

Öğrencilerin Mezunluk Alanlarını Tahmin Edebilme

(Predicting Students' Graduation Fields)

Lilas Muhammed

Bilgisayar Mühendisliği

İstanbul Topkapı

Üniversitesi

İstanbul, Türkiye

[lilasmuhammed@stu.topk](mailto:lilasmuhammed@stu.topkapi.edu.tr)

[api.edu.tr](mailto:lilasmuhammed@stu.topkapi.edu.tr)

Fadila Hachach

Bilgisayar Mühendisliği

İstanbul Topkapı

Üniversitesi

İstanbul, Türkiye

[fadilahachach@stu.topka](mailto:fadilahachach@stu.topkapi.edu.tr)

[pi.edu.tr](mailto:fadilahachach@stu.topkapi.edu.tr)

Musaab Al Taweel

Bilgisayar Mühendisliği

İstanbul Topkapı

Üniversitesi

İstanbul, Türkiye

[musaabaltaweel@stu.to](mailto:musaabaltaweel@stu.topkapi.edu.tr)

[pakpi.edu.tr](mailto:musaabaltaweel@stu.topkapi.edu.tr)

Özet— Yeni mezun öğrencilerin iş bulmada karşılaştıkları zorluklar, yalnızca üniversitelerden, öğrencilerden veya işyerlerinden kaynaklanmamakta, birçok farklı faktörün birleşimiyle şekillenmektedir. İş piyasası, öğrencilerin sahip olması gereken becerileri net bir şekilde belirlerken, üniversiteler bu becerileri tam anlamıyla öğretmekte yetersiz kalmaktadır. Bu durum, mezun olduktan sonra öğrencilerin hangi alanda uzmanlaşacaklarını belirlemede zorlanmalarına yol açmaktadır.

Bu raporda, öğrencilerin eğitim süreçlerinde doğru şekilde yönlendirilmesi ve gelişimlerinin sağlanması gerektiği vurgulanmaktadır. Proje, öğrencilerin akademik başarılarına dayalı olarak, hangi alanda uzmanlaşabileceklerini tahmin edebilen bir derin öğrenmesi modeli geliştirmeyi hedeflemektedir.

Anahtar Kelime - Yeni mezun, iş bulma zorluğu, üniversite eğitimi, iş piyasası, beceri geliştirme, makine öğrenmesi, akademik başarı, uzmanlık alanı, öğrenci yönlendirme, eğitim süreçleri, veri bilimi, oyun tasarımı, üniversite kulübü, kariyer gelişimi, mezuniyet sonrası destek.

Abstract— The challenges faced by new graduates in finding employment are shaped by a combination of various factors, not solely by universities, students, or employers. While the job market clearly defines the skills that students must acquire, universities often fall short in fully teaching these skills. As a result, graduates struggle to determine which

field they should specialize in after graduation. This report emphasizes the importance of correctly guiding students and ensuring their development throughout their educational journey. The project aims to develop a deep learning model that can predict the area in which students may specialize based on their academic performance.

Key words— New graduates, employment challenges, university education, job market, skill development, machine learning, academic performance, specialization, student guidance, educational processes, data science, game design, university clubs, career development, post-graduation support.

I. GİRİŞ

Her dört yeni mezun öğrenciden biri iş bulamıyor. peki bu sorun üniversitelerden mi ? yoksa öğrencilerden mi ? yoksa işyerlerindeki çalışma koşullarının zorluğundan mı kaynaklanıyor ? araştırdık ve sorunların neredeyse herkesten kaynaklandığını gördük. iş piyasası, öğrencinin mutlaka hakim olması gerektiği becerilere ihtiyaç duyar ve bu beceriler öğrencilere üniversite öğrenimi sırasında tam olarak öğretilmiyor.

Elbette çok fazla çözüm bulundu, bunlar arasında yeni mezun olan öğrencilere yönelik, onları iş hayatına hazırlayan eğitimler ve kurslar yer almaktadır, ama bu bir çözüm değildir bizim sorunumuz bundan daha derin.

