BİTKİ TRANSKRİPSİYON FAKTÖRLERİNİN HİBRİT DERİN ÖĞRENME İLE SINIFLANDIRILMASI

Ali Burak ÖNCÜL

2022
DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ

Tez Danışmanı
Dr. Öğr. Üyesi Yüksel ÇELİK

BİTKİ TRANSKRİPSİYON FAKTÖRLERİNİN HİBRİT DERİN ÖĞRENME İLE SINIFLANDIRILMASI

Ali Burak ÖNCÜL

T.C.
Karabük Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalında
Doktora Tezi
Olarak Hazırlanmıştır

Tez Danışmanı
Dr. Öğr. Üyesi Yüksel ÇELİK

KARABÜK
Temmuz 2022

Ali Burak ÖNCÜL tarafından hazırlanan "BİTKİ TRANSKRİPSİYON FAKTÖRLERİNİN HİBRİT DERİN ÖĞRENME İLE SINIFLANDIRILMASI" başlıklı bu tezin Doktora Tezi olarak uygun olduğunu onaylarım.

Dr. Öğr. Üyesi Yüksel ÇELİK .....
Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından Oy Birliği ile Bilgisayar Mühendisliği Anabilim Dalında Doktora tezi olarak kabul edilmiştir. 28/07/2022

Ünvanı, Adı SOYADI (Kurumu)	İmzası
Başkan : Prof. Dr. Mehmet Cengiz BALOĞLU (KÜ)	.....
Üye : Dr. Öğr. Üyesi Yüksel ÇELİK (KBÜ)	.....
Üye : Prof. Dr. Oğuz FINDIK (KBÜ)	.....
Üye : Dr. Öğr. Üyesi Yasin ORTAKÇI (KBÜ)	.....
Üye : Dr. Öğr. Üyesi Erdal BAŞARAN (AİÇÜ)	.....

KBÜ Lisansüstü Eğitim Enstitüsü Yönetim Kurulu, bu tez ile, Doktora derecesini onamıştır.

Prof. Dr. Hasan SOLMAZ .....
Lisansüstü Eğitim Enstitüsü Müdürü

"Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim."

Ali Burak ÖNCÜL

ÖZET

Doktora Tezi

BİTKİ TRANSKRİPSİYON FAKTÖRLERİNİN HİBRİT DERİN ÖĞRENME İLE SINIFLANDIRILMASI

Ali Burak ÖNCÜL

Karabük Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:
Dr. Öğr. Üyesi Yüksel ÇELİK
Temmuz 2022, 129 sayfa

Amino asit dizileri, protein yapısı ve amino asitlerin ilişkileri üzerine yapılan çalışmalar biyolojide hala büyük ve zorlu bir problemdir. Bu problemlerin çözümünde biyoinformatik çalışmalar ilerlemiş olsa da amino asitler arasındaki ilişki ve amino asitlerin oluşturduğu protein türünün belirlenmesi hala tam olarak çözülememiş bir problemdir. Proteinlerin kimliğini oluşturan motifler, aynı protein türünde dahi farklı farklı dizilişlere sahiptir ve bu yapı biyolojik olarak tespit edilebilmektedir. Bu sorun, mevcut protein dizilerinden bazılarının kullanımının da sınırlı olmasının nedenidir. Çünkü tür ve aile gibi çeşitli nitelikleri belirlemek için yapılan bu biyolojik deneyler maliyetli ve zaman alıcıdır. Bunun için de bu çalışmada proteinlerin türlerini belirlemek amacıyla hibrit bir derin öğrenme modeli tasarlanmış ve gerçeklenmiştir. Hazırlanan hibrit modelde, dizilerin yakınlık özellikleri için bir Word2Vec modeli, ardından özellik çıkarımı ve sınıflandırma için Evrişimli Sinir Ağları ve Çift Yönlü Kapılı Tekrarlayan Birim Ağları katmanları kullanılmış ve yüksek bir başarı ve hız ile

proteinlerin sınıflandırmasını yapmıştır. Modelin eğitiminde Bitki Transkripsiyon Faktörü Veritabanı (PlantTFDB)'ndan yararlanılarak oluşturulan bitki transkripsiyon faktör protein veritabanı kullanılmıştır. Önerilen bu hibrit model ve çift katlı çift yönlü LSTM modeli, hazırlanan bitki transkripsiyon faktör proteinleri veri seti ile sırasıyla %98.23 ve %97.80 test başarısına, %95.36 ve %96.60 f-skor değerine ve %98.07 ve %97.91 10-katlı çapraz doğrulama sonucuna ulaşmıştır. Hibrit model gerek ön işleme kısmının model başarısına yaptığı etki, gerekse CNN ve GRU mimarilerinin farklı özellik çıkarımı ve veri sınıflandırma alanlarındaki başarıları ile literatürde bir ilk olarak göze çarpmaktadır.

Ayrıca Basic Helix-Loop-Helix (bHLH) bitki transkripsiyon faktör proteinleri için bir referans veritabanı hazırlanmış ve bu veritabanının internet sitesi içerisine de Çift Yönlü Uzun Kısa-Vadeli Bellek Ağları temelli bir derin öğrenme sınıflandırıcısı eklenmiştir.

Hazırlanan model ile transkripsiyon faktör proteinleri başta olmak üzere diğer proteinler de sınıflandırılarak tür tanımlamasının verimli ve başarılı bir şekilde yapılması sağlanamıştır. Tasarlanan üçlü hibrit yapı bitki transkripsiyon faktörlerinin sınıflandırılmasında kullanılması literatüre kazandırılmış bir yenilik olarak öne çıkmaktadır.

Anahtar Sözcükler **:** Protein sınıflandırma, Derin öğrenme, LSTM, GRU, CNN, Hibrit model, Word2Vec.

Bilim Kodu **:** 92432

ABSTRACT

Ph. D. Thesis

CLASSIFICATION OF PLANT TRANSCRIPTION FACTORS BY HYBRID DEEP LEARNING

Ali Burak ÖNCÜL

Karabük University
Institute of Graduate Programs
Department of Computer Engineering

Thesis Advisor:
Assist. Prof. Dr. Yüksel ÇELİK
July 2022, 129 pages

The study of amino acid sequences, protein structure, and the relationships of amino acids is still a large and challenging problem in biology. Although bioinformatics studies have advanced in solving these problems, the relationship between amino acids and determining the type of protein formed by amino acids are still unsolved. The motifs that make up the identity of the proteins have different sequences even in the same protein type, and this structure can be determined biologically. This problem is why some of the available protein sequences are also limited in use. Because these biological experiments to determine species, family, etc., are costly and time-consuming. Therefore, in this study, a hybrid deep learning model was designed and implemented to determine the types of proteins. The prepared hybrid model used a Word2Vec model for the affinity features of the sequences, followed by CNN and Bidirectional GRU layers for feature extraction, classification, and classified proteins with high success and speed. In the training of the model, the plant transcription factor

protein database created by us using the Plant Transcription Factor Database (PlantTFDB) was used. This proposed hybrid and bi-layer bidirectional LSTM model had test success of 98.23% and 97.80%, f-scores of 95.36% and 96.60%, and 10-fold cross-validation of 98.07% and 97.91%, respectively, with the prepared plant transcription factor proteins dataset. This proposed hybrid model stands out as a first in the literature, with the effect of the preprocessing part on the model success and the success of the CNN and GRU architectures in different feature extraction and data classification areas.

In addition, a reference database for Basic Helix-Loop-Helix (bHLH) plant transcription factor proteins has been prepared, and a deep learning classifier based on Bidirectional LSTM has been added to this database's website.

With the prepared model, other proteins, especially transcription factor proteins, will be classified, and species identification will be made efficiently and successfully. The use of such a triple hybrid structure in the classification of plant transcription factors stands out as an innovation brought to the literature.

Key Word : Protein classification, Deep learning, LSTM, GRU, CNN, Hybrid model, Word2Vec.

Science Code : 92432

TEŞEKKÜR

Bu tez çalışmasının planlanmasında, araştırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteğini esirgemeyen, engin bilgi ve tecrübelerinden yararlandığım, yönlendirmeleri ve bilgilendirmeleriyle çalışmamı bilimsel temeller ışığında şekillendiren sayın hocam Dr. Öğr Üyesi Yüksel ÇELİK'e sonsuz teşekkürlerimi sunarım.

Biyoinformatik ve genetik alanındaki yoğun destekleri, bilgilendirmeleri ve veri desteği için tez izleme komitesi üyelerinden Kastamonu Üniversitesi Mühendislik ve Mimarlık Fakültesi Genetik ve Biyomühendislik Bölümü öğretim üyesi Prof. Dr. Mehmet Cengiz BALOĞLU hocama ve kendisinin doktora öğrencisi Necdet Mehmet ÜNEL'e teşekkür ederim.

Ayrıca tüm destekleri ve yönlendirmeleri için tez izleme komitesi üyelerinden Prof. Dr. Oğuz FINDIK hocama teşekkür ederim.

Kastamonu Üniversitesi Mühendislik ve Mimarlık Fakültesi Bilgisayar Mühendisliği Bölümündeki tüm hocalarıma eğitimim süresinceki tüm destekleri ve yardımları için teşekkür ederim.

Sevgili aileme tüm destekleri ve maddi, manevi hiçbir yardımı esirgemeden yanımda oldukları için tüm kalbimle teşekkür ederim.

İÇİNDEKİLER

Sayfa

Sayfa

Sayfa

Sayfa

ŞEKİLLER DİZİNİ

Sayfa

Sayfa

Sayfa

ÇİZELGELER DİZİNİ

Sayfa

Sayfa

SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

x : giriş değeri

y : çıkış değeri

W, R : ağırlık

b : bias

c : hücre değeri

i : giriş kapısı

z : güncelleme değeri

$f^{(t)}$: unutma kapısı

$o^{(t)}$: çıkış kapısı

σ : sigmoid aktivasyon fonksiyonu

g, h : aktivasyon fonksiyonu

erf(z) : hata işlevi

t : zaman

r : sıfırlama kapısı

f : giriş katmanından gelen değer

j : satır

k : sütun

o : çıkış görüntü boyutu

s : filtrenin kayma miktarı

$\forall$: her

β : hiper parameter

m, θ : önceki gradyanların üssel ortalaması

v : gradyanların kareleri

KISALTMALAR

DNA	:Deoxyribonucleic Acid (Deoksiribo Nükleik Asit)
RNA	:Ribonucleic Acid (Ribonükleik Asit)
TF	:Transcription Factor (Transkripsiyon Faktör)
bHLH	:Basic Helix-Loop-Helix (Temel Heliks-İlmek-Heliks)
HMM	:Hidden Markov Model (Gizli Markov Modeli)
BLAST	:Basic Local Alignment Search Tool (Temel Yerel Hizalama Arama Aracı)
GD	:Gradient Descent (Gradyan İniş)
SVM	:Support Vector Machine (Destek Vektör Makinesi)
NB	:Naive Bayes
KNN	:K-Nearest Neighbors (K-En Yakın Komşu)
CNN	:Convolutional Neural Network (Evrişimli Sinir Ağı)
RNN	:Recurrent Neural Network (Tekrarlamalı Sinir Ağı)
LSTM	:Long Short-Term Memory (Uzun Kısa Süreli Bellek)
GRU	:Gated Recurrent Unit (Kapılı Tekraryan Birim)
DNN	:Deep Neural Network (Derin Sinir Ağları)
PSI-BLAST	:Position-Specific Iterated Basic Local Alignment Search Tool (Konuma Özgü Yinelenen Temel Yerel Hizalama Arama Aracı)
PSSM	:Position Specific Scoring Matrice (Konuma Özgü Puanlama Matrisi)
NPIDB	:Nucleic Acid-Protein Interaction Database (Nükleik Asit-Protein Etkileşim Veritabanı)
SCOP	:Structural Classification of Proteins (Proteinlerin Yapısal Sınıflandırılması)
PDB	:Protein Data Bank (Protein Veri Bankası)
HSPIR	:Heat Shock Protein Information Resource (Isı Şoku Protein Bilgi Kaynağı)
bZIP	:Basic Region-Leucine Zipper (Temel Bölge-Lösin Fermuarı)
bZIPDB	:Basic Region-Leucine Zipper Transcription Factor Database (Transkripsiyon Faktörleri Veritabanı)
LEAPdb	:Late Embryogenesis Abundant Proteins Database (Late Embryogenesis Abundant Proteinleri Veritabanı)

PDIdb	:The Protein-DNA Interface Database (Protein-DNA Arabirimi Veritabanı)
PRIDB	:Protein-RNA Interface Database (Protein-RNA Arabirimi Veritabanı)
JGI	: Joint Genome Institute (Ortak Genom Enstitüsü)
PlantTFDB	:Plant Transcription Factor Database (Bitki Transkripsiyon Faktörü Veritabanı)
NCBI	:National Center for Biotechnology Information (Ulusal Biyoteknoloji Bilgi Merkezi)
CBOW	:Continuous Bag of Words (Sürekli Kelime Torbası)
ReLU	:Rectified Linear Unit (Doğrultulmuş Doğrusal Birim)
ADAM	:Adaptive Moment (Uyarlanabilir Moment Tahmini)
DP	:Doğru Pozitif
DN	:Doğru Negatif
YP	:Yanlış Pozitif
YN	:Yanlış Negatif
ROC	:Receiver Operating Characteristic Curve (Alıcı İşlem Karakteristiği)
ER	:Entity Relationship (Varlık İlişki)

BÖLÜM 1

GİRİŞ

Birçok veritabanında ve yapılan birçok araştırmada keşfedilmiş çok sayıda farklı amino asit dizisi vardır. Bu diziler farklı ailelere, farklı türlere ve farklı alemlere sahiptir. Bu dizilerden bazıları deneyler yoluyla sınıflandırılmış veya etiketlenmiştir. Ancak, yapılan birçok çalışmaya rağmen sınıflandırılmamış veya etiketlenmemiş verilerin miktarı oldukça fazladır. Bu amino asit dizilerinden oluşan proteinleri sınıflandırmak için kullanılan biyolojik yaklaşımlar, diğer canlılarda veya türlerde küçük farklılıklarla çalışan proteinleri farklı sınıflara ayırabilmektedir. Ancak bu proteinlerin dizilimleri incelendiğinde yüksek benzerlik gösterdikleri ve yapısal olarak benzer oldukları ortaya çıkarılabilmektedir. Durum böyle olunca da protein sınıflandırmasının gerekliliği gözler önüne serilmektedir. Protein sınıflandırması, belirli bir protein tipine veya belirli bir fonksiyona odaklanan araştırmacıların detaylı ve başarılı sonuçlar elde edebilmeleri için gereklidir [1].

