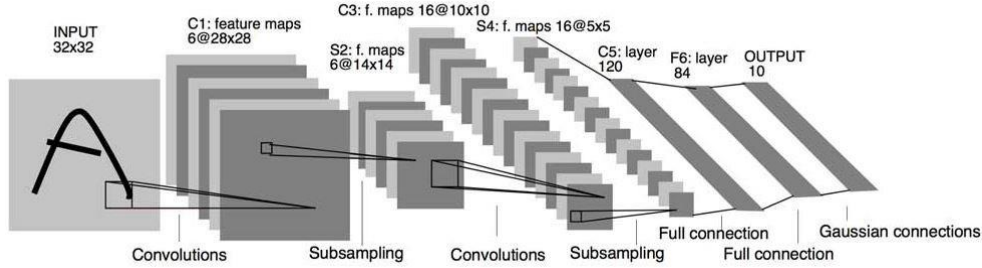
Bir evrişimsel katmandan çıkan çıktı (Output) sayısını denkleme dönecek olursak:

$$\text{Evrişimsel Katmanın Çıktı Sayısı} = (\text{Katmanda Kullanılan Filtre Sayısı}) \times (\text{Kullanılan Filtrelerin Boyutu}) \quad (3)$$

Denklemden yola çıkılarak, katmandaki eğilim (Bias) sayısı incelenecek olursa, o katmandaki filtre sayısına eşit olduğu gözlemlenir.

2.2) LeNet-5 Evrişimsel Sinir Ağı Mimarisi

Kurulacak olan evrişimsel sinir ağı mimarileri “LeNet-5” mimarisinden esinlenerek benzer yapıya sahip olacak şekilde tasarlanmıştır. LeNet-5 yedi adet evrişimsel katmandan oluşan ve bankacılık sisteminde çekler üzerine el ile yazılmış rakamları tanımak için üretilmiş bir Evrişimsel Sinir Ağı Mimarisi'dir. LeNet-5 mimarisi bir örnek ile açıklanacaktır.



Şekil 7: Örnek LeNet-5 mimarisi

Yukarıda verilen örnekte elle yazılan rakamlar 32 x 32’lik bir renksiz (grayscale) fotoğrafa dönüştürülerek dijitalize edilirler. Keşfedildiği zamanın hesaplama kapasitesinin limitli olmasından dolayı sistem büyük ölçekli resimlere ölçeklenebilir (Scalable) değildir. Model Girdi katmanı dışında 7 katmandan oluşmakta. Model az sayıda katmandan oluştuğu için her katman açıklanacaktır.

1. Katman: 5 x 5 boyutunda beş kernel’a ve 1 x 1 atlama (Stride) değerine sahip bir evrişimsel katman. Bu da demek ki 32 x 32 x 1 boyutundaki girdi fotoğrafı 28 x 28 x 6 boyutunda çıktıya dönüşür. Katmandaki toplam öğrenilebilir değişken sayısı (Bias) = 5 x 5 x 6 + 6 olur.

2. Katman: 2 x 2’lik kernel boyutuna ve 2 x 2 kayma (Stride) değerine ve toplamda 6 kernel’a sahip bir pooling katmanıdır. Bu pooling katmanındaki input değerleri toplanıp sonrada eğitilebilir parametre (her filtre için 1 tane) ile çarpılır, daha sonra elde edilen sonuç son olarak bias değeri (her bir filtre için 1 tane) ile toplanır. Elde edilen bu değer sigmoid aktivasyon fonksiyonuna verilir. Sonuç olarak bir önceki katmandan gelen 28 x 28 x 6 boyutundaki girdi 14 x 14 x 6 boyutuna indirgenir. Katmandaki toplam değişken sayısı = [1(egitilebilir parametre) + 1 (Bias)] x 6 = 12 olur.

3. Katman: Birinci katmana benzer olarak bu kamanda evrişimsel bir katmandır. Bu katmanın birinci katmandan farkı 6 yerine 16 katmana sahip olmasıdır. Bu da demek oluyor ki bir önceki katmandan gelen 14 x 14 x 6 boyutundaki girdi 10 x 10 x 16 boyutunda çıktı verir. Katmandaki toplam değişken sayısı = 5x5x16+16 = 416 olur.