Asıl problem öğrencinin eğitim süreci boyunca etkili bir şekilde yetiştirilmemesidir. Öğrenciyi eğitim süreci boyunca geliştirmeye yönelik çabalarımızı yoğunlaştırırsak, doğal olarak öğrenci mezun olduktan sonra kendisine uygun olan işi bulmakta zorlanmayacaktır. bu nedenle sonuç olarak sorun öğrencinin kendini doğru bir şekilde geliştirememesiyle ilgilidir.

Ancak öğrenci bu dönemde birçok engelle karşılaşiyor. en önemlisi: hangi yönde ilerlemeliyim? ne öğrenmeliyim? birçok öğrenci mezun olduğunda ya da mezun olduktan sonra hangi alanda uzmanlık yapacağını bilemiyor. ve tam olarak burada bizim projemiz ortaya çıkıyor, Üniversite sisteminde bir makine inşa edeceğiz, öğrencinin akademik notlarına dayanarak gelecekteki uzmanlık alanını tahmin edebilecek. Örnek olarak bir öğrenci Lineer Cebir, Olasılık ve İstatistik gibi derslerin yanında Python dersinde başarılıysa veri bilimine yönelebilir ve başarılı olabilir. veya bir öğrenci Matematik derslerinin yanında C# ya da Javada başarılıysa oyun tasarımına yönelebilir. ve bu şekilde öğrenciler sınıflandırıldıktan sonra eğitim süreçleri boyunca onlara destek olabilmek için her alanda uzmanlaşan bir üniversite kulübü kurabiliriz.

II. KULLANILAN VERİLER

Veri kaynağımız, bilgisayar mühendisliği bölümünde eğitim gören mezun öğrencilerin akademik verilerinden oluşan kapsamlı bir veri setine dayanmaktadır. Bu veri seti, öğrencilerin aldıkları derslerin notlarını ve mezuniyet sonrası yöneldikleri uzmanlık alanlarını içermektedir. Modelimizin temelini oluşturan bu veriler, öğrencilerin gelecekteki uzmanlık alanlarına dair tahminlerde bulunabilmek için kritik öneme sahiptir. Veri setinde, bilgisayar mühendisliği bölümündeki her öğrenci için 17 farklı dersin notları yer almaktadır. Bu ders notları, öğrencilerin genel akademik performansını ve başarı düzeylerini doğrudan yansıtmaktadır. Öğrencilerin akademik başarılarını modelimize aktarırken, bu derslerin her birinin bilgisayar mühendisliğine özgü beceri ve bilgi birikimi sağladığı göz önünde bulundurulmaktadır. Bu veriler, modelin doğru ve etkili bir şekilde öğrencilere uygun uzmanlık alanlarını tahmin edebilmesini mümkün kılmaktadır.

Projemizde kullanılan harf notları (AA, BA, BB vb.) önceden sayısal verilere dönüştürülerek işlenmektedir. Bu dönüşüm işlemi, **Label Encoder** adı verilen bir teknikle gerçekleştirilir. Harf notlarının sayısal verilere dönüştürülmesi, makine öğrenmesi algoritmalarının daha verimli çalışabilmesi için kritik bir adımdır. Sayısal veriler, bilgisayar mühendisliği öğrencilerinin performanslarının daha doğru bir şekilde değerlendirilmesini sağlar ve modelin tahmin doğruluğunu artırır. Bu dönüşüm, ders notları üzerinde yapılan analizlerin daha hızlı ve etkili bir şekilde yapılmasına olanak tanır, aynı zamanda modelin öğrenci profilini daha doğru bir şekilde öğrenmesini sağlar.

II. YÖNTEM

Bu projede, öğrencilerin notları, bölümleri ve diğer önemli faktörler gibi geçmiş verilere dayalı olarak öğrencilerin gelecekteki uzmanlık alanlarını tahmin etmekte Derin Sinir Ağı (DNN) modelini kullanıyoruz. Derin öğrenme, makine öğrenmesinin bir alt kümesi olup, giriş özellikleri ile çıktı tahminleri arasındaki karmaşık, doğrusal olmayan ilişkileri öğrenme yeteneği nedeniyle bu tür görevler için son derece uygundur ve eğitim verisi analitiği için güçlü bir araçtır.