Tıpkı insanların diller vasıtasıyla birbirleriyle yazılı ve sözlü iletişim kurmaları gibi canlı organizmaların vücudundaki iletişim ve düzen için DNA'yı, RNA'yı ve proteinleri adeta bir iletişim aracı gibi kullanmaları gerekmektedir. Anlaşmak için bir dilin yapısını ve kurallarını bilmek gerektiği gibi bitki, hayvan ve diğer alemlerdeki organizmaların iletişimini anlamak için de bu proteinlerin yapılarını ve türlerini bilmek gerekmektedir. Bu yapıları öğrenmek için ilk etapta biyolojik deneyler ve çalışmalar kullanılmıştır. Bu sayede belli yapılar ve işlevler öğrenilmiş ve bu bilgiler ışığında sınıflandırmalar yapılabilmiştir [1]. Tabii bu sınıflandırmaların yapısı yöntem itbariyle maliyetli ve görece zaman alıcı olmuştur. Ayrıca insan eliyle yapılan bu çalışmalarda araştırmacı kaynaklı hatalar da olabilmektedir.

Ek olarak, biyolojik çalışmalardan elde edilerek kaydedilen bu proteinlerin harflerle ifade edilen dizilerinin gözle veya görsel olarak incelenmesi sonucunda sınıflandırılabilmesi de neredeyse mümkün değildir. Bilgisayar bilimleri alanında çalışmalar yapan araştırmacıların dikkatini çeken bu diziler, yapılan istatistiksel temelli çalışmalara ilham ve kaynak olmuş ve alanda birtakım çalışmalar yapılmıştır [2]. Bu istatistik temelli çalışmalara Gizli Markov Modeli (HMM) [3] kullanılarak yapılan çalışmalar [4] ve Temel Yerel Hizalama Arama Aracı (BLAST) [5] birer örnek olarak verilebilir. Bu çalışmalar, gücünü istatistik biliminden almakta ve amino asitlerin dizi içerisinde belli konumlarda bulunma olasılığını belirleyerek bu olasılıklara göre dizilerin karakterini belirlemeye ve dizileri sınıflandırmaya çalışmaktadır. Bu uygulamalar tam açıklamalı (dizi ismi ve bitki ismi gibi bilgilerin yer aldığı) dizilerde daha başarılıdır ve eksik açıklama olduğunda başarı oranları daha düşük olacaktır [6,7].

Birçok farklı alanda ve farklı çalışmada kullanılan makine öğrenmesi ve derin öğrenme gibi yapay zekâ uygulamalarının biyoinformatik alanında ve alan içinde de protein sınıflandırmada kullanıldığı örnekler görülmüştür. Gradyan İniş Algoritması (GD), Destek Vektör Makinesi (SVM), Naive Bayes (NB) Sınıflandırıcısı ve K-En Yakın Komşu (KNN) Algoritması, bu amaçla kullanılan makine öğrenmesi algoritmalarına birer örnek olarak gösterilebilir. Bilgisayar bilimleri ve yapay zekâ alanındaki son gelişmelerle beraber literatüre kazandırılmış olan Yinelemeli Sinir Ağı (RNN) ve Evrişimli Sinir Ağı (CNN) tabanlı derin öğrenme algoritmaları da bu alandaki sınıflandıma problemlerinde farklı varyasyonlarıyla kullanılmaktadır [8]. Bu uygulamalarla beraber bilimin ve çoklu derin öğrenme modellerinin gelişmesiyle [9] beraber benzer veriler üzerinde çalışmalar yapılmıştır.

Bitki transkripsiyon faktör (TF) proteinleri, hücrelerin ve organizmanın ömrü boyunca doğru zamanda ve doğru miktarda ifade edilmelerini sağlamak için genlerin açılıp kapanmasını düzenleyen, bu şekilde bitkinin gelişmesinden, hücreler arası iletişimden, çevreye tepkiden, stress durumu yönetiminden hücre döngüsüne kadar birçok işlemi yöneten [10,11] ve biyolojik deneyler ile keşfedilmiş proteinlerdir. Bu tezin amacı; bitki transkripsiyon faktör (TF) proteinlerinden oluşan bir veri seti derleyip, hazırlanan özgün ve yeni nesil hibrit derin öğrenme modeli ve diğer derin öğrenme modelleri ile

bu veri setindeki proteinleri ailelerine (türlerine) göre sınıflandırmaktır. Bu hibrit derin öğrenme modeli, üç temel kısımdan oluşan yapısı itibariyle literatüre yenilik katmaktadır. Bu sayede biyolojik deneylerin vakit ve kaynak kaybının önüne geçilmesi, insan ve dış etken kaynaklı hataların önüne geçilmesi, gelecekteki DNA-protein ve protein-protein etkileşimi ve etkileşim bölgesi çalışmalarına temel oluşturması, analiz ve hizalamalarda etiketli ve birden çok veri yerine sade ve tek dizinin kullanımını sağlayarak zaman ve kaynak tasarrufu sağlanması ve ayrıca sonuç doğruluğunun artırılması amaçlanmıştır. Yine hazırlanan özgün çalışma ile birlikte literatürde yer alan BLAST ve benzer modeller gibi herhangi bir veritabanı sorgusuna gerek duyulmadan sonuç alınabilmektedir. Önerilen özgün model Word2Vec, CNN ve Çift Yönlü GRU olmak üzere üç kısımdan oluşan özgün bir hibrit yapıya sahiptir. Bu hibrit yapı içerisinde gelecek bölümlerde detaylı olarak açıklanacak olan Word2Vec ile bir sözlük oluşturulmuş, diziler vektörize edilmiş ve dizi içi yakınlıklar belilenmiş, CNN tabanlı kısımda CNN'in özellik çıkarımı başarısından yararlanmış ve Çift Yönlü GRU kısmında da Çift Yönlü GRU'nun da hafıza ve uzun-kısa vadeli bağımlılıktaki başarısından yararlanılarak hafif ve hızlı bir model oluşturulmuştur. Bu hibrit model ile beraber çalışma ve sonuç üretme süresi büyük ölçüde kısalmıştır ve literatürdeki en yüksek doğruluk oranına sahiptir. Önerilen diğer derin öğrenme modeli ise Word2Vec ön işleminin ardından iki katmanlı ve çift yönlü bir LSTM katmanı ile beraber hafif ve yüksek başarıda bir sınıflandırma sağlamıştır. Ayrıca çalışma esnasında hazırlanan bitki transkripsiyon faktör veri seti ve protein-vektör temsili de literatürdeki diğer çalışmalar için referans ve kaynak olacak, alandaki çalışmaların gelişmesine faydalı olacaktır.

Çalışmanın içerisinde bir de yeni nesil bir Basic Helix-Loop-Helix (bHLH) transkripsiyon faktör proteinleri veritabanı oluşturulmuştur. Bu veritabanı, literatürde kullanılan arama ve analiz araçlarını (HMM ve BLAST) ve bir derin öğrenme sınıflandırıcısını içermektedir. Bu veritabanı da içerdiği veriler ve araçlar ile literatüre yenilik getirmiştir.

Hazırlanan bu çalışma; genel bir çerçevede literatür taraması, yeni bir veri seti ve yeni nesil bir veritabanı ve tekli ve hibrit derin öğrnme modellerinin geliştirilmesi olmak üzere üç ana parçası bulunmaktadır. Hazırlanan tez ise yedi ana bölümden

oluşmaktadır. Birinci bölüm “Giriş” bölümü olup, burada çalışmanın tanıtıcı, kısa bir özeti sunulmuştur. İkinci bölümde, yapılan çalışma ile benzer alanda çalışılmış ve literatüre kazandırılmış önceki çalışmalar geniş bir literatür taraması ile sunulmuştur. Üçüncü bölümde yapılan çalışmanın konusu olan transkripsiyon faktörleri ve bHLH trankripsiyon faktörü anlatılmıştır. Dördüncü bölüm “Materyal Metot” bölümü olup, bu bölümde veri setinin ve veritabanının oluşturulma detayları, modelin geliştirilmesinde kullanılan teknolojiler ve veritabanı ile internet sitesi araçları anlatılmıştır. Beşinci bölüm olan “Deneysel Çalışmalar” bölümünde tekli modellerin, hibrit modellerin ve önerilen hibrit modelin tüm tasarım detayları geniş bir şekilde sunulmuştur. Altıncı bölüm “Bulgular ve Tartışma”ya ayrılmış ve tüm bu modellerin başarıları, yorumları ve sonuçları karşılaştırmalı olarak verilmiştir. Yedinci ve son bölümde ise sonuçlar çalışmanın yapısına ve amacına uygun olarak yorumlanmış, başarısı, literatüre ve bililme katkısı yorumlanarak ortaya konmuş ve tez sonuçlandırılmıştır.

BÖLÜM 2

LİTERATÜR ARAŞTIRMASI

Literatürde Deoksiribo Nükleik Asit, Ribonükleik Asit ve proteinlerin yapısına, işlevine, mensubu olduğu aileye başta olmak üzere birçok parametreye göre analiz, sınıflandırma hizalama gibi görevleri olan birçok çalışma vardır. Bu çalışmalar biyolojik temelli ve biyoinformatik temelli olmak üzere genel olarak değerlendirilebilir. Biyolojik deneyler, laboratuvar ortamında, temel yöneticisi ve karar vericisi büyük ölçüde alanında yetkin insan olan ve birçok kimyasal ve solüsyon yardımıyla yapılan çalışmalardır. Biyoinformatik çalışmalar ise biyolojinin başta moleküler biyoloji olmak üzere çeşitli alanlarını bilgisayar bilimlerinin gücü ile birleştirerek ortak bir ilerleme kaydeden çalışmalardır [12]. Bu çalışmalar, biyolojik deneylerden gelen verilerin depolanarak kullanılması, arama araçları, istatistiksel temelli sınıflandırıcılar ve hizalama araçları, makine öğrenmesi, yapay sinir ağı ve derin öğrenme temelli yapay zekâ çalışmaları olarak belirlenebilir.

Literatürde çeşitli protein dizilerini analiz etmek veya sınıflandırmak için bazı modeller ve uygulamalar önerilmiştir. Bu uygulamalar farklı veri setleri ve farklı tasarımlarla ön plana çıkmıştır. Enzimler üzerine yapılan çalışmalardan birinde, protein dizisinin enzim olup olmadığını, enzim ise enzimlerin alt fonksiyonel sınıflarını tespit eden çalışmada 3 katmanlı, KNN temelli bir model olan OET-KNN sınıflandırılmış ve veri türlerine göre ortalama %86 ila %98 arası başarı elde edilmiştir [13]. Gen ontolojisi alanları üzerinde çalışan bir modelde makine öğrenmesi modellerinden biri olan Destek Vektör Makinesi (SVM) modeli kullanılmış ve benzer başarılar elde edilmiştir [14]. Protein dizilerinin enzimatik fonksiyonlarını tahmin eden bir başka çalışma olan ECPred modeli, literatüre SPMap, BLAST-kNN ve Pepstats-SVM makine öğrenme modellerinin bir kombinasyonu olarak dahil edilmiştir. 6 ana sınıf, 55 alt şube sınıfı, 163 alt şubelerim alt sınıfı ve 634 substrat sınıfı dahil olmak üzere toplam 858 EC numarası için tahminler sağlamıştır [15].

Gen ontolojisi tabanlı protein fonksiyon tespiti yapan makine öğrenmesi modelinde ise PSI-BLAST tarafından üretilen çoklu dizi hizalamaları ve pozisyona özgü puanlama matrisi (PSSM) ile çalışan yapıdadır. Yeni genomlarda protein fonksiyonlarına açıklama ekler ve proteinlerin fonksiyonlarını tanımlamıştır [16]. Bu ve benzer çalışmalar, ana dizinin yanı sıra bazı ek bilgi ve özellikler gerektirdiğinden ve deneysel olarak elde edilen bazı açıklamaları içerdiğinden, doğrudan protein dizisinden tahmin ve sınıflandırma yapabilecek modellere ihtiyaç duyulmuştur [7].