4. Katman: Bu katmanda ikinci katmana benzer ancak ikinci katmandan farklı olarak 16 filtreden oluşan bir pooling katmanıdır. Çıktılar sigmoid fonksiyonundan geçtiği için bir önceki katmandan gelen 10 x 10 x 16 boyutundaki girdi 5 x 5 x 16 boyutuna indirgenir. Katmandaki toplam değişken sayısı = (1 + 1) x 16 = 32 olur.

5. Katman: Bu evrişimsel katman 5 x 5 kernel boyutuna ve 120 tane filtreye sahiptir. 5 x 5 x 16 boyutuna sahip girdi 1 x 1 x 120 boyutunda çıktı verir. Katmandaki toplam değişken sayısı = 5 x 5 x 120 = 3000 olur.

6. Katman: 84 parametreye sahip bir dense katman, yani 120 elemandan oluşan girdi 84 elemana dönüştürülür. Toplam değişken sayısı 84 x 120 + 84 = 10164 olur.

7. Katman: en sonda bulunan bu katman ise 10 adet potansiyel çıktı sınıfından oluşmakta Toplam değişken sayısı = 84 x 10 + 10 = 924 olmaktadır.

Son katmanda ise softmax aktivasyon fonksiyonu ile birlikte cross-entropy loss fonksiyonu tercih edilmiştir.

LeNet-5 mimarisini bir örnek ile açıklanmıştır ve çalışmada LeNet-5 mimarisinden esinlenerek türetilmiş olan Evrişimsel Sinir Ağı yapısı “Gerçekleştirilen Çalışma” bölümünde detaylarıyla incelenecektir.

2.2) Elde Edilen Sonuçları Değerlendirmek İçin Kullanılacak Yöntemler

Değerlendirme metriklerini anlamak için, metrik değerlerin açıklamalarına bir göz atalım. Doğruluk (Accuracy) en direct performans ölçüsüdür ve basitçe doğru tahmin edilmiş gözlemin toplam gözlemlere oranıdır. Yüksek doğruluk (Accuracy) varsa, modelimizin başarılı olduğunu ancak sadece yanlış pozitif ve yanlış negatif değerlerinin neredeyse aynı olduğu simetrik veri kümeleriniz olduğunda düşünülebilir. Bu nedenle, modelinizin performansını değerlendirmek için diğer parametrelere bakmak gerekir. TP = Doğru Pozitif Tahminler, FP = Yanlış Pozitif Tahminler, TN = Doğru Negatif Tahminler, FN = Yanlış Negatif Tahminler olmak üzere:

$$\text{Overall}_{\text{Accuracy}} = \frac{TP_c + TN_c}{TP_c + TN_c + FP_c + FN_c} \quad (4)$$

Kesinlik (Precision), doğru tahmin edilen pozitif gözlemlerin toplam tahmin edilen pozitif gözlemlere oranıdır. Bu metriğin cevapladığı soru, modelin sınıflandırma yaptığı durumlarda, kaç tanesinin doğru sınıfla eşleştirilmiştir. Yüksek kesinlik (Precision) düşük yanlış pozitif oranı ile orantılıdır.

$$\text{Precision}_c = \frac{TP_c}{TP_c + FP_c} \quad (5)$$

Hatırlama (Recall), doğru tahmin edilen pozitif gözlemlerin sınıfında bulunan tüm gözlemlere oranıdır. Hatırlamanın (Recall) cevapladığı soru; belli bir sınıfa ait örneklerden kaç tanesinin etiketlendiğidir.