Neden DNN?

DNN modeli, her katmanın giriş verilerini daha yüksek düzeyde özellikler çıkarmak için dönüştürdüğü birden fazla katmandan oluşur. Programımızın bağlamında, giriş verileri öğrencilerin notları, önceki bölümleri ve muhtemelen diğer özellikler, örneğin ders dışı faaliyetler veya sosyo-ekonomik arka plan gibi faktörleri içerir. Bu veriler, ağın veriyi daha soyut temsillerini öğrenmesine olanak tanıyan bir dizi gizli katman aracılığıyla işlenir.

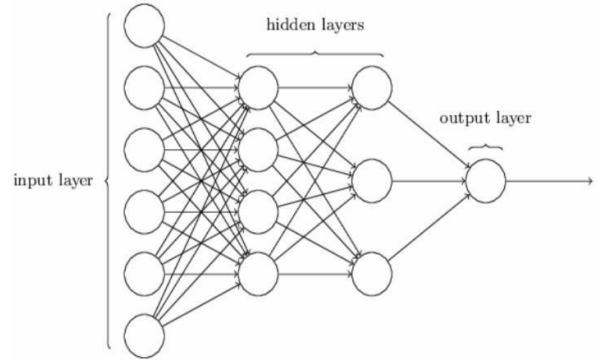
DNN'ler, büyük veri setlerinde karmaşık desenleri otomatik olarak tanımlayabilme yetenekleri nedeniyle bu görev için özellikle kullanışlıdır. Geleneksel istatistiksel modellerle el ile yapılması zor veya imkansız olan bu desenleri öğrenerek, DNN, öğrencinin muhtemel gelecekteki uzmanlık alanı hakkında doğru tahminler yapabilir ve bu da eğitim kurumları ve politika yapıcılar için değerli içgörüler sağlar.

Aktivasyon Fonksiyonları: Gizli Katmanlar için ReLU ve Çıktı için Softmax ReLU (Rectified Linear Unit), DNN modelimizin gizli katmanları için seçilen

aktivasyon fonksiyonudur. ReLU, modele doğrusal olmayanlık katarak, verilerde daha karmaşık desenleri öğrenmesine olanak tanır. Basitliği ve hesaplama verimliliği, derin öğrenme uygulamaları için mükemmel bir seçim olmasını sağlar. Fonksiyon kendisi şu şekilde tanımlanır:

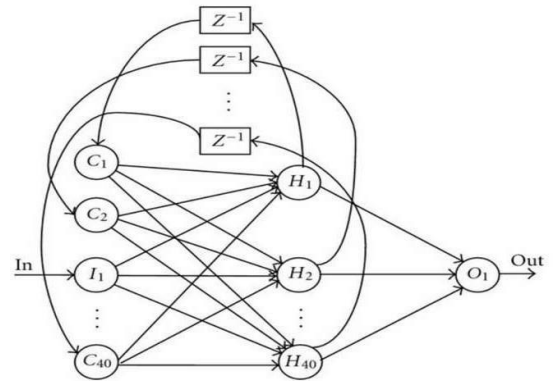
$$f(x) = \max(0, x)$$

Bu, negatif girişlerin sıfıra ayarlandığı, pozitif girişlerin ise olduğu gibi geçtikleri anlamına gelir. ReLU, sigmoid veya tanh gibi diğer aktivasyon fonksiyonları kullanıldığında derin ağlarda yaygın olan "kaybolan gradyan" problemini hafifletmeye yardımcı olur. Kaybolan gradyan problemi, gradyanların son derece küçük hale gelmesine neden olarak öğrenmeyi yavaşlatabilir. ReLU kullanarak, ağın verimli bir şekilde öğrenmesini sağlarız, hatta çok katmanlı ağlarda bile.