Yalnızca dizileri kullanarak sınıflandırma yapan önceki çalışmalardan, Swiss-Prot veri seti ile önerilen Keras gömme ve Destek Vektör Makinesi (SVM) modeli kullanılarak yapılan çalışma yaklaşık %93 başarı sağlamıştır. Bu çalışmada SVM'ye ek olarak ProtVec protein-vektör temsil aracı kullanılmıştır. [17]. Yine Swiss-Prot veri setini ve Keras gömme, Derin Sinir Ağları (DNN), LSTM ve CNN ağlarını kullanan başka bir çalışmada %81.2 ile %91.24 arasında çapraz doğrulama başarısı elde edilmiştir. Bu çalışmada da ProtVec benzeri bir protein-vektör temsil aracı kullanılmıştır [8]. Daha önce Swiss-Prot veri setiyle eğitilmiş, derin öğrenme ağına sahip bir enzim sınıflandırma çalışması, %92 ila %97 oranında başarılı olmuştur. Bu modelde işlenmemiş ve işlenerek basitleştirilmiş veri setlerinde ileri beslemeli ve geri beslemeli ağlar ile CNN ve LSTM tabanlı ağlar ile yapılan deneyler sonucunda %98 başarıya ulaşılmıştır [7]. Burada alınan diziler üç karakterden oluşan kelimelere ayrılmış ve dizi her turda bir karakter kaydırılarak üç farklı k-mer grubu elde edilmiştir [17]. Bir veziküler taşıma proteinleri tanımlama çalışmasında, PSSM'ler ile CNN ve GRU modelleri kullanılmış ve %85.8 doğruluk elde edilmiştir [18].

Proteinlerin yapısı ve fonksiyonu hususunda hayati bir role sahip olan uzaktan protein homoloji tespiti için kullanılan bir uygulamada, dizi ön işlemesinde tek-sıcak kodlama, homoloji tespiti için ise Çift Yönlü LSTM ağı kullanılmış %97 başarı sağlanmıştır [19]. Pfam-seed veri seti kullanılarak hazırlanmış olan bir başka protein analiz çalışmasında ise yine bir LSTM ağı kullanılmış ve %95.8 başarı elde edilmiştir. Bu çalışmada diziler harflere ayrılmış ve kullanılan 20 temel amino asit için 1'den 20'ye kadar sayı verilerek sayısal dönüşüm yapılmış ve arta kalan nadir durumlar için 0 ataması yapılmıştır [20]. Protein dizilerinin sınıflandırılması için yapılmış olan bir transfer öğrenme çalışmasında ise 3 katmanlı bir LSTM ağı kullanılmış ve %85 başarı

sağlanmıştır [21]. Fonksiyonel protein sınıflandıma ve protein mühendisliği çalışmasında tek-sıcak kodlama, 2 boyutlu evrişimsel katman ve residual bloklara sahip bir sınıflandırıcı ağ kullanılmış ve %93.7 başarı elde edilmiştir [22]. Ayrıca birçok farklı protein sınıflandırma çalışmasında, bileşik-protein etkileşim tahminlerinde ve genlerden protein ifadesi tahminlerinde yapay zekâ modelleri sıklıkla kullanılmaktadır [23–25].

bHLH süper ailesi, hemen hemen tüm organizmalarda ve bitki aleminde birçok hayati görevde yer alır. bHLH süper ailesi üzerine yapılan ilk çalışmalar [26] bunun önemini ortaya koymuş ve o zamandan beri ilgiyle incelenen bir TF protein ailesi olmuştur [27]. bHLH ailesi bitki aleminde birçok çalışmanın odağı olmuştur (antosiyanin biyosentezi, globulin ekspresyonu, karpel, epidermal gelişim, fitokrom sinyalleşmesi ve diğerleri). *Arabidopsis thaliana*'da 118 bHLH proteini ve çeltik (*Oryza sativa*) genomunda 131 bHLH proteini tanımlanmıştır [28]. 3 bHLH geninin stoma oluşumu için organize bir şekilde çalışması ve birbirini izleyen stereo-tipik hücre bölünmeleri nedeniyle *Arabidopsis thaliana*'da stoma gelişmektedir [29]. bHLH alt ailelerinin çoğu, *Arabidopsis thaliana*, *Oryza sativa* gibi tohumlu bitkilerde ve erken farklılaşma gösteren kara bitkilerinde görülmüştür. Bununla birlikte, çalışılan diğer bitkilerle karşılaştırıldığında, yeşil ve kırmızı algler gibi klorofitlerde bHLH protein çeşitliliğinin daha düşük olduğu bulunmuştur. Bu paragraftaki tüm bilgiler dikkate alındığında, ilk kara bitkilerinin ortaya çıkmasından kısa bir süre sonra bHLH ailesinin büyüdüğü ve kara bitkilerinde korunduğu sonucuna varılmıştır [30].

Arabidopsis thaliana ve *Oryza sativa* bitkilerinin genom dizilerinde bulunan bHLH proteinlerinin sınıflandırılması, çiçekli bitkilerde protein çeşitliliğinin belirlenmesine katkıda bulunmuştur. Bununla birlikte, farklı bitkilerdeki çeşitli bHLH proteinlerinin evrimsel ilişkileri tam olarak bilinmemektedir. Bu çeşitliliğe katkıda bulunmak için bir çalışmada alg ve 9 tür kara bitkisinin genom dizilimi değerlendirilmiştir. Buna göre, kırmızı alg ve klorofit genomunda 5'ten az bHLH proteini kodlanırken, kara bitkilerinin genomlarında 100-170 bHLH proteini kodlanmıştır. Korunmuş bHLH alanlarının dışında, amino asitlerin birçok alt aileyi karakterize ettiği ve bu proteinlerin çeşitliliğinin 440 milyon yıldan daha uzun bir süre önce kara bitkileri tarafından tanıtıldığı düşünülmektedir [31]. Başka bir çalışmada *Fragaria vesca*'da (dağ çileği)

bulunan bHLH proteinlerinin kromozomal yerleşimlerinin belirlenmesi ve biyoinformatik analizleri yapılmıştır. Buna göre RT-PCR sonuçlarında 7 *FabHLH* geninin antosiyanin biyosentezinde rol oynadığı bulunmştur. Protein etkileşim ağları incelendiğinde bu genlerden 4'ünün meyve antosiyanin biyosentezi ve hormon sinyal iletimi ile ilişkili olduğu sonucuna varılmıştır [32].

Soya fasulyesi (*Glycine max*) genom dizisi, düzenleyici genlerin değerli işlevlerini anlamak için önemli bir avantaj sunmaktadır. Bu genlerin çoğunun mRNA düzeyinde ifade edildiğini gösteren kanıtlar vardır [33]. Transkripsiyon faktörleri arasında önemli bir yere sahip olduğu bilinen ve çok çeşitli fonksiyonel özelliklere sahip olduğu bilinen bHLH proteinleri hakkında çok yıllık ağaç türleri üzerinde birçok çalışma yapılmıştır. Bir örnek çalışmaya göre *Malus pumila* (elma) genomunda toplam 175 bHLH proteini tanımlanmış ve elma bHLH ailesinin transkripsiyon faktörlerinin evrimi ve yapıları belirlenmeye çalışılmıştır. Bu çalışma ve bu makalede yapılan araştırma sonucunda bitkilerde bulunan bHLH gen ailesi üyelerinin bitkiye özgü organların gelişmesinde ve karasal ortama uyumun düzenlenmesinde önemli roller üstlendiği sonucuna varılmıştır [34].

Bu tüm çalışmalarda kullanılan protein ve DNA dizileri rahat bir kullanım ve araştırma olanağı sağlamak için tutarlı, düzenli ve kolay ulaşılabilir bir şekilde depolanmalı ve bilim dünyasının kullanımına sunulmalıdır. Bu depolanma ve sunulma işlemi ancak veritabanları ile mümkündür.

Veritabanları, verilerin belli arama kriterleri kullanılarak kolayca erişilebilecek şekilde saklandığı ve organize edildiği, bilgisayarla işlenmiş arşivlerdir. Veritabanları, bilgisayar donanımları ve yazılımlarından oluşurmaktadırlar. Bir veritabanının geliştirilmesindeki ana amaç verilerin belli bir yapıda kayıt altına alınması ve bilgilerin daha sonra kolayca erişilebilirliklerinin sağlanmasıdır [35]. Her ne kadar bilginin geri çağırılması her veritabanının ana amacı olsa da biyolojik veri tabanlarında genellikle daha spesifik gereksinimlere de ihtiyaç duyulmaktadır. Bu gereksinimlerin başında veritabanına ilk yüklendiklerinde aralarındaki ilişki bilinmeyen bilgi parçalarının bağlantılarının tanımlanması gelmektedir. Örneğin ham dizileri içeren bir veritabanı, dizi homolojilerinin ve korunmuş motiflerin tanımlanması, transkripsiyon

faktörlerinin DNA üzerindeki bağlanma bölgelerinin belirlenmesi gibi hesapsal bir görevi yerine getirebilir [36]. Biyolojik veritabanları, bilimsel deneylerden, yayınlanmış literatürden, yüksek verimli deney teknolojilerinden ve hesaplama analizlerinden toplanan yaşam bilimleri bilgilerinin kütüphaneleri olarak nitelendirilmektedir [37]. Bu veritabanları, araştırmacılara çok çeşitli biyolojik olarak ilgili verilere erişme fırsatı sunmaktadırlar.

Literatürde farklı amaçlar için özelleşmiş çeşitli veritabanları yer almaktadır. Nükleik Asit-Protein Etkileşim Veritabanı (Nucleic Acid-Protein Interaction Database (NPIDB)), Protein Veri Bankası'ndan çıkarılan DNA-protein ve RNA-protein komplekslerinin yapılarından elde edilen bilgileri içermektedir. MySQL ile oluşturulmuş olan bir ilişkisel veritabanı tasarımı, web arayüzü ve nükleoprotein komplekslerinin biyolojik olarak anlamlı özelliklerinin çıkarılması için birtakım araçlar bulundurmaktadır. NPIDB, haftalık olarak güncellenen bir yapıdadır. Moleküller arası etkileşimlerin hesaplanması için, DNA bağlayıcı protein alanlarının ve verilerini içeren SCOP [38] ailelerinin sınıflandırılması için araçlar bulundurmaktadır. Ayrıca DNA-protein ve RNA-protein komplekslerinin tüm mevcut yapıları hakkında bilgiye erişim sağlamaktadır [39]. NPIDB'ın barındırdığı bilgiler Protein Veri Bankası (Protein Data Bank-PDB)'ndan otomatik olarak alınmakta ve veritabanının internet arayüzünde sunulmaktadır. Önceki yazılımların aksine JMol [40] görselleştirme yazılımı vasıtasıyla fotoğraflar yerine üç boyutlu görselleştirmeler yine veritabanının internet arayüzünde araştırmacılara sunulur [39].

Isı Şoku Protein Bilgi Kaynağı (Heat Shock Protein Information Resource (HSPIR)), altı ana ısı şoku proteini (HSP), yani Hsp70, Hsp40, Hsp60, Hsp90, Hsp100 ve küçük HSP'nin birleşik veritabanıdır. Isı şoku proteinleri (HSP'ler), yükseltilmiş sıcaklıklar dahil olmak üzere çeşitli stres koşullarına yanıt olarak tüm canlı organizmalarda güçlü bir şekilde sentezlenen özel bir protein grubudur. Gerçekleme için ise yapılan literatür çalışması sayesinde her bir HSP ailesi için standart isimlerin ve alternatif isimlerin kapsamlı bir listesi oluşturulmuş, yapılar ve bunlara karşılık gelen HSP dizileri Protein Data Bank'dan (PDB) alınmış, protein varoluş seviyesi protein seviyesinde ve transkript seviyesinde olanlar alınmıştır. Daha sonrasında ise veriler, ilgili kıstaslara göre işlenerek ilişkisel veritabanına eklenmiştir. Arama kısmında ise BLAST,

CLUSTALW ve HMM analiz araaçları kullanılmıştır. Örneğin birden çok dizinin karşılaştırılması CLUSTALW aracı kullanılarak yapılabilmekte ve ağaç Archæopteryx kullanılarak görselleştirilebilmektedir [41].

Temel Bölge-Lösin Fermuarı (Basic Region-Leucine Zipper-bZIP) proteinleri, ökaryotlarda farklı roller oynayan bir TF sınıfıdır. Bu proteinlerdeki arızalar, kansere ve çeşitli başka hastalıklara yol açmaktadır. Bu proteinlerle ilgili bilgiler içeren bZIP Transkripsiyon Faktörleri Veritabanı (bZIPDB) oluşturulurken 49 insan ZIP TF'si kullanılmıştır. bZIPDB'deki amaç, insan bZIP ailesinin gen düzenleyici ağının, örneğin düzenleyici modüllerin tanımlanması için değerlendirme veya referans bilgilerinin deşifre edilmesi için gerekli olan açık kaynaklı verilerin sağlanmasıdır. Veritabanı vasıtasıyla sağlanan bilgiler arasında doğrudan etkileşim, fosforilasyon, bZIP TF'ler ve diğer hücresel proteinler arası fonksiyonel ilişkiler, bZIP TF-Hedef Gen ilişkileri ve hücresel proteinler bulunmaktadır. Veritabanında zaman içerisinde yapılan güncellemelerde 721 protein etkileşimi ve 560 TF-hedef gen ilişkisi kaydedilmiştir [42].