$$\text{Recall}_c = \frac{TP_c}{TP_c + FN_c} \quad (6)$$

F1 Puanı, kesinlik (Precision) ve Hatırlamanın (Recall) ağırlıklı ortalamasıdır. Bu nedenle, bu puan hem yanlış pozitifleri hem de yanlış negatifleri dikkate alır. Doğruluk (Accuracy) kadar anlaşılması kolay değildir, ancak F1, özellikle eşit olmayan bir sınıf dağılımınız varsa, doğruluktan (Accuracy) genellikle daha kullanışlıdır. Yanlış pozitifler ve yanlış negatifler veri içinde yakın sayıda örneklere sahipse doğruluk en iyi sonucu verir. Yanlış pozitiflerin ve yanlış negatiflerin maliyeti çok farklıysa, hem Hassaslığa (Precision) hem de Hatırlama'ya (Recall) bakmak daha iyi bir yöntemdir.

$$F\text{-measure}_c = 2 \frac{\text{Precision}_c \times \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c} \quad (7)$$

Bahsedilen metriklere ek olarak bir de Destek (Support) vardır. Destek ise, sınıfta yer alan gerçek tahminlerin örnek sayısıdır.

Karmaşıklık matrisi, sınıflandırma problemine ilişkin tahmin sonuçlarının bir özetini sunar. Doğru ve yanlış tahminlerin sayısı sayım değerleri ile özetlenir ve her sınıf tarafından ayrılır. Bu, karmaşıklık matrisinin anahtarıdır. Karmaşıklık matrisi, bir bakıma ne tür tahminlerde ne kadar hata oranları olduğunu belirtir. Sadece sınıflandırıcı tarafından yapılan hatalar hakkında değil, yapılan hata türleri hakkında da bilgi verir. Karmaşıklık matrisini hesaplayabilmek için beklenen sonuç değerlerine sahip bir test veri kümesine ve doğrulama veri kümesine ihtiyacınız vardır. Test veri kümesindeki her değer için bir tahmin yapılır. Bu tahminler kullanılarak daha sonra bulgular bir tabloya veya bir matrise dizilir. Matrisin her satırı tahmin edilen bir sınıfa karşılık gelmektedir. Matrisin her sütunu ise gerçek bir sınıfa karşılık gelir. Daha sonra doğru ve yanlış sınıflandırma sayıları tabloya doldurulur. Bir sınıf için toplam doğru tahmin sayısı, o sınıf değeri için beklenen satıra ve bu sınıf değeri için öngörülen sütuna gider. Aynı şekilde, bir sınıf için toplam yanlış tahmin sayısı, o sınıf değeri için beklenen satıra ve bu sınıf değeri için öngörülen sütuna gider.

3. ÇALIŞMADA KULLANILAN VERİLER

Çalışmanın veri seti ‘‘Iam On-Line Handwriting Database’’ den temin edilmiştir. Veri seti beyaz tahta üzerine yazılmış el yazılarının dijital ortama geçirilmesiyle elde edilmiştir. 221 kişi tarafından farklı yapıda metinler beyaz tahtaya yazılmıştır. Kurulan ağa veri olarak bu metinler satır satır etiketlenerek verilmiştir. Her yazar ortalama olarak 5 satır katkıda bulunmuştur. Veri setindeki sınıfların dağılımı kurulacak üç adet sinir ağı için incelenirse:

- I. Cinsiyet sınıflandırma probleminin veri seti
 - 4493 adet kadın el yazısı satırı
 - 8456 adet erkek el yazısı satırı
- II. Kullanılan el sınıflandırma probleminin veri seti
 - 11804 adet sağ elle yazılmış satır
 - 1145 adet sol elle yazılmış satır
- III. Alt gruplarının sınıflandırılması için kullanılan veri seti
 - 412 adet sol elli kadın el yazısı satırı
 - 733 adet sol elli erkek satırı
 - 4081 adet sağ elli kadın el yazısı satırı
 - 7723 adet sağ elli erkek el yazısı satırı

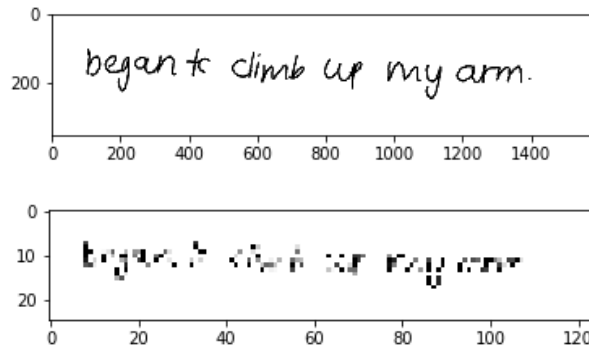
Veri setindeki dengesizlik probleminin evrışimsel sinir ağı kurulum aşamasında, ağdaki sınıfların (Çıktıların) başlangıç ağırlıklarının verideki dengesizlik oranlarına göre belirlenerek üstesinden gelinmesi planlanmıştır.