Şekil 1. Çok katmanlı sinir ağı modeli. [1]

Ancak, ReLU'nun "ölü ReLU" problemi yaşamayı mümkündür; bu durumda nöronlar sıkışabilir ve tüm girişler için sıfır çıktı verir. Bu, ağın ağırlıkları kötü bir şekilde başlatıldığında meydana gelebilir. Bunun önüne geçmek için ReLU'ya (örneğin Leaky ReLU veya Parametrik ReLU) çeşitli modifikasyonlar uygulanabilir, ancak bu modelin amaçları için standart ReLU, basitlik ve performans arasında en iyi dengeyi sunar.



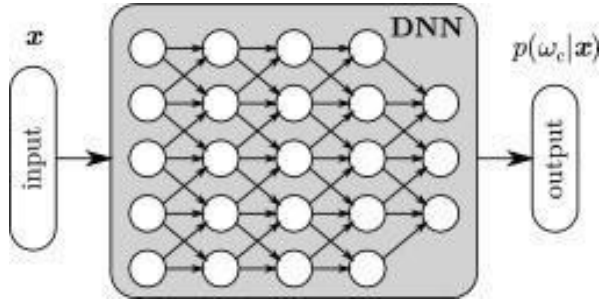
Şekil 2

Çıktı katmanı için Softmax aktivasyon fonksiyonu kullanıyoruz, bu fonksiyon özellikle öğrencinin uzmanlık alanını tahmin etmek gibi çok sınıflı sınıflandırma problemleri için uygundur. Öğrenciler birden fazla alanda uzmanlaşabileceği için, çıktı katmanı mühendislik, beşeri bilimler veya iş gibi farklı uzmanlık kategorileri arasında bir olasılık dağılımı sağlar.

Softmax, modelin ham çıktısını (herhangi bir reel sayı olabilen) her çıktıyı üssüne alarak ve ardından normalize ederek bir olasılık dağılımına dönüştürür:

$$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum(e^{x_j})}$$

Burada her x_i , çıktı katmanından gelen ham skor (logit) olup, i'nci uzmanlığa karşılık gelir. Sonuç, her değer öğrencinin belirli bir alanda uzmanlaşma olasılığını gösteren bir olasılık vektörüdür. Model, en yüksek olasılığa sahip olan uzmanlık alanını seçerek bir tahminde bulunabilir.



Şekil 3. Birbirine bağlı birçok nörona oluşan bir derin Sinir Ağı örneği.[2]

Neden ReLU ve Softmax?

ReLU ve Softmax kombinasyonu, DNN'nin hem giriş özelliklerinden karmaşık desenleri öğrenme hem de net, yorumlanabilir tahminler yapma konusunda iyi performans göstermesini sağlar. ReLU, karmaşık, doğrusal olmayan ilişkilerin verimli bir şekilde öğrenilmesini sağlarken, Softmax, farklı uzmanlık alanlarının olasılıklarına dayalı kararlar almak için ideal olan net, yorumlanabilir bir çıktı sunar.

Bu aktivasyon fonksiyonlarını kullanarak, model karmaşıklığı, verimlilik ve yorumlanabilirlik arasında bir denge kurar ve DNN'yi yüksek doğrulukla öğrenci uzmanlık tahmini yapmak için ideal bir seçim haline getirir.

III. ENTEGRASYON

Bu modelin üniversitenin mevcut bilgi sistemine entegrasyonu, öğrenci verilerinin daha etkin bir şekilde kullanılmasını ve akademik başarının tahmin edilmesinde dahadoğru sonuçların elde edilmesini sağlar.

Entegrasyon, üç temel yöntem aracılığıyla gerçekleştirilebilir: **Veritabanı bağlantısı**, **Web servisi (API)** ve **Kullanıcı arayüzü (Dashboard)**.