Late Embryogenesis Abundant Proteinleri Veritabanı (Late Embryogenesis Abundant Proteins Database-LEAPdb), bitkilerden ve diğer organizmalardan Late Embryogenesis Abundant Proteinleri (LEAP) ile ilgili referans bir veritabanı olarak hazırlanmıştır. LEAP, diğer proteinleri düşük sıcaklıkla ilişkili olan osmotik stres veya kurumaya bağlı olarak agregasyondan koruyan, hayvanlar ve bitkilerde bulunan proteinlerdir [43]. LEAPdb, Pfam, Korunan Etki Alanı ve InterPro veritabanları tarafından tanımlanan 8 LEAP alt-ailesi hakkındaki verileri toplamaktadır. LEAPdb tüm veri tabanı hakkında bilgi almak için bir inceleme arayüzü vardır. Gelişmiş metin ifadesi, amino asit motifleri ve diğer kullanışlı parametreler gibi çeşitli kriterleri kullanan bir arama arayüzü, rafine edilmiş alt grup girişlerinin alınmasına izin vermektedir. LEAPdb ayrıca dizi benzerlik araması yapan araçları da araştırmacılara sunmaktadır. Bilgi, verilerin analizini kolaylaştıran yeniden sıralama tablolarında görüntülenmektedir. Tüm bilimsel çalışmalarda olduğu gibi LEAP ile ilgili de yayınlanan, elde edilen verilerin artması, bu verilerin organize edilmesini, sınıflandırılmasını ve kullanılmasını zorlaştırmaktadır. Bu sebepten de LEAPdb'nin

bir amacı da bilim insanlarının çok miktarda veriyi gezinmesini, yorumlamasını ve anlamasını kolaylaştırmaktır [44].

Protein-DNA Arabirimi Veritabanı (The Protein-DNA Interface Database-PDIdb), X-ışını kristalografisi ile çözülen ve Protein Data Bank'ta bulunan Protein-DNA komplekslerinin ilgili yapısal bilgilerini içeren bir depodur. Veritabanı, üç hiyerarşik seviyeden oluşan protein-DNA komplekslerinin basit bir fonksiyonel sınıflandırmasını içermektedir. Bunlar; sınıf, tür ve alt tür olarak üç sınıfta belirtilmektedir. Bu sınıflandırma, PDIdb araştırmacıları tarafından PDB, PubMed, CATH, SCOP ve COPS gibi çeşitli kaynaklardan toplanan bilgilere dayanarak tanımlanmış ve elle seçilmiştir. Bu veritabanının ana amacı, DNA ve proteinler arasındaki moleküler tanıma sürecinin altında yatan ana kuralların anlaşılmasına katkıda bulunmaktır. PDIdb, PHP frameworkü, AJAX teknolojisi ve MySQL veritabanı yönetim sistemi kullanılarak gerçeklenmiştir. İlişkisel veritabanı içerisinde temel ve gelişmiş olmak üzere iki çeşit arama yapılabilmektedir. Temel arama, bir anahtar kelime girilmesiyle yapılabilirken; gelişmiş arama yapısı, birçok alt sorgunun birleşiminden oluşmaktadır [45].

Protein-RNA Arabirimi Veritabanı (Protein-RNA Interface Database-PRIDB), PDB içerisindeki komplekslerden çıkarılan protein-RNA arayüzlerinin kapsamlı bir veritabanıdır. İstatistiksel analizler ve makine öğrenmesi uygulamaları için protein-RNA arayüzlerinin kullanıcı tanımlı veri kümelerinin otomatik oluşturulmasına ek olarak, tek tek protein-RNA komplekslerinin ve ara yüzlerinin detaylı analizini kolaylaştırmak için tasarlanmıştır. Seçilen herhangi bir PDB kompleksi veya kompleksler listesi için PRIDB, etkileşimli protein ve RNA zincirlerinin birincil sekansları içinde ara yüzey amino asitleri ve ribonükleotitleri göstermektedir. 3B kompleks yapıları bağlamında etkileşimli atomların ve kalıntıların görselleştirilmesi işlemi ise JMol uygulaması ile gerçeklenmiştir. PRIDB, Apache, AJAX ve PHP 5 ve MySQL kullanarak tasarlanmış bir ilişkisel veritabanı ve web kaynağıdır. Diğer bazı işlevler için ise Perl programlama dili kullanılmıştır [46].

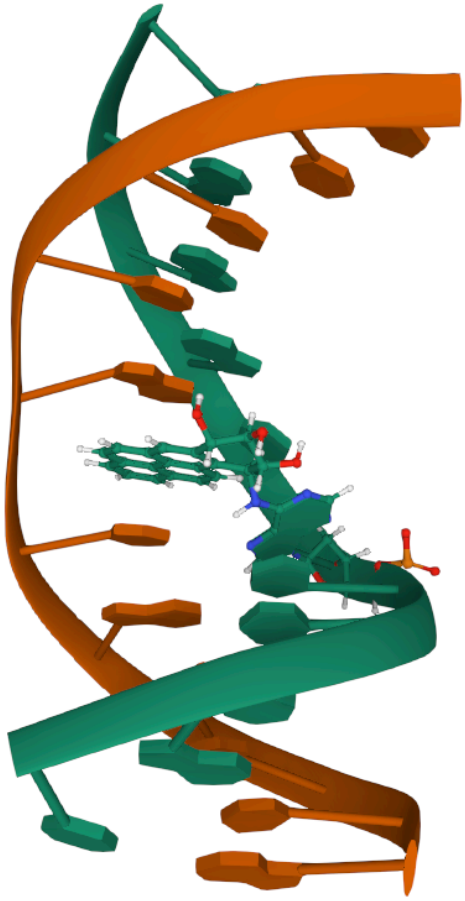
BÖLÜM 3

DNA, AMİNO ASİT VE PROTEİN

3.1. DNA

Genetik kod, canlı hücreler tarafından genetik materyalde (DNA veya mRNA dizileri) kodlanmış bilgileri proteinlere dönüştürmek için kullanılan bir kurallar dizisidir. Genetik kod, protein sentezi sırasında hangi amino asidin ekleneceğini belirleyen kodon adı verilen nükleotid üçlü dizilerini tanımlamaktadır [47].

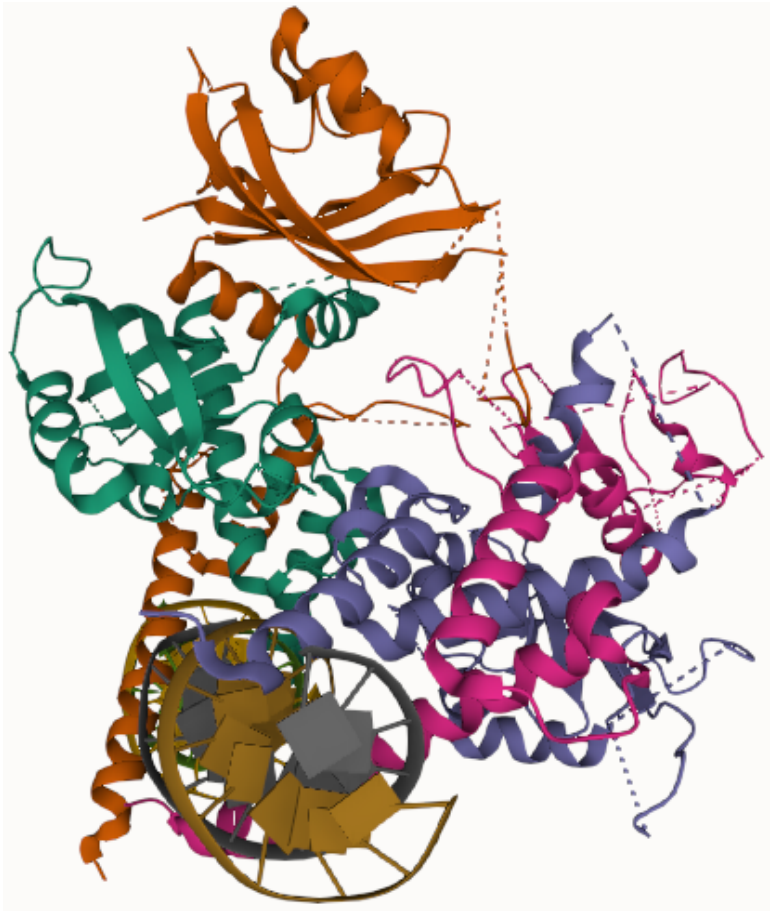
DNA veya deoksiribonükleik asit, insanlarda ve neredeyse tüm diğer organizmalarda kalıtsal materyaldir. Bir insanın vücudundaki hemen hemen her hücre aynı DNA'ya sahiptir. "DNA'daki bilgi, dört kimyasal bazdan oluşan bir kod olarak saklanmaktadır. Bu kodlar; adenin (A), guanin (G), sitozin (C) ve timin (T)'dir." Bu bazların sıralanma sırası, bir organizmanın yaratılması ve sürdürülmesi için mevcut olan bilgiyi, tıpkı alfabedeki harflerin belirli bir sırada kelimeleri oluşturmasına ve kelimelerin de cümleleri oluşturmasına benzer şekilde oluşturmaktadır. Gen, kalıtımın temel fiziksel ve işlevsel birimidir. DNA'dan oluşan genler, protein adı verilen molekülleri yapmak için talimat görevi görmektedir. İnsanlarda genler birkaç yüz DNA bazından 2 milyondan fazla baza kadar değişmektedir [48]. Örnek bir DNA 3 boyutlu yapısı Şekil 3.1'de verilmiştir. Bu 3 boyutlu temsildeki sarmal yapı karşılıklı olarak dört kimyasal baz olan A, G, C ve T'den oluşmakta ve karşılıklı olarak sıralanmaktadır. Bu sıralı zincir yapısı DNA'yı oluşturan yapıdır.



Şekil 3.1. Örnek bir DNA yapısı [49].

3.2. AMİNO ASİT VE PROTEİN

Proteinler, biyolojik sistemlerdeki birçok işlevi yerine getirmede önemli bir rol oynayan amino asitlerden oluşan makromoleküllerdir. Proteinlerin, peptitlerin ve enzimlerin en küçük yapı taşları olan amino asitler, bir ucunda karboksil, diğer ucunda amino grubu bulunan ve yan zincirlerinde nötr, polar veya iyonlaşabilir gruplar taşıyan küçük moleküllerdir. Birçok amino asit ve türevleri canlılarda metabolizmada çeşitli işlevlere sahiptir [50]. Bir protein dizisini temsil eden sarmalları ve zincirleri gösteren 3 boyutlu protein yapı örneği Şekil 3.2'de verilmiştir.



Şekil 3.2. Örnek bir protein yapısı [51].

Doğadaki tanımlanmış amino asitlerin sayısı yaklaşık olarak 300'den fazladır. Bu amino asitlerden 20 tanesi sıklıkla memelilerde ve bitki aleminde yer almaktadır. Bahsedilen bu 20 amino asit, DNA tarafından kodlanmış olan amino asitlerdir [52]. Buradan da anlaşıldığı gibi DNA amino asitleri üretmekte, amino asitler de belli kurallara göre bir dizi oluşturarak proteinleri oluşturmaktadır.

Proteinler tüm organizmalarda yaşam için gerekli olan işlevleri yerine getirmekle görevlidir. Büyüme, hücreler arası iletişim, dış etkenlere karşı tedbirler, enzim olarak katalizör görevi gibi hayati görevlere sahiptirler. Proteinlerde 20 ortak amino asit birbirlerine peptit bağı ile bağlıdır. Bu bağlı olan amino asitlerin dizilişi ile proteinlerin türleri ve kendilerine has 3 boyutlu yapıları ortaya çıkmaktadır. Her bir farklı diziliş bir motif oluşturur. Bu motifler proteinlerin türlerini ve işlevlerini belirlemede etkilidir [52].