4. GERÇEKLEŞTİRİLEN ÇALIŞMA

Gerçekleştirilen çalışma kapsamında veri ön işleme, evrışimli sinir ağlarının kurulması ve elde edilen çıktılar incelenecektir.

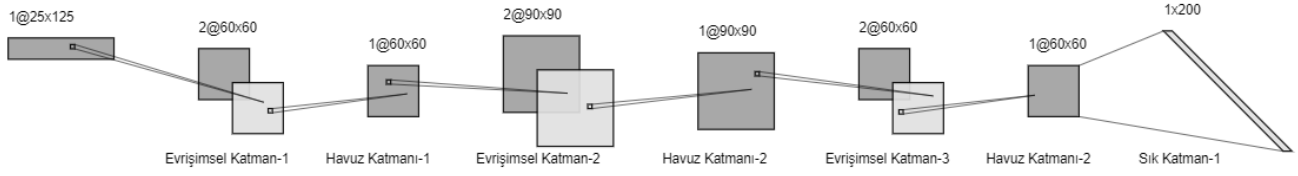
4.1. Veri Ön İşleme

Görseller elde edildikten sonra sınıflarına uygun dosyalara ayrılmaları için ‘‘Iam On-Line Database’’ tarafından sağlanan ayrıştırma (Parse) dosyası kullanılmıştır. Görseller siyah-beyaz (Grayscale) formata çevrilip, sabit 25 x 125 boyutlarına getirilmiştir. Kurulması planlanan ağlarda dengesiz sınıf veri örneklemelerinin ağı sınıflandırma yeteneklerini kısıtlamaması amacıyla sınıf veri miktar oranları hesaplanmıştır. Hesaplanan değerlerin ağların kurulum aşamalarında sınıf başlangıç ağırlıklarını belirlemede kullanılması planlanmıştır. Veri setleri her problem için %90’ı eğitim %10’ ı ise test olacak şekilde bölünmüştür. Veri sızdırılmasını (Data Leakage) önlemek için eğitim setindeki yazarların test setindeki yazarlardan farklı kişiler olmasına dikkat edilmiştir, burada amaç aynı yazarda görülebilecek potansiyel kalıplardan öğrenimi engelleyerek istenilenlerden farklı metriklerden başarı oranının artırılmasını önlemektir.



Şekil 8: Boyutları 25 x 125’e sabitlenmiş el yazısı görseli örneği

4.2. Kurulan Evrişimsel Sinir Ağı Mimarisi



Şekil 9: Kullanılan Evrişimli Sinir Ağı mimarisi

Çözümlerde kullanılan yapının temel mimari yapısı şekil 7. de temsil edilmiştir, görüldüğü üzere tasarlanan evrişimli sinir ağı mimarisi LeNet-5 yapısını andırır fakat arada bazı temel farklar bulunur. Kullanılan mimariden detaylı olarak bahsedilecek olursa:

- Girdi katmanında 25 x 125 boyutlarında görseller sisteme dahil edilir.
- İlk evrişimsel katman iki boyuta sahip ve 60 x 60 boyutlarındadır.
- İlk evrişimsel katmandan sonra ilk havuz katmanı bulunur.
- İlk havuz katmanı 90 x 90 boyutlarındaki iki boyutlu ikinci evrişimsel katmana bağlanır.
- İkinci evrişimsel katmandan sonra aynı boyuta sahip ikinci havuz katmanı bulunur.
- İkinci havuz katmanı 60 x 60 boyutundaki iki boyutlu üçüncü evrişimsel katmana bağlanır.
- Üçüncü evrişimsel katmandan sonra aynı boyutlarda üçüncü havuz katmanı bulunur.
- Son havuz katmanından çıkan veriler düzleştirme (Flattening) işleminden sonra sık katmana bağlanır.
- Sık katman ile çıktı katmanı arasında 0.5 büyüklüğünde bir adet eleme (Drop Out) katmanı bulunur.
- Eleme (Drop Out) katmanı son olarak çıktı döndüren katmana bağlanır.