İlk yöntem olan **Veritabanı bağlantısı**, modelin üniversitenin mevcut öğrenci işleri sistemi (OIS) ile doğrudan entegrasyonunu içerir. Bu entegrasyon, öğrenci verilerinin otomatik olarak modele aktarılmasına olanak tanır ve öğrenci notları, ders bilgileri gibi verilerdeki güncellemeler doğrultusunda modelin tahminlerinin sürekli olarak doğru kalmasını sağlar. Örneğin, öğrencinin ders notları güncellendiğinde, bu yeni veriler modelin tahminlerini etkileyecek şekilde otomatik olarak işlenebilir. Böylece model her yeni öğrenci verisi eklendikçe güncellenir ve gerçek zamanlı veri ile tahminlerde bulunabilir.

Diğer bir yöntem olan **Web servisi (API)**, modelin bir web servisi olarak sunulmasını sağlar. Bu durumda, API aracılığıyla öğrenci verileri alınır, veriler üzerinde derin öğrenme modeli çalıştırılır ve tahminler, üniversitenin yönetim sistemine iletilir. API'nin en önemli avantajı, verilerin sürekli güncellenmesi gerektiği durumlarda bile modelin esnek bir şekilde çalışabilmesidir. Yani, API aracılığıyla model her zaman en güncel öğrenci verisi ile çalışarak doğru tahminler yapmaya devam eder.

Son olarak, **Kullanıcı arayüzü (Dashboard)**, öğrenciler ve akademik danışmanlar için etkileşimli bir platform sunar. Bu arayüzde, modelin tahmin ettiği uzmanlık alanları ve öğrenci profilleri görsel bir biçimde sunulabilir. Öğrenciler, hangi akademik alanlarda daha başarılı olabileceklerini kolayca görebilir, akademik danışmanlar ise bu verilere dayanarak öğrencilere daha verimli rehberlik yapabilir. Bu dashboard, öğrenci başarısını izlemek, rehberlik sağlamak ve öğrencilerin akademik kariyerlerinde doğru yönlendirmeleri yapmak için önemli bir araç haline gelir.

Entegrasyonun her üç yöntemi de, verilerin doğruluğunu artırırken, öğrencilerin akademik başarısını tahmin etme konusunda daha hassas ve etkili sonuçlar elde edilmesine olanak tanır. Bu sayede üniversite yönetimi, öğrenci destek hizmetleri ve akademik danışmanlar, daha hedeflenmiş ve veri odaklı kararlar alabilir.

Antrenman sonrası sonuç

Derin öğrenme modelimizi oluştururken, toplam parametre sayısı, modelin karmaşıklığını ve öğrenme kapasitesini etkileyen önemli bir unsurdur. Bu yazıda, modelimizdeki 23.00 parametreyi inceleyecek ve bu parametrelerin farklı katmanlar arasında nasıl dağıldığını açıklayacağız.

1. Embedding Katmanları

Embedding katmanları, kategorik verileri yoğun vektör formatında temsil etmek için kullanılır. Her kategorik özellik (örneğin, veri setimizdeki not sütunları) bir embedding'e dönüştürülür. Her embedding katmanındaki parametre sayısı şu şekilde hesaplanır:

$$\text{Embedding başına parametreler} = (\text{benzersiz kategori sayısı}) \times (\text{embedding boyutu}) + (\text{embedding boyutu})$$

Bu parametre sayısının hem bir özelliğin benzersiz değer sayısına hem de embedding boyutuna bağlı olduğunu gösterir.

1. Dense Katmanları

Embedding'ler düzleştirilip birleştirildikten sonra, bu özellik vektörü dense (tam bağlantılı) katmanlardan geçirilir. Bir dense katmanındaki parametre sayısı şu formül ile hesaplanır:

$$\text{Dense katmanındaki parametreler} = (\text{girdi sayısı}) \times (\text{nöron sayısı}) + (\text{nöron sayısı})$$

Parametre sayısı, her dense katmanındaki özellik sayısına ve nöron sayısına göre artar.[6]