3.2.1. Transkripsiyon Faktör Proteinleri

Transkripsiyon, bir genin DNA'sından birincil RNA transkriptinin üretilmesine kadar olan gen ifadesinin ilk adımıdır. Bu süreç, ardından gelen adımlarla beraber fonksiyonel bir proteinin üretimini sağlamaktadır. Transkripsiyon, farklı dokularda farklı proteinlerin üretimini gerçekleştirdiği için bir diğer çıktısı da özgüllük olmaktadır. Hem bazal transkripsiyon hem de düzenlenmesi, transkripsiyon faktörleri olarak bilinen spesifik protein faktörlerine bağlıdır. Bunlar, gen düzenleyici bölgelerdeki belirli DNA dizilerine bağlanmakta ve bu DNA dizilerinin transkripsiyonunu kontrol etmektedir. Moleküler biyolojide, bir transkripsiyon faktörü (veya diziye özgü DNA bağlama faktörü), belirli bir DNA dizisine bağlanarak DNA'dan mRNA'ya genetik bilginin transkripsiyonunu kontrol eden bir proteindir. Transkripsiyon faktörleri, arttırıcı ve hızlandırıcı bölgelere bağlanmaktadır. Transkripsyon faktörleri ayrıca transkripsiyonun meydana geldiği herhangi bir hücrede yalnızca küçük bir hızlandırıcı grup ile etkileşime girmektedir. Birçok transkripsiyon faktörü, belirli DNA dizilerine bağlanabilmekte ve bu trans-düzenleyici proteinler, yapısal benzerliklerine göre ailelerde gruplandırılmaktadır. Böyle bir aile içinde proteinler, DNA bağlanma bölgelerinde bir iskelet yapısını paylaşmaktadırlar ve bağlanma bölgesindeki amino asitlerdeki küçük farklılıklar, proteinlerin bağlandığı DNA dizilerini değiştirebilmektedir. Bu nedenle, diziye özgü DNA bağlayıcı proteinler olarak da adlandırılmaktadırlar. TF'lerin işlevi, hücre ve organizmanın yaşamı boyunca gen ekspresyonunu düzenlemek olarak özetlenebilir [53,54]. Transkripsiyon faktörleri, gen ekspesyonunun en önemli kısmında görev almaktadırlar. TF'ler gelişme, büyüme, hücreler arası haberleşme, çevre tepkisi gibi birçok hayati süreci DNA bağlantısı ile birlikte yönetmektedirler [55]. Bu transkripsiyon faktörleri, DNA'ya bağlanmaya aracılık etmek veya genellikle DNA bağlanması için gerekli olan faktör dimerizasyonuna neden olmak için kullandıkları kesin protein yapısı temelinde yaygın olarak ailelere sınıflandırılmaktadırlar [53].

3.2.1.1. Basic Helix-Loop-Helix Transkripsiyon Faktör Proteinleri

Basic helix-loop-helix (bHLH) proteinleri, mayadan insana kadar birçok organizmada cinsiyet belirleme, sinir sistemi ve kas gelişimi gibi birçok temel işlevi olan

transkripsiyon faktörleridir [27,56]. Aynı zamanda bitki büyümesi, gelişmesi ve stres tepkilerinde önemli bir role sahip olan bir transkripsiyon faktörü ailesidir [57]. Literatürdeki bir çalışma, NaCl ile muamele edilmiş şeker pancarı yapraklarının proteomiklerinin artmasında bHLH proteininin önemini göstermiştir. bHLH, bir döngü ile ayrılan iki alfa sarmal yapısı ve ardından 60 aa'lık bir DNA bağlanma bölgesi (temel) içermektedir. Bu sarmal yapılar, diğer bHLH proteinleri ile etkileşime izin vermektedir. Alanın 19 amino asidi mayadan insana yüksek oranda korunaktadır. bHLH'ler omurgalıların gelişiminde önemli bir rol oynamaktadır. Hücre proliferasyonu ve farklılaşmasında aktif rol oynaryan bHLH'lerin kardiyovasküler sistem oluşumunda ekspresyon düzeylerinin oldukça arttığı gözlemlenmiştir [27,58]. bHLH süper ailesinin üyeleri, 60 amino asit kalıntılı bir bölgeyi oluşturan, yüksek düzeyde korunmuş ve işlevsel olarak farklı iki alana sahiptir. E-kutusu olarak bilinen bir konsensüs heksanükleotid dizisinde transkripsiyon faktörünü DNA'ya bağlayan temel alan, bu bölgenin amino-terminal ucunda yer almaktadır. Farklı bHLH ailesi üyeleri, farklı E-box konsensüs dizilerini tanımaktadır. Karboksi-terminal ucunda bulunan HLH alanı, homo ve heterodimerik kompleksler oluşturmak için diğer protein alt birimleri ile etkileşime olanak tanımaktadır. Her biri monomerler arasında farklı bağlanma afinitelerine sahip birçok farklı dimerik yapı kombinasyonu mümkündür. Tanınan E-box dizisindeki heterojenlik ve farklı bHLH proteinleri tarafından oluşturulan dimerler, transkripsiyonel düzenleme yoluyla çeşitli gelişim fonksiyonlarını nasıl kontrol ettiklerini belirlemektedir [29].

bHLH süper ailesi, hemen hemen tüm organizmalarda ve bitki aleminde birçok hayati görevde yer almaktadır. bHLH süper ailesi üzerine yapılan ilk çalışmalar [26] bu hayati görevlerin ve bHLH TF proteinlerinin canlılardaki önemini ortaya koymuş ve ilk çalışmalardan itibaren bHLH TF proteinleri ilgiyle incelenmiştir [27]. bHLH ailesi bitkilerde üzerinde çok miktarda araştırma yapılan ve çalışılma yürütülen bir alan olarak göze çarpmaktadır. Bu çalışma alanlarına antosiyanin biyosentezi, globulin ekspresyonu, karpel, epidermal gelişim ve fitokrom sinyalleşmesi örnek olarak verilebilmektedir. *Arabidopsis thaliana*'da 118 bHLH proteini ve çeltik (*Oryza sativa*) genomunda 131 bHLH proteini tanımlanmıştır [28]. 3 bHLH geninin stoma oluşumu için organize bir şekilde çalışması ve birbirini izleyen stereo-tipik hücre bölünmeleri nedeniyle *Arabidopsis thaliana*'da stoma gelişmektedir [29]. bHLH alt ailelerinin

çoğu, *Arabidopsis thaliana*, *Oryza sativa* gibi tohumlu bitkilerde ve erken farklılaşma gösteren kara bitkilerinde görülmüştür. Bununla birlikte, çalışılan diğer bitkilerle karşılaştırıldığında, yeşil ve kırmızı algler gibi klorofitlerde bHLH protein çeşitliliğinin daha düşük olduğu bulunmuştur. Bu paragraftaki tüm bilgiler dikkate alındığında, ilk kara bitkilerinin ortaya çıkmasından kısa bir süre sonra bHLH ailesinin büyüdüğü ve kara bitkilerinde korunduğu sonucuna varılmıştır [30].

BÖLÜM 4

MATERYAL VE METOT

Bitki transkripsiyon faktör proteinleri de diğer transkripsiyon faktör proteinleri gibi bitkinin yaşamındaki hayati işlevleri yerine getirmesi açısından oldukça büyük bir öneme sahiptir. Araştırmacılar tarafından her bitkide farklı aileye sahip çok sayıda bulunan TF proteinlerinin işlevlerinin bilinmesi TF proteinleri üzerinde, dolayısıyla da bitkilerin yaşamı, tepkileri ve protein işlevleri hakkındaki önemli bilgilere ve bitkilerin neredeyse tüm özelliklerine hakimiyet sağlamaktadır. Bitki TF proteinlerinin türlerinin belirlenmesi ve sınıfladrılması için yapılan çalışmalar incelendiğinde ve literatür araştırıldığında bu bitki TF proteinlerine ait bir veri setine rastlanılmamıştır. Ayrıca yine literatür araştırıldığında, bHLH TF proteinlerine ait, içerisinde yeni nesil arama ve analiz modelleri barındıran özel bir biyoinformatik veritabanı literatürde yer almamaktadır. Bu bilgiler ışığında bu tez kapsamında yapılmış olan derin öğrenme modeli çalışması için bir bitki TF protein veri seti oluşturmak üzere Bitki Genomik Kaynağı (Phytozome) [59] veritabanı ve Bitki Transkripsiyon Faktörü Veritabanı (PlantTFDB) [60] incelenmiş ve hazırlanan Python programlama dili [61] betikleri ile otomatik olarak bir bitki TF protein veri seti oluşturulmuştur. Hazırlanmış olan yeni nesil arama ve analiz modelleri barındıran özel biyoinformatik veritabanı için ise QIAGEN CLC Genomics Workbench [62] ile bitkilerde bulunan bHLH TF proteinleri taranmış, sonrasında da bu bHLH TF proteinleri, hazırlanan Pyhton Programlama Dili [61] betikleri ile otomatik olarak bitki türlerine göre tasnif edilmiş ve veritabanının verileri oluşturulmuştur.

4.1. BİTKİ GENOMİK KAYNAĞI (PHYTOZOME) VERİTABANI

Bitki Genomik Kaynağı (Phytozome) [59] veritabanı, bitki bilimi ve biyoinformatik alanındaki topluluklar ve bireyler için, bitki genomlarına ve Joint Genome Institute (JGI) ile dizilenen seçilmiş genomlara ve veri setlerine erişmek, bu genomlar ile bu

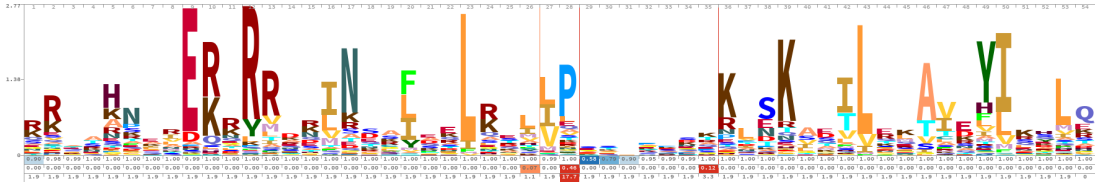
verileri görselleştirmek ve analiz etmek için oluşturulmuş olan bir merkezdir. Sürüm v12.1.6 itibariyle, Phytozome, 82 Viridiplantae türünden 93 kaynaşmış ve açıklamalı genom içermektedir [59]. Bu veritabanından elde edilen genomik bilgiden bHLH proteinlerini ayırmak için, bHLH proteinlerinin korunan alanları, QIAGEN CLC Genomics Workbench [62] yazılımı tarafından gerçekleştirilen PFAM Etki Alanı Araması ile elde edilmiştir.

4.2. BİTKİ TRANSKRİPSİYON FAKTÖRÜ VERİTABANI (PlantTFDB)

PlantTFDB, çalışmaların ve taleplerin artması nedeniyle bir grup araştırmacı tarafından geliştirilen bir merkezdir. Bitki genomu ile iletişim kurmayı, gen ailelerindeki uygulanabilir TF verilerini incelemeyi ve karşılaştırmayı ve bunlarla ilgili ek bilgiler sağlamayı amaçlamaktadır. TF'ler için kapsamlı bir veritabanı olan PlantTFDB, çeşitli açıklamalar, düzenlemeler, mutasyonlar ve fenotip verilerini içermektedir [60]. PlantTFDB'nin internet sitesi, her bir aile ve transkripsiyon faktörü için ayrı DNA ve protein dizileri ve TF listeleri içermektedir.

4.3. PROTEİN DİZİLERİNİN TEMSİLİ VE YAPISI

Protein dizileri, amino asitlerin bir araya gelerek bağ kurmaları ile oluşurmaktadır [63]. Bu dizilerdeki her bir farklı amino asit dizilimi, farklı bir proteini oluşturmaktadır. Bu dizilişler, proteinlerin 3 boyutlu yapısını tanımlayan bilgileri içermekte ve bu bilgiler ile her bir proteinin kendisine özgü yapısını meydana getirmesini sağlamaktadır [52]. Her dizi farklı bir protein oluştursa bile bu diziler içindeki belirli dizi bölümleri yani motifler o protein türünü ortaya çıkarmaktadır. İnsanların kullandıkları dillerin harflerle ifade edilmesi gibi, protein dizileri de görselleştirmeler ve araştırmalar için harflerle ifade edilmektedir. Dizilişleri açısından farklı diziler olmalarına rağmen aynı sınıfa ait motifleri içeren diziler aynı protein sınıfına aittir [64]. Şekil 4.1'de bHLH TF proteinlerinin motif yapısı gösterilmiştir. Bu motif HMM vasıtası ile elde edilerek görselleştirilmiştir [65].



Şekil 4.1. bHLH TF protein ailesi motif yapısı [66].