Çıktı veren katmandaki düğüm sayısı cinsiyet ve kullanılan el tahmini üreten ağlarda 2, bu problemlerde bahsedilen sınıfların alt sınıflarını oluşturarak çözüm üretmeyi amaçlayan ağda ise potansiyel çıktı sayısına bağlı olarak 4'tür.

Evrişimli katmanlarda kullanılan çekirdek boyutları (Kernel Size) 4x4'dür. Kullanılan optimizasyon fonksiyonu "Adadelta" dır. Öğrenim hızı (Learning Rate) 1.0 olarak başlatılıp her devirde (Epoch) 0.95'lik miktarı ile bir sonraki devirde işlem yapılır. Çıktı katmanında "Sigmoid" aktivasyon fonksiyonu kullanılır.

Kurulması planlanan ağlarda dengesiz sınıf veri örneklemelerinin ağı sınıflandırma yeteneklerini kısıtlamaması amacıyla sınıf veri miktar oranları hesaplanmıştır. Hesaplanan değerler ağların kurum aşamalarında sınıf başlangıç ağırlıklarını belirlemede kullanılmıştır.

4.3. Elde Edilen Çıktılar

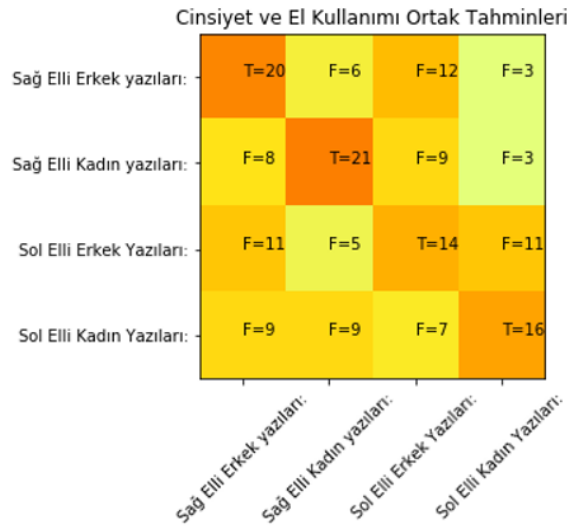
Ön işleme aşamalarından geçirilen görseller kurulan evrişimsel sinir ağlarına yerleştirilip, elde edilen çıktılar incelendiğinde ilk görülen, alt sınıflara ayrıştırılmış veri sınıflarıyla tahmin üretilen ağda, diğer ikili (binary) sınıflandırma yapan ağlara kıyasla çok daha düşük başarı oranları elde edilmiş olmasıdır. Kurulan iki ikili (Binary) sinir ağında da yaklaşık olarak %21 daha fazla genel başarı (accuracy) elde edilmiştir.

4 alt sınıf üzerine tahmin üretmek için tasarlanan evrimsel sinir ağından elde edilen test tahminlerine yönelik sonuç değerlendirme metrikleri incelenecek olursa:

Tablo 1: Kullanılan el ve cinsiyet tahmini yürüten ağı test sonuçları

	precision	recall	f1-score	support
Sağ Elli Erkek yazıları:	0.42	0.49	0.45	41
Sağ Elli Kadın yazıları:	0.51	0.51	0.51	41
Sol Elli Erkek Yazıları:	0.33	0.34	0.34	41
Sol Elli Kadın Yazıları:	0.48	0.39	0.43	41
accuracy			0.43	164