Çıktı katmanındaki parametreler ise, önceki dense katmandaki nöron sayısına ve hedef değişkenindeki sınıf sayısına bağlıdır. örneğin 5 sınıftan oluşan bir sınıflandırma problemi için çıktı katmanındaki parametreler şu şekilde hesaplanır:

$$\text{Çıktı katmanındaki parametreler} = (\text{Önceki katmandaki nöron sayısı}) \times (\text{sınıf sayısı}) + (\text{sınıf sayısı})$$

4. Toplam Parametreler

Embedding katmanları, dense katmanları ve çıktı katmanından gelen parametreleri topladığımızda, modelimizdeki toplam parametre sayısını elde ettik. Modelimiz, toplamda **23.000 parametreye** sahiptir, çünkü her bir özellik için embedding katmanları ve ardından birleşik embedding vektörlerini işleyen dense katmanlar bulunmaktadır.

Modelimizdeki 23.000 parametre, eğitim sırasında tahminleri iyileştirmek ve ayarlamak için kullanılan öğrenilebilir ağırlıklar ve biaslardır. Daha fazla parametre bulunması, modelimizin verilerdeki karmaşık ilişkileri öğrenme kapasitesini artırmaktadır. Ancak, modelin mevcut verilere göre aşırı karmaşık olmaması önemlidir; aksi takdirde, bu durum aşırı uyuma (overfitting) neden olabilir. Doğru düzenleme teknikleri, yeterli veri ve dikkatli model ayarlamaları, modelin karmaşıklığı ile genelleştirme yeteneği arasında denge sağlamada kritik öneme sahiptir.[7]

IV. SONUÇ

Bu uygulama, çeşitli paydaşlara önemli fırsatlar ve faydalar sunarak eğitim süreçlerini daha etkili hale getirmeyi hedeflemektedir. İlk olarak, devlet ve eğitim politikası yapıcılar bu programı kullanarak sektördeki iş gücü ihtiyaçlarını ve eğitimdeki gelişen trendleri daha iyi anlayabilir. Bu analizler, eğitim programlarını endüstri gereksinimlerine göre yeniden şekillendirmek için politika geliştirme süreçlerine katkı sağlayacaktır. Üniversiteler ve kolejler de bu programdan büyük ölçüde faydalanabilirler. Akademik danışmanlar, öğrencilerin akademik başarılarını göz önünde bulundurarak onlara kişiselleştirilmiş rehberlik sunabilir ve güçlü yönlerine uygun dersler ve projeler seçmelerine yardımcı olabilir. Bu sayede, öğrenciler daha bilinçli kararlar alarak eğitim süreçlerini optimize edebilirler. Öğrenciler ise, program sayesinde akademik performanslarına dayalı olarak kariyer yollarını daha net

bir şekilde görebilir, hangi alanlarda başarılı olabileceklerini belirleyerek kariyerlerini doğru bir yönlendirme ile şekillendirebilirler. Bu yaklaşım, öğrencilerin mezuniyet sonrası iş gücüne daha hazırlıklı bir şekilde adım atmalarını sağlar.

V. DEĞERLENDİRME - TARTIŞMA

Bu projeyi geliştirirken, öğrencilerin mezuniyet sonrasında hangi alanlarda uzmanlaşabileceklerini tahmin eden bir sistem oluşturmayı hedefledik. Projemiz boyunca, akademik verilerden yola çıkarak bir Derin Sinir Ağı (DNN) modeli tasarladık ve bu modeli 7 farklı uzmanlık sınıfı için eğittik. Amacımız, öğrencilerin akademik notlarına dayanarak, hangi alanda başarılı olabileceklerini anlamaktı. Ancak bu süreçte kullandığımız verilerin tamamen sahte olduğunu belirtmek önemlidir. Kişisel Verilerin Korunması Kanunu (KVKK) gereği, üniversiteden mezun öğrencilerin gerçek verilerini temin edemedik. Bu nedenle, modelimizi eğitmek ve test etmek için gerçekçi ancak sahte bir veri seti oluşturduk. Bu veri seti, gerçek öğrenci verilerinden beklenen özellikleri ve desenleri yansıtmaya çalışsa da, elbette gerçek verilerle birebir aynı doğruluğa sahip değildir.