Şekil 4.1'de bir örneği gösterilmiş olan bu motifler, proteinlerin ailelerini tespit etmektedir. Şekil 4.1'de görselleştirilmiş olan bu motifte büyük veya küçük görünen her bir karakterin boyutu, o karakter ile ifade edilen amino asitin motifteki ilgili konumda olma olasılığını göstermektedir. Dolayısıyla bu motif protein ailesine göre farklı uzunluklara sahip olmaktadır. Ayrıca motifteki amino asitler yani karakterler her protein sınıfı için dizilik ve miktar açısından farklılık göstermektedir. Motifte her bir karakter ne kadar büyük ifade ediliyorsa, o karakter ile temsil edilen amino asitin belirtilen konumda yer alma olasılığı o kadar yüksek olmaktadır. Yine her bir karakter ne kadar küçük olarak ifade ediliyorsa da o karakter ile temsil edilen amino asitin belirtilen konumda yer alma olasılığı bir o kadar düşük olmaktadır. Bu nedenle proteinler üzerinde biyolojik deneyler yapılmadan veya yapay zekâ uygulamaları olmadan protein sınıfını tahmin etmek imkansızdır. İstatistiki veya sinir ağı temelli yapay zekâ uygulamaları tam olarak bu olasılık hesabından faydalanarak analizler yapmaktadır. Hal böyle olunca da yapay zekâ yöntemleri hem zaman hem de hız açısından kazançlı olmaktadır. Şekil 4.2, Phytozome veritabanında yer alan bir protein dizisinin protein numarası, aile bilgisi gibi bilgilerler beraber karakterlerle ifadesinin bir örneğidir.

```
>AT2G43010.1 pacid=19638516 transcript=AT2G43010.1 locus=AT2G43010 ID=AT2G43010.1.TAIR10 annot-version=TAIR10
MEHQGWSFEENYSLSTNRRSIRPQDELVELLWRDGQVVLQSQTHREQTQTQKQDHHEEALRSSTFLEDQETVSWIQYPPD
EDPFEPDDFSSHFFSTMDPLQRPTSETVKPKSSPEPPQVMVKPKACPDPPPQVMPPPKFRLTNSSSGIRETEMEQYSVTT
VGPSHCGSNPSQNDLDVSMSHDRSKNIEEKLNPNASSSSGGSSGCSFGKDIKEMASGRCITTDRKRKRINHTDESVSLSD
AIGNKSNQRSGSNRRSRAAEVHNLSERRRRDRINERMKALQELIPHCSKTDKASILDEAIDYLKSLQLQLQVMWMGSGMA
AAAASAPMMFPGVQPQQFIRQIQSPVQLPRFPVMDQSAIQNNPGLVCQNPVQNQIISDRFARYIGGFPHMQAATQMQPME
MLRFSSPAGQQSQQPSSVPTKTTDGSRLDH*
```

Şekil 4.2. Phytozome veritabanından örnek bir protein karakter temsili [59].

Şekil 4.2, Phytozome'dan edinilmiş olan bir bitki türüne ait olan fasta dosyasından bir protein dizisini ifade etmektedir. Burada ">" karakteri ile başlayıp "*" karakteri ile biten her bir bölüm bir proteinin gösterimidir. İlk satır; proteinin tanımlayıcı numarasını, konum bilgilerini ve ek bilgilerini içermektedir. Geri kalan satırlar ise

amino asitlerin karakter temsillerinin aralarındaki bağlara göre dizilmesiyle oluşan protein karakter temsilidir.

Phytozome'dan yapılan kontroller sonrası veritabanı ve derin öğrenme modelleri için veri setlerini elde etmek üzere PlantTFDB'den elde edilen veriler 3 ana yapıdadır. İlk dosyada, her bir bitki türü için TF protein tanımlayıcı numaraları bulunmaktadır. Örnek bir TF listesi dosya içeriği Şekil 4.3'te gösterilmiştir.

TF_ID	Gene_ID	Family
AT3G25730.1	AT3G25730	RAV
AT1G68840.1	AT1G68840	RAV
AT1G68840.2	AT1G68840	RAV
AT1G13260.1	AT1G13260	RAV
AT1G25560.1	AT1G25560	RAV
AT1G50680.1	AT1G50680	RAV
AT1G51120.1	AT1G51120	RAV
AT1G01010.1	AT1G01010	NAC
AT1G01260.1	AT1G01260	bHLH
AT1G01260.2	AT1G01260	bHLH
AT1G01260.3	AT1G01260	bHLH
AT1G01720.1	AT1G01720	NAC
AT1G02065.1	AT1G02065	SBP
AT1G02065.2	AT1G02065	SBP

Şekil 4.3. PlantTFDB veritabanından örnek bir TF listesi [60].

İkinci dosyada, veritabanında kullanmak üzere her bir bitki türü için bHLH dizilerinin analizinin yapıldığı CDS dizileri yani ilgili TF'lerin DNA dizileri, TF tanımlayıcı numaraları, bitki türlerinin isimleri, TF protein ailesinin kısa ismi ve aile tanımlaması yer almaktadır. Dosyadaki bilgiler "|" karakterleri ve yeni satırlar ile ayrılmaktadır. "|" karakterleri, TF numarası, kısa isim gibi tanımlayıcı bilgileri birbirinden ayırırken yeni satırlar ise CDS dizilerini tanımlayıcı bilgilerden ve diğer CDS dizilerinden ayırmaktadır. Her ">" karakteri ise yeni bir CDS dizisinin tanımlandığını dosya içerisinde ifade etmektedir. Örnek bir CDS dizisi temsili dosya içeriği Şekil 4.4'te gösterilmiştir.

```
>AT4G29930.1 Arabidopsis thaliana|bHLH|bHLH family protein
ATGGAAGATCTCGACCATGAGTACAAGAATTACTGGGAAACCACAATGTTCTTCCAGAAT
CAAGAACTCGAATTTGACAGTTGGCCGATGGAGGAAGCGTTTTCGGGTTCCGGCGAGTCG
AGTTCGCCAGACGGAGCGGCAACGTCGCCGGCTTCTTCGAAGAACGTTGTCTCCGAGAGA
AACAGACGGCAAAAGCTTAATCAGAGACTTTTTGCTCTCCGGTCAGTTGTTCCCAATATA
AGCAAGTTGGACAAGGCATCTGTCATCAAAGATTCTATCGACTATATGCAAGAACTTATT
GATCAAGAGAAGACTCTAGAAGCAGAGATCAGAGAGCTAGAATCACGGTCAACATTGCTA
GAAAATCCGGTAAGAGATTACGATTGCAATTTTGCGGAAACTCATCTGCAAGATTTCTCA
GACAATAATGACATGAGATCAAAAAAGTTTAAGCAGATGGATTACAGTACTAGAGTACAA
CACTACCCCATTGAAGTTCTCGAAATGAAAGTGACATGGATGGGAGAGAAGACGGTAGTG
GTATGCATAACATGTAGCAAGAAAAGAGAGACAATGGTGCAGCTTTGTAAAGTGTTGGAG
TCTTTGAATCTCAACATTCTCACTACTAACTTCTCTTCCTTCACCTCTCGTCTCTCCACC
ACCCTCTTCCTCCAGGCGGATGAAGAAGAAAGCAGTGCAGTAGAGGCCAAGATACAGATG
GCCATCGCAGCTTATAATGATCCAAATTGTCTTATCAACTTCTAA
```

Şekil 4.4. PlantTFDB veritabanından örnek bir CDS dizisi temsili [60].

Üçüncü dosyada ise hem veritabanında kullanmak üzere her bir bitki türü için bHLH dizilerinin analizinin yapıldığı, hem de derin öğrenme modelleri için bitki TF protein veri seti oluşturmak üzere tasnif yapıldığı PEP dizilerini yani ilgili TF'lerin protein dizileri, TF tanımlayıcı numaraları, bitki türlerinin isimleri, TF protein ailesinin kısa ismi ve aile tanımlamasını içermekedir. Dosyadaki bilgiler "|" karakterleri ve yeni satırlar ile ayrılmaktadır. "|" karakterleri, TF numarası, kısa isim gibi tanımlayıcı bilgileri birbirinden ayırırken yeni satırlar ise protein dizilerini tanımlayıcı bilgilerden ve diğer protein dizilerinden ayırmaktadır. Her ">" karakteri ise yeni bir protein dizisinin tanımlandığını dosya içerisinde ifade etmektedir. Örnek PEP dizileri temsili dosya içeriği Şekil 4.5'te gösterilmiştir.

```
>AT4G29930.1 Arabidopsis thaliana|bHLH|bHLH family protein
MEDLDHEYKNYWETTMFFQNQELEFDSWPMEEAFSGSGESSSPDGAATSPASSKNVVSER
NRRQKLNQRLFALRSVVPNISKLDKASVIKDSIDYMQELIDQEKTLEAEIRELESRSTLL
ENPVRDYDCNFAETHLQDFSDNNDMRSKKFKQMDYSTRVQHYPIEVLEMKVTWMGEKTVV
VCITCSKKRETMVQLCKVLESLNLNILTTNFSSFTSRLSTTLFLQADEEEESSAVEAKIQM
AIAAYNDPNCLINF
>AT5G51780.1 Arabidopsis thaliana|bHLH|bHLH family protein
MEKMMHRETERQRRQEMASLYASLRSLLPLHFIKGKRSTSDQVNEAVNYIKYLQRKIKEL
SVRRDDLMVLSRGSLLGSSNGDFKEDVEMISGKNHVVVRQCLVGVEIMLSSRCCGGQPRF
SSVLQVLSEYGLCLLNSISSIVDDRLVYTIQAEVNDMALMIDLAELEKRLIRMK
>AT3G55370.3 Arabidopsis thaliana|Dof|OBF-binding protein 3
MVFSSLPVNQFDSQNWQQMISILVFFSTSRLFKKLFLVDKNLFSCLLQGLMYNVFLTGLI
FSLQGNQHQLECVTTDQNPNNYLRQLSSPPTSQVAGSSQARVNSMVERARIAKVPLPEAA
LNCPRCDSTNTKFCYFNNYSLTQPRHFCKTCRRYWTRGGSLRNVPVGGGFRRNKRSKSRS
KSTVVVSTDNTTSTSSLTSRPSYSNPSKFHSYGQIPEFNSNLPILPPLQSLGDYNSSNTG
LDFGGTQISNMISGMSSSGGILDAWRIPPSQQAQQFPFLINTTGLVQSSNALYPLLEGGV
SATQTRNVKAEENDQDRGRDGDGVNNLSRNFLGNININSGRNEEYTSWGGNSSWTGFTSN
NSTGHLSF
```

Şekil 4.5. PlantTFDB veritabanından örnek PEP dizileri temsili [60].

Tüm bu dosyalar ve içindeki veriler hem TF ailelerine hem de bitki türlerine göre dağınık şekilde yer almaktadır. Hem veritabanı tasarımı hem de derin öğrenme modelleri için oluşturulacak bitki TF protein veri seti için hem diziler satır bazında bir formata sokulmalı, hem de gerekli ayırma ve birleştirme işlemleri ile tasnif tamamlanmalıdır.

4.4. VERİLERİN ELDE EDİLMESİ, UYGUN DOSYA FORMATINA DÖNÜŞTÜRÜLMESİ VE TASNİFİ

Phytozome kontrolü için Python programlama dili kullanılarak hazırlanan betik ile her bir proteinin bilgisi ve dizisi fasta dosyaları dikkate alınarak tek satırda ve belirli bir formatta düzenlenmiştir. Daha sonra, CLC Genomics'in sonuçlarına göre, her fasta dosyasından bHLH proteinleri ayrıştırılmış ve farklı bir dosyaya kaydedilmiştir. İçerisinde TF proteinlerinin DNA dizilerini (CDS sequences) ve amino asit dizilerini (PEP sequences) barındıran veritabanından bilgiler birer fasta dosyası olarak indirilmiş ve dosya içerikleri hazırlanan Python programlama dili ile hazırlanan betik bitki türlerine göre tasnif edilmiş olan verileri otomatik olarak birer sekmelerle ayrılmış metin (tab delimited text-txt) dosyalarına aktarılmıştır. Bu dosyalar içerisinde bitkilerin bHLH TF proteinleri ile alakalı TF ID, GENE ID, NCBI ID, CDS dizisi, PEP dizisi ve protein ailesi bilgileri bulunmaktadır. Burada TF ID, dizilerin PlantTFDB'deki TF protein numaralarını, GENE ID, dizilerin gen ailesine ait numarayı, NCBI ID, dizilerin Ulusal Biyoteknoloji Bilgi Merkezi (National Center for Biotechnology Information-NCBI) [67] içerisindeki numarasını içermektedir. Bu işlemler, her bir protein ailesi için PlantTFDB'den alınan ekstrakt ve her aile içindeki CDS ve PEP dizileri için ayrı ayrı çalışmış ve veri setinin hazırlığını sağlamıştır.