Modelin bulduğu tahminlerin genel başarı oranı sadece %43 olduğu, hassasiyet (Presicion) ve hatırlama (Recall) kullanılarak hesaplanan en yüksek F1 puanına “Sağ Elli Kadın Yazıları” sınıfının sahip olduğu görülür. Tahminlerdeki hataları daha iyi anlayabilmek için 4 alt sınıf üzerine tahmin üretmek için tasarlanan evrimsel sinir ağından elde edilen karmaşıklık matrisi incelenir, satırlar gerçek etiketleri sütunlar ise tahmini değerleri verecek şekilde görselleştirilmiştir:



Şekil 10: Cinsiyet ve El Kullanımı Tahminlerinin Karmaşıklık Matrisi

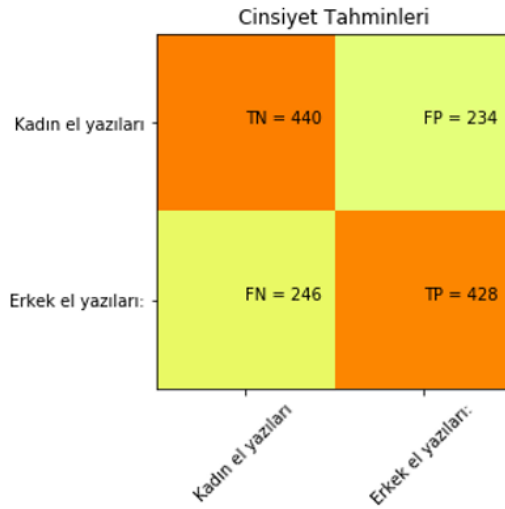
Karmaşıklık matrisinden de anlaşıldığı üzere yapay sinir ağı öğrenim gerçekleştirmeyi başarmıştır fakat başarısı oldukça düşüktür. Dolayısıyla problemler ayrıştırılarak ikili (Binary) problemler haline getirilip, tekrar tahmin üretilmiştir.

İkili (Binary) tahmine yönelik geliştirilen ilk yapay ağıın cinsiyet tahmini sonuçlarına dayalı test değerlendirme metrikleri şu şekildedir:

Tablo 2: Cinsiyet tahmini yürüten ağıın test sonuçları

	precision	recall	f1-score	support
Kadın el yazıları:	0.64	0.65	0.65	674
Erkek el yazıları:	0.65	0.64	0.64	674
accuracy			0.64	1348

Genel başarı oranı %64'e kadar ulaşan ağ yapısında sınıflar arasında yakın bir başarı oranına rastlamıştır. 4 alt sınıfa sahip ilk kurulan ağ yapısıyla karşılaştırıldığında ise genel başarı oranının %21 daha fazla olduğu saptanmıştır. Cinsiyet tahmini üretmek için tasarlanan evrimsel sinir ağından elde edilen karmaşıklık matrisi, satırlar gerçek etiketleri sütunlar ise tahmini değerleri verecek şekilde görselleştirilmiştir:



Şekil 11: Cinsiyet Tahminlerinin Karmaşıklık Matrisi

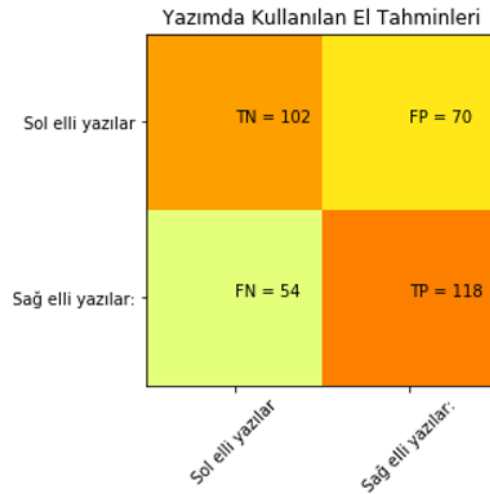
Cinsiyet tahmini için kurulan ağıın karmaşıklık matrisi de ilk kurulan ağ yapısının karmaşıklık matrisi ile karşılaştırıldığında daha belirgin ve başarılı tahminlerin ortaya çıktığı görülmektedir. Kadın el yazılarının eğitim setindeki örneklem sayılarının daha az olmasına rağmen doğru tahmin edilebilen daha fazla kadın el yazısı görseli olduğu görülür.