Eğer bu proje üniversite tarafından aktif olarak kullanılacak olursa, gerçek öğrenci verileriyle çalışarak çok daha doğru ve güvenilir sonuçlar elde edilebilir. Gerçek verilerin kullanılması, modelin öğrencilerin akademik performanslarını ve gelecekteki uzmanlık alanlarını tahmin etme konusundaki doğruluğunu büyük ölçüde artıracaktır. Ayrıca, gerçek verilerden beslenen bir sistem, üniversite yönetimlerinin öğrenciler için daha kişiselleştirilmiş rehberlik sağlamasına ve eğitim programlarını iyileştirmesine de yardımcı olacaktır.

Bu durumu göz önünde bulundurarak, projemiz hem sahte verilerle bir prototip oluşturmayı hem de gerçek verilerle kullanıma alındığında yüksek doğruluk sağlayacak bir temel altyapı geliştirmeyi hedefledi. Dolayısıyla, projemiz gerçek verilerle desteklendiğinde çok daha etkili sonuçlar ortaya koyacağına inanıyoruz.

Projede, veri setinin %80'ini modelin eğitimi için, %20'sini ise modeli test etmek için ayırdık. Bu yöntem, derin öğrenme modellerinde yaygın olarak kullanılan bir yaklaşımdır. Eğitim sürecinde model, verilen verilerden ilişkileri öğrenir ve bu bilgiler ışığında tahmin yapma kapasitesini geliştirir. Ayrılan test verileri ise modelin daha önce görmediği veriler üzerinde nasıl performans gösterdiğini değerlendirmek için kullanılır. Bu süreç, modelin genelleme yeteneğini anlamamıza yardımcı oldu ve doğruluk oranını daha gerçekçi bir şekilde ölçmemizi sağladı.

Makineyi eğitmek ve çalıştırmak için elimizdeki veri setini temizledik, harf notlarını sayısal değerlere dönüştürdük ve modelin anlayabileceği bir forma getirdik. Her bir sınıf, belirli bir uzmanlık alanını temsil ediyordu ve öğrencilerin bu alanlardaki olasılıklarını belirlemek için softmax aktivasyon fonksiyonunu kullandık. Bu süreçte, öğrencilerin notlarını giriş verisi olarak kullanarak tahminler yaptık. Modelin eğitim süreci boyunca kaç kez çalıştığını ve elde edilen doğruluk oranlarını dikkatle takip ettik. Proje sürecinde makinenin performansını değerlendirmek büyük bir öneme sahipti. Bu nedenle modeli çeşitli veri setleri üzerinde test ettik ve sonuçların doğruluğunu dikkatlice inceledik. Her bir test sonucunda modelin tahminleri daha güvenilir hale geldi ve öğrenci verilerindeki ilişkileri daha iyi analiz edebildi. Bu süreç, bizim için çok öğreticiydi çünkü hem derin öğrenme modelleriyle çalıştık hem de gerçek verilerle pratik yapma şansı bulduk.

REFERENCES

- [1] Gökçay, M. Murat. *Yapay sinir ağları ve uygulamaları*. MS thesis. Fen Bilimleri Enstitüsü, 1993.
- [2] Montavon, Grégoire, Wojciech Samek, and Klaus- Robert Müller. "Methods for interpreting and understanding deep neural networks." *Digital signal processing* 73 (2018): 1-15.
- [3] Sharma, Sagar, Simone Sharma, and Anidhya Athaiya. "Activation functions in neural networks." *Towards Data Sci* 6.12 (2017): 310-316.
- [4] Priya, B. "Softmax Activation Function: Everything You Need to Know."
- [5] Koech, K. E. "Softmax Activation Function—How It Actually Works. Medium." 2020,
- [6] Zhang, Chiyuan, et al. "Understanding deep learning (still) requires rethinking generalization." *Communications of the ACM* 64.3 (2021): 107-115.
- [7] Houlsby, Neil, et al. "Parameter-efficient transfer learning for NLP." *International conference on machine learning*. PMLR, 2019.