63 bitki familyasının her biri için protein dizilerinin dosyaları PlantTFDB'den Bölüm 4.3'te bahsedilen 3 ayrı dosyada fasta formatında indirilmişir. İndirilen dosyalarda, çok satırlı dizilerin her biri, dizilerin analizi ve derin öğrenme modelinin eğitimi için kullanılacak olan bitki TF protein veri seti için otomatik olarak tek bir satırda ifade edilmiş, yalnızca dizi ve protein ailesi bilgileri alınmış, diğer bilgiler ve ayırıcı semboller ayıklanmış ve tüm yalın bilgiler iki sütunda sekmeyle ayrılmış tek bir metin dosyasında toplanmıştır. Bu süreçte bir Python (sürüm 3.8) betiği ve manuel

kopyalama kullanılmıştır. Bu işlemler sonucunda bitki genetik kaynağından 132330 satırda 58 farklı protein transkripsiyon faktörü sınıfının dizi ve sınıf bilgileri elde edilmiştir. Dosyada bir sütun protein dizilerine, diğer sütun da her satırdaki dizilere karşılık gelen protein ailesine atanarak tek bir dosya oluşturulmuştur. Bu dosya ile beraber bitki TF protein veri seti bu çalışmada gelecek çalışmalarda ve diğer protein dizileri sınıflandırma çalışmalarında kullanılmak üzere hazırlanmış ve literatüre kazandırılmıştır. Hazırlanan bitki TF protein veri setinden örnek bir kısım Şekil 4.6'da gösterilmiştir.

```
family  sequence
bHLH    MTEHKRRPASLEPAVSLSCGTRQRYKTEVPISRKEKKEIMGERVATLQQLVSPFDKTDTASVLFEAMEYIKFLHDQVKVLSAPYLLSASTRE
MQVDYLQFQLLEERKLSGISAV
C3H     MGGEEEDEEAALAVALGFPPYRRSLLKSKTYEAFVKISSLFDQISPKSGKETQDKERDFISENSVEKELVNCKAITSDDLGQREQQILDISN
RTSLSEEREGGGGSGREEREKGEDLIKDLDVEEGEISDDTEEINVSDDQEHCSVDIAERDLEEGQICGDFIEKEEFPGTCTEENRTHEKDILSTSSIGEM
SNLIDDLGLNVVKDKKFEELSSENLICGSSQLDRRKNNGVITRDTLGHMGAASERVMADIEGNKASVKDFEVGVKRKRVLTKERKERKKAKRRIKRVQKD
RQEGVKRLNLRSNVTEKPKPKLPCKHYLKGKCQQGDSCKFSHDCIPLTKSEPCRFFACNKCLKGDDCPYDHELFKYQCDSYKSKGFCPRGDTCLFSHKML
LVRPESDANKKAQENFSLQPLKGSTSQHTLSMSKNMTLRTPASVGSITRHPGQKAVEMKPAAQAPKGISFLLFDKDPSDESSKQQKSNLPPIGDNSVSQK
IHNSNEILVRTPTSTPLYQSKMTTEGPDKNINSAQKAPTSQFVRTMDQSAQSSVSHGNFPTVSSILQEFLFNGH
GRAS    MIQSLLPRSPITMKKTKRLNRDESQQVQEQELIVVEDYQRSKKRTSFEPETVLVDEVEVENQVEEIETIVSSSSSSSRRNEFSHPCESNGLR
LLGLLLQCAEAVAMDNLNKATNLLPEISEISSPFGSSIERVAAYFIEALRSRIITSFLGLFHILHSRSKKIQSIRITGVGSSFELLEATGRRLTDFASSF
GLPFEFIPLEGKIGNIIDLSQLAPRQNEATVVHWMHHCLYDITGSDLGTLKLLKVLKPKLITIVEQDLSNGGNFLGRFVEALHYYSALFDALGDGQEWDS
MERYQVEQQIFGCEIKNIVAVGGPKRTGEVKVDRWGDELRQIGFNPVSLGGNPAAQASLLLGMFPWKGYTLVEENGCLKLGWKDLSLLTASAWQPSD
FAR1    MAEEMDIYGGMFEIENIGEVEKDEVADDNIPVELNEINEPGVGMIFSSFHEAKSFYDIYAQSKGFSIRTRSTYRDRRGPDLSSALFVCTCEG
FNQRMPKSDDEGKIRRTTSIVRSGCKASMRVTNIRGTTAWKVTVFSDQHNHEFIPQNKGDLVKCNKRKSRAAKTPTEAMRSCDVYRQATRLAHMAGRSEE
IYEVIMAVMEETFKKVSQMEKELFNPENDKRCLDNGYDNSNNLGNSDQLIPAEPHISRTNLKKRDRKGKEMENNINILDTGLTKWQPLNHEEDISVVS
C2H2    MPVANLNSYNMLDVMRPDQRVGKLEEGNDSLDTFIRQAIGKEPVLSFSRAGDNPVQWIQLLQALDHQDLPGWPLLAPPKVQMQKCEKCSREF
CSPINYRRHKRVHRRALKVDKDFPKNRDLLGAFWDKLSLDDAKEIVSFKNITLEDVPGSSIVKTLTSLIRKPGFFPLPQVYIRAGSALLDVVQARPSGFP
LSSQELFNILDDASEKTFLCAGTALSLQKFVFDGEAGKIGLEMKNIVACTCFLLEQNLVNAWLADKDAEALRCHKLLVEEEEAAQRRQADLLERRRLKKL
RKESLRVKEQIDTEEADLEENSSDSLPDLTSIEASSPPEASNSDLYTLEEPSDLEPLHPMNFNKEADASCFSGLDNVQIDTASSNADLNNFRNVDHRLKH
GNMRRNHNVARRPSPSSRRGASNGFHSGVVVPKLGPVQRQGVHRDQKSTLLNGHKIWTRKTKPDNEGEDLSVGLLGESRDLNDRDSIDHSDVDESCEVLI
GSISVTLRECNAQSQGAMLTSVVDQCAADNQLPQNKCLQKKHAKPDSGQCGVNRSVKLWRPVGKHEAAGALTLKDGEKDTVMDMHVNMADQASTIEMSPL
VEECADPRGLKLFCSSSAEAFLTQRWKEAVAADHVKLVLSSESNSPPNPNILCSSENRLSRVGPLEPSNHGATKLKSKDKSVKNGRTKYIPKQIKST
```

Şekil 4.6. Bitki TF protein veri setinden bir bölüm.

Ayrıca bHLH TF protein ailesi için hazırlanan yeni nesil arama ve analiz modelleri barındıran özel biyoinformatik veritabanı için de CDS ve PEP dizilerini barındıran dosyalar, içerisindeki tanımlayıcı numaralar ve bitki tür isimleri ile beraber bir Python betiği vasıtasıyla tasnif edilmiş ve NCBI'dan elde edilen protein dizileri bilgilerinden NCBI tanımlayıcı numaraları yine bir Python betiği ile otomatik olarak çıkarılarak tasnif edilen dosyaya eklenmiştir.

4.5. HAZIRLANAN VERİ SETİ

Bu çalışma kapsamında PlantTFDB'den elde edilmiş olan genel bitki TF proteinlerinin ve bu TF proteinlerinin aile bilgilerinin yer aldığı dosyalardan Bölüm 4.4'te detayları verildiği gibi yapılan işlemlerle bitki TF protein veri seti elde edilmiştir. Bu veri seti

içerisinde uzunlukları yaklaşık olarak 1200 amino asite kadar uzanabilen çeşitli uzunluklarda protein dizileri ve her bir dizinin mensup olduğu bitki TF protein ailelerinin isimleri yer almaktadır. Bu diziler, veritabanından indirilen ham dosyalardan Python programlama dili betikleri ile toplanmış ve diziler ve aileler (sınıflar) olmak üzere iki sütun olarak hazırlanmış olan esas veri seti dosyasına her bir dizi tek bir satırda ifade edilmek üzere aktarılmıştır. Bu dosya, tüm derin öğrenme modellerinde ve sonraki çalışmalarda kullanılmak üzere sekmeyle ayrılmış metin (tab delimited text) türüyle kaydedilmiştir. Dizilerin uzunluk dağılımı Şekil 4.8'deki grafikte verilmiştir. Ayrıca veri setindeki her bir sınıfın dağılımı Çizelge 4.1'de verilmiştir.

Çizelge 4.1. Dizilerin sınıflarına göre sayıları.

TF Ailesi	Dizi Sayısı	TF Ailesi	Dizi Sayısı	TF Ailesi	Dizi Sayısı	TF Ailesi	Dizi Sayısı
M-type_MADS	2950	HD-ZIP	3492	CPP	721	E2F/DP	804
bHLH	12061	SRS	537	GeBP	615	SAP	60
C3H	4056	Trihelix	2752	BBR-BPC	527	BES1	648
GRAS	3729	TALE	1881	NZZ/SPL	41	CAMTA	569
FAR1	3136	WRKY	5713	TCP	1678	CO-like	920
C2H2	7110	LSD	399	GRF	760	LFY	122
ERF	8285	G2-like	4208	HB-other	894	STAT	102
GATA	2264	DBB	735	ARF	2057	NF-X1	172
MYB_related	6245	Dof	2301	YABBY	783	HB-PHD	253
HSF	1788	bZIP	6711	ZF-HD	1040	EIL	462
MYB	9008	NAC	7938	NF-YC	1003	RAV	270
B3	4698	Nin-like	1175	WOX	930	HRT-like	122
AP2	1836	NF-YB	1302	Whirly	212	VOZ	277
NF-YA	1103	LBD	2778	S1Fa-like	136		
SBP	1816	MIKC_MADS	3090	ARR-B	1055		

4.6. ÇALIŞMADA YAPILAN VERİ ÖN İŞLEMLERİ

Dizilerin Şekil 4.5'te de bir örneğinin gösterildiği gibi "MEDLDHEYKNY..." şeklinde devam eden yapısı, biyoinformatik araştırmacıları için belirli bir netlik düzeyine sahip olsa da bilgisayarlar ve dolayısıyla bu çalışma ve diğer benzer birçok çalışmada kullanılan istatistiksel modeller ve diğer yapay zekâ modelleri için tam olarak anlaşılamamaktadır. Şekil 4.6'da bir bölümü gösterilmiş olan elde edilen ham

diziler, eğitim için derin öğrenme modellerine verilmeden önce ön işleme tabi tutularak belirli bir formata getirilmelidir. Görüntüler üzerinde yapılan çalışmalarda bu işlemler bir dizi boyutlandırma ve gürültü azaltma işlemi iken, metin tabanlı verilerde daha farklı ön işleme adımları uygulanmaktadır. Doğal dil işleme çalışmalarında kullanılan ön işleme yöntemlerine benzer şekilde, bu çalışmada olduğu gibi metin verileri üzerinde yapılan çalışmalarda da benzer yöntemler kullanılmaktadır [64]. Bu çalışmada dizilerin boyut sınırlaması, boşlukların doldurulması, dizilerden gömmelerin (embedding) hazırlanması gibi bir dizi işlem veri kümesine uygulanmıştır.

4.6.1. Dizilerin Kod Sözlüğü ile Temsili

Doğada tüm organizmalarda ortak olan 20 amino asit vardır. Bu amino asitler de önceki bölümlerde bahsedildiği gibi harfler ile ifade edilir [63]. Bu harfler A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W ve Y'dir. Bu 20 amino asit dışında bazı nadir durumlar olabilmektedir. Bu durumlar da diziler içinde diğer harflerle veya harf gruplarıyla belirtilmektedir ancak proteinlerin sınıflandırılmasında diziler içinde çok düşük bir olasılıkta ve miktarda bulunma oranı olduğundan ve bu nadir durumlar proteinin motifini ve ailesini doğrudan etkilemediğinden protein sınıflandırma çalışmalarında göz ardı edilebilmektedir [20].

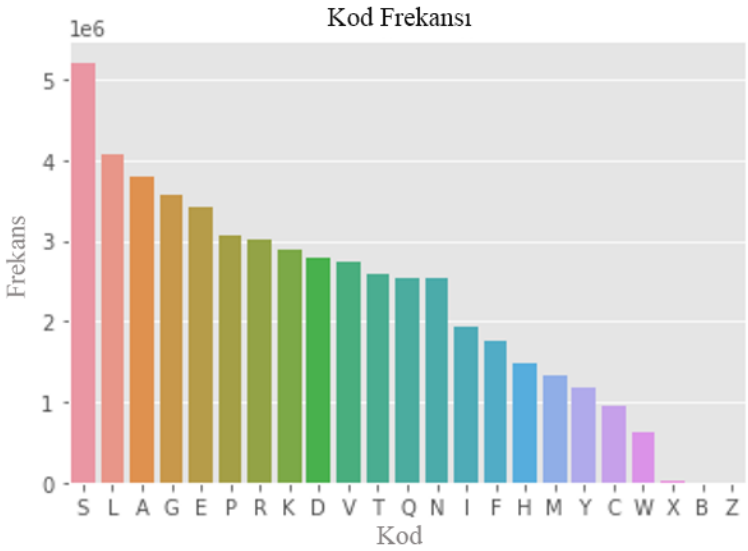
Amino asitlere karşılık gelen bu harfler bilgisayarlarda temsil edilebilseler de yapay zekâ modelleri ve daha özele inilecek olursa derin öğrenme modelleri ve bu modellerin fonksiyonları sayısal veriler ile çalıştığından, bu dizilerin, dolayısıyla da dizileri oluşturan amino asitlerin sayısal olarak temsil edilmesi gerekir. Modellerde kullanılan sayısallaştırma yöntemlerinin ilki kod sözlüğü yani harf-sayı dönüşümüdür. Bu dönüşümde, modelde kullanılacak olan dizilerdeki amino asitlerin, yani harflerin her birine 1'den 20'ye kadar birer sayı atanmıştır. Harici kalan ve göz ardı edilebilecek olan nadir durumların her birine de 0 rakamı ataması yapılmıştır. Bu işlem için bir Python fonksiyonu hazırlanmış ve bu fonksiyon ile program içerisinde tanımlanan her bir amino asite karşılık gelen sayısal değer bir değişkene atanmış ve fonksiyonun içerisinde yer alan döngünün her bir turunda bu değişken dönüştürülmüş dizilerin tutulduğu listeye eklenmiş ve sayısallaştırılmış veri seti oluşturulmuştur. Sıklıkla

kullanılan bu yönteme Bileshi ve arkadaşlarının çalışmasındaki ön işleme kısmının adımlarından biri verilebilir [20]. Anlatılan kod sözlüğü dönüşümünün tablosu Çizelge 4.2'de verilmiştir.

Çizelge 4.2. Kod sözlüğü ile harf-sayı dönüşümü.