Son olarak, “kullanılan el” tahmini yapan evrişimsel sinir ağı kurulmuştur. Yazımda kullanılan el tahmini üreten ağı tahmin sonuçlarına dayalı test değerlendirme metrikleri şu şekildedir:

Tablo 3: Kullanılan el tahmini yürüten ağı test sonuçları

	precision	recall	f1-score	support
Sol elli yazılar:	0.65	0.59	0.62	172
Sağ elli yazılar:	0.63	0.69	0.66	172
accuracy			0.64	344

Yazımda kullanılan elin tahmini için kurulan ağ yapısında da cinsiyet tahmini için kurulan ağa eş değer olacak şekilde %64 başarıya ulaşılmıştır. Ayrıca tahminlerin F1 puanlarından “Sağ Elli Yazılar” sınıfında daha yüksek başarı elde edildiği görülür. Sınıflar arasındaki başarı farkı, veri setindeki kullanılan el sınıfları arasındaki 1’e 10 dengesiz dağılım göze alındığında aslında oldukça azdır. Yazımda Kullanılan El tahmini üretmek için tasarlanan evrişimsel sinir ağından elde edilen karmaşıklık matrisi sınıflar arasındaki başarı farkını daha iyi anlamak için, satırlar gerçek etiketleri sütunlar ise tahmini değerleri verecek şekilde görselleştirilmiştir:



Şekil 12: El Kullanımı Tahminlerinin Karmaşıklık Matrisi

Karmaşıklık matrisi üzerinde de görülebileceği üzere sol elli yazılar sınıfının az miktardaki veri sayısı, ağı öğrenimini başlangıç ağırlıklarının oranlanarak kurulmasına rağmen etkilemiş ve ağ daha çok sağ elli yazılara eğimli olarak tahmin üretmiştir.

5. SONUÇ ve DEĞERLENDİRME

Bu çalışmada, belirli el yazısına dayalı demografik sınıflandırma problemlerine Evrişimsel Sinir Ağlarıyla çözüm aranmıştır. Ele alınan üç sorun: cinsiyet, kullanılan el ve birleşik “cinsiyet ve kullanılan el” sınıflandırmaları olmuştur. Modelimizin eğitim ve test aşamalarında “Iam On-Line Database”den elde edilen veriler kullanılmıştır. Kurulan modellerimizin mimarileri yedi adet evrişimsel katmandan oluşan ve bankacılık sistemlerinde çekler üzerine el ile yazılmış rakamları tanımak için üretilmiş bir Evrişimsel Sinir Ağı Mimarisi olan “LeNet-5”ten esinlenerek tasarlanmıştır. Evrişimli sinir ağlarında, veri setindeki sınıfların dengesiz dağılımlarından doğabilecek eğimli öğrenimin önüne geçebilmek için, sınıfların ağlardaki başlangıç ağırlıkları (Weights) veri setindeki dengesizlik oranlarına göre belirlenmiştir. Kurgulanan modellerin test sonuçlarında görülmüştür ki, 4 adet sınıf oluşturmak yerine ayrı ayrı ikili (Binary) sinir ağlarında çözüm arandığında genel başarı (Accuracy) oranları her iki ağ için de yaklaşık olarak %21 daha yüksek olmuştur.

İkili (Binary) sınıflandırma çözümlerinde elde edilen test sonuçlarında daha yüksek başarı elde edilebilmiş olsa da veri setindeki sınıf dağılımlarının dengesizliği, kurulan ağların başarı oranlarını kısıtlamıştır. Bahsedilen kısıtlamaların üstesinden gelebilmek için gelecek çalışmalarda veri çoğaltmaya (Data Augmentation) yönelik işlemler içerebilir veya daha düzenli dağılmış veri setlerinden yararlanılabilir, böylece modelin sınıflandırma yaparken daha yüksek başarı oranları elde etmesi sağlanabilir.

6. KAYNAKÇA

- [1] R. Plamondon and S. N. Srihari, "On-line and of-line handwriting recognition: A comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63–84, 2000.
- [2] S. N. Srihari, S.-H. Cha, H. Arora, and S. Lee, "Individuality of Handwriting," *Journal of Forensic Science*, vol. 47, no. 4, pp. 856–872, 2002.
- [3] K. R. Bandi and S. N. Srihari, "Writer demographic classification using bagging and boosting," in *Proceedings of the Intl. Graphonomics Society Conference (IGS'05)*, pp. 133–137, 2005.
- [4] S.-H. Cha and S. N. Srihari, "A priori algorithm for sub-category classification analysis of handwriting," in *Proceedings of the 6th International Conference on Document Analysis and Recognition, ICDAR 2001*, pp. 1022–1025, IEEE, Seattle, WA, USA, September 2001.
- [5] S. Hamid and K. M. Loewenthal, "Inferring gender from handwriting in Urdu and English," *The Journal of Social Psychology*, vol. 136, no. 6, pp. 778–782, 1996.
- [6] J. Chapran, "Biometric writer identification: Feature analysis and classification," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 20, no. 4, pp. 483–503, 2006.
- [7] L. Ballard, D. Lopresti, and F. Monrose, "Evaluating the security of handwriting biometrics," in *Proceedings of the Proc.*, pp. 461–466, 2006.
- [8] S. Al Maadeed and A. Hassaine, "Automatic prediction of age, gender, and nationality in offline handwriting," *Eurasip Journal on Image and Video Processing*, vol. 2014, article no. 10, 2014.
- [9] T. Bluche, "Joint Line Segmentation and Transcription for End-to-End Handwritten Paragraph Recognition," in *Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS'16)*, Allen Institute for Artificial Intelligence, Barcelona, Spain, 2016.
- [10] R. Hecker, "The scientific examination of sex differences," *Proc. Fifty-Fourth Annual Meeting of the American Society of Questioned Document Examiners*, 1996.
- [11] M. Koppel, S. Argamon, and A. R. Shimoni, "Automatically categorizing written texts by author gender," *Literary and Linguistic Computing*, vol. 17, no. 4, pp. 401–412, 2003.
- [12] C. I. Tomai, D. Kshirsagar, and S. N. Srihari, "Group Discriminatory power of handwritten characters," *Document Recognition and Retrieval XI, Proc. of SPIE-IS & T Electronic Imaging*, vol. 5296, pp. 116–123, 2004.
- [13] M. Liwicki, A. Schlupbach, P. Loretan, and H. Bunke, "Automatic detection of gender and handedness from on-line Handwriting," in *Proceedings of the 13th Conf. Intl Graphonomics Society*, 2007.
- [14] V. Saran, S. Kumar, A. K. Gupta, and S. Ahmad, "Differentiation of handedness of writer based on their strokes and characteristic features," *Journal of Forensic Research*, vol. 4, no. 5, Article ID 1000204, pp. 1–3, 2013.
- [15] S. Al-Maadeed, F. Ferjani, S. Elloumi, A. Hassaine, and A. Jaoua, "Automatic handedness detection from of-line handwriting," in *Proceedings of the 2013 7th IEEE GCC Conference and Exhibition, GCC 2013*, pp. 119–124, IEEE, Doha, Qatar, November 2013.
- [16] N. Bouadjene, H. Nemmour, and Y. Chibani, "Histogram of Oriented Gradients for writer's gender, handedness and age prediction," in *Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications, INISTA 2015*, IEEE, Madrid, Spain, August 2015.
- [17] S. Al-Maadeed, F. Ferjani, S. Elloumi, and A. Jaoua, "A novel approach for handedness detection from of-line handwriting using fuzzy conceptual reduction," *Eurasip Journal on Image and Video Processing*, vol. 2016, no. 1, article no. 1, pp. 1–14, 2016.
- [18] Ángel Morera, Ángel Sánchez, José Francisco Vélez, and Ana Belén Moreno, "Gender and Handedness Prediction from Offline Handwriting Using Convolutional Neural Networks," *Complexity*, vol. 2018, Article ID 3891624, 14 pages, 2018.