Harf	A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y
Sayı	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

Çizelge 4.2'deki yöntem ile sayısallaştırılan diziler, yapılacak diğer ön işleme adımlarından sonra derin öğrenme modeli için kullanılacak olan Python programlama dilinin Keras kütüphanesinin gömme katmanına girdi olarak verilebilir. Veri setindeki amino asitlerin dağılım frekansı Şekil 4.7'de gösterilmiştir.

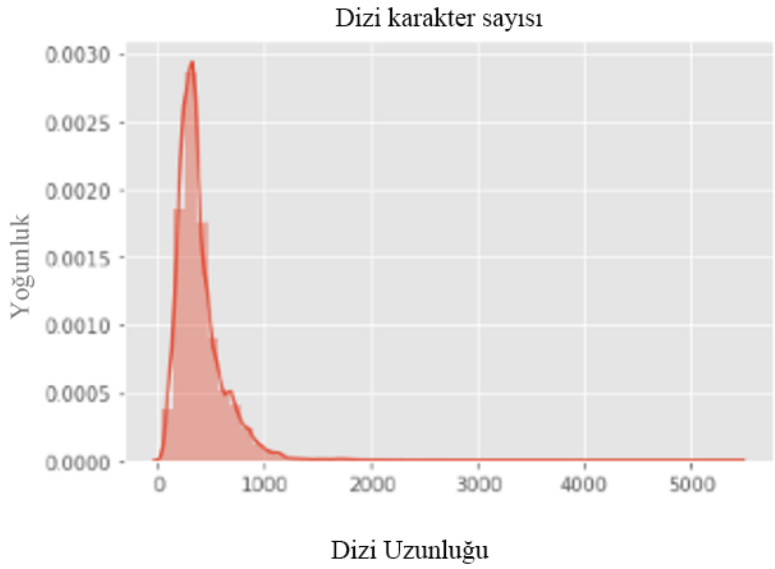


Şekil 4.7. Amino asit harf dağılım grafiği.

Şekil 4.7'deki amino asitlerin diziler içerisindeki yoğunluğunu gösteren grafik incelendiğinde Çizelge 4.2'de verilen ve tüm organizmalarda ortak bulunan 20 amino asitin miktarları oldukça fazla iken, bu 20 amino asit dışında olan ve nadir olarak bulunan özel durumların gerçekten de yok sayılabilecek kadar az olduğu rahatlıkla görülebilmektedir.

Sayısallaştırılan dizilerin uzunluklarına bakıldığında neredeyse tüm dizilerin uzunluklarının farklı olduğu açıkça görülmektedir. Bazı diziler kısa, bazı diziler ise

oldukça uzun olabilir. Derin öğrenme çalışmalarında girdi olarak alınan verilerin uzunluklarının farklı olması başarıyı olumsuz yönde etkilemektedir. Bu sebepten de dizilerin aynı uzunlukta olması başarıyı artıran bir etken olacaktır. Dizilerin tespit edilen sabit bir uzunluğa getirilmesi, dizilerde olası veri kayıplarının olma ihtimalinden ötürü akla dizinin sınıflandırılmasında doğabilecek başarısızlıkları getirse de daha önceki bölümlerde de belirtildiği gibi bir protein dizisinin sınıfını, amino asitlein farklı bağlarının oluşturduğu motifler belirlediğinden ötürü herhangi bir başarı kaybı olmayacaktır. Çünkü bu motifler, dizinin sadece belli bir kısmında yer almaktadır. Derin öğrenme modellerinin doğru çalışabilmesi için dizilerin uzunluklarına göre miktarları incelenerek belirli bir sabit uzunluk belirlenmelidir. Bunun için önce uzunluklarına göre dizilerin sayısı kontrol edilmiştir. Uzunluklarına göre dizi sayılarını gösteren grafik Şekil 4.8'de verilmiştir.



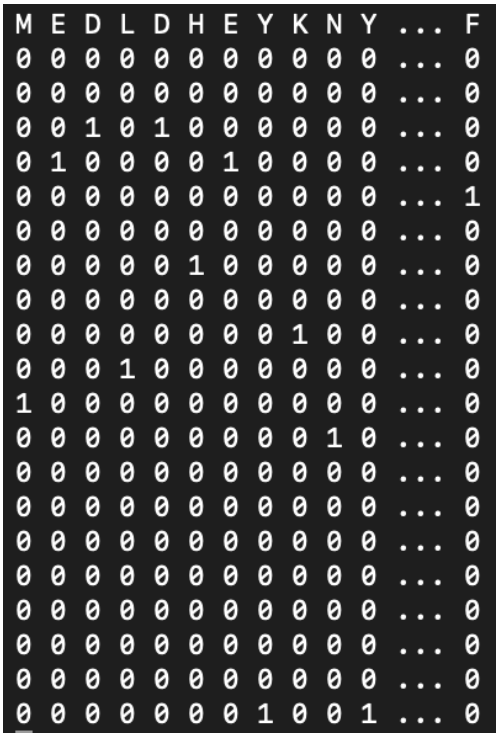
Şekil 4.8. Uzunluklarına göre dizilerin yoğunluk grafiği.

Şekil 4.8'deki grafik incelendiğinde dizilerin çoğunlukla 300 ila 500 karakter uzunluğunda olduğu görülmektedir. Daha kısa diziler daha az sayıda olduğu gibi yaklaşık 1200 karakterden fazla olan diziler ihmal edilebilecek kadar az sayıdadır. Bu sebeplerden ötürü dizi uzunlukları sırası ile 305, 450 ve 525 seçilerek hazırlanan modellerde denemeler yapılmıştır. Yapılan denemelerin sonucunda eğitim süresi ve eğitim hızının yanı sıra dizilerin uzunluklara göre dağılımını da göz önünde bulundurarak kod sözlüğü ön işleme yöntemini kullanan modellerde dizilerin sabit

uzunluğunun 450 karakter olarak seçilmesine karar verilmiştir. Belirlenen bu maksimum uzunluk ile birlikte maksimum uzunluk olan 450 karakterden daha uzun olan diziler son kısmından kırpılarak kısaltılmış, daha kısa olan diziler ise son kısmına 0 rakamı eklenmesi suretiyle doldurularak 450 uzunluğa çıkarılmıştır. Bu sayede tüm diziler eşit uzunluğa getirilmiştir [20]. Bu işlem için Python programlama dilinin Keras kütüphanesideki pad_sequences fonksiyonu kullanılmıştır. Bu işlem yapılırken bir Python fonksiyonu hazırlanmış, bu fonksiyon içerisinde bir önceki adımda sayısallaştırılmış dizilerin yer aldığı liste değişkeni üzerinde işlem yapılmış, bu liste içerisinde yer alan her bir dizi kısaltma/doldurma işlemine tabi tutulmuş ve fonksiyonun içerisinde yer alan döngünün her bir turunda işlemden geçen bir dizi yeni oluşturulan ve sabit uzunlukta dizileri içinde bulunduran listeye yerleştirilmiştir.

4.6.2. Dizilerin Tek-Sıcak Kodlama ile Temsili

Diğer bir dizi sayısallaştırma yöntemi, tek sıcak kodlama (one-hot encoding) yöntemidir. Bu yöntemde veriler ikili bir sistemde ifade edilmektedir. Kategoriler, kullanılacak kategori sayısı büyüklüğünde bir vektör olarak uygulanır. Her kategori için vektörün bir indeksi 1 rakamı, diğerleri 0 rakamı olarak uygulanır. Bu yöntem, Python programlama dilinin scikit-learn kütüphanesi ile uygulanabilir. Bu işlem, sınıf sayısının fazla olmadığı çalışmalarda uygulanması görece kolay bir yöntemdir. Ancak sınıf sayısı arttıkça vektör boyutu artar ve hesaplama maliyeti artabilir. Ayrıca doğal dil işleme çalışmaları gibi çalışmalarda çokça karşılaşılan devam eden dizilerde her karakterin ve kelimenin önceki ve sonraki karakterlerle ve kelimelerle ilişkisini tanımlamak kolay değildir. Bu sebepten de teknik olarak uygulaması kolay görünse dahi uygulamada eksik yönleri vardır. Yang ve arkadaşlarının yaptığı çalışma bu yönteme örnek olarak verilebilir [68]. Şekil 4.9'da bir tek-sıcak kodlama ile dizi temsili örneği verilmiştir.



Şekil 4.9. Dizilerin tek-sıcak kodlama ile temsili.

Şekil 4.9'da da görüldüğü her bir sütunda sadece bir satır 1 ile, diğerleri 0 ile ifade edildiğinden, bu yöntemdeki vektör gösterimi içerisinde çokça boş tanımlama vardır ve bellekte gereksiz yer kaplamaktadır. Ayrıca yine şekilde görüldüğü gibi her bir amino asitin arasındaki ilişki, bu amino asitlerin birbiri ile yakınlıkları ve birbirlerinin önünde ve/veya sonunda bulunma olasılıkları tespit edilememektedir.

Bu çalışma kapsamında eşit uzunluğa (450 karakter) getirilmiş diziler tek-sıcak kodlama yöntemiyle vektörlere dönüştürülmüştür. Bu işlem gerçekleştirilirken bir Python fonksiyonu hazırlanmış, bu fonksiyon içerisinde bir önceki adımda sayısallaştırılmış dizilerin yer aldığı liste değişkeni üzerinde işlem yapılmış, bu liste içerisinde yer alan her bir dizi kısaltma/doldurma işlemine tabi tutulmuş ve fonksiyonun içerisinde yer alan döngünün her bir turunda işlemden geçen bir dizi yeni oluşturulan ve sabit uzunlukta dizileri içinde barındıran listeye yerleştirilmiştir. Ayrıca hazırlanan modellerde bu yöntemi veri setindeki dizilerin sınıfları için kullanılmış ve 58 sınıf tek-sıcak kodlama ile temsil edilmiştir. Sınıflar için yapılan işlemde de benzer şekilde tek-sıcak kodlama uygulaması yapılmıştır. Dizilerdeki başarısını görmek için de bu çalışmadaki veri setine benzer bir veri seti ile yapılmış bir çalışmadaki ResNet

[20] ağını test etmek için yine bu yöntem kullanılmış ve bu çalışmadaki veri seti de bahsedilen ResNet modeli ile kıyaslama amaçlı çalıştırılmıştır.

4.6.3. Dizilerin k-mer'ler ve Word2Vec Gömmeleri ile Temsili

Gerçek hayattaki dillerde belli cümle kalıpları ve bir arada kullanımı sık olan kelimeler vardır. Bir kelimenin öncesindeki ve sonrasındaki kelimelerin olasılıkları ve dolayısıyla da kelimelerin tahminleri, doğal dil işleme çalışmalarının başarılarını artıracaktır. Nitekim önceki iki kısımda bahsedilen ön işleme teknikleri bu desteği sağlamamaktadır. Kaynak veya hedef kelimeden önceki ve sonraki kelimeler arasındaki ilişki, tıpkı doğal dil işleme çalışmalarında olduğu gibi [69] doğru sınıflandırma için dizilerdeki protein özelliklerini ve sınıfını gösteren motiflerin kaçırılmaması için çok önemlidir. Doğal dil işleme çalışmalarında kelimeler ve aralarındaki ilişki üzerinde çalışıldığı gibi protein dizileri ile yapılan çalışmalarda da benzer yapıda kelimelere ihtiyaç vardır. Bu sebepten de boşluksuz, tek parça olan protein dizileri kelimelere (k-mer) bölünmelidir. Literatürde daha önce benzer birkaç çalışma yapılmış olup, bu çalışmalarda oluşturulan k-merlerin karakter sayıları geneli itibariyle 3 ila 6 karakter arasında değişmektedir [70,71].

Dizileri daha başarılı bir şekilde sınıflandırmak için bu k-merlerin birbirine yakınlıklarının bilinmesi gerekmektedir. Doğal dil işleme çalışmalarında daha önce öğrenilen kelimeler ile oluşturulan sözlük yapıları gibi bu çalışma kapsamında literatüre kazandırılmış olan bitki TF protein veri setinden benzer bir kelime dağarcığı oluşturmak var olan başarıyı daha da artıracaktır. Ayrıca oluşturulacak böyle bir sözlük, gelecek devam çalışmalarında ve yapılacak olan diğer benzer protein sınıflandırıcı çalışmalarda da aktif olarak kullanılabilecektir. Bunun için veri setindeki diziler üçer karakterli kelimelere (3-mer) bölünerek bir kelime hazinesi oluşturulmalıdır. Ne kadar çok kelime olursa o kadar çok amino asitin ve protein parçasının (k-mer) arasındaki ilişki o kadar yüksek başarıyla tespit edilebileceğinden, daha fazla kelime elde etmek ve olası üçlü kelime kombinasyonlarını kaçırmamak için her bir diziyi üç kez birer karakter kaydırarak her diziden üç ayrı 3-mer grubu oluşturmak daha başarılı bir kelime haznesi sağlayacaktır [17]. Bu sayede neredeyse

hiçbir kombinasyon gözden kaçmayacak ve başarı da aynı nispette artacaktır. Bu metot ile oluşturulmuş bir k-mer örneği Şekil 4.10'da verilmiştir.

Şekil 4.10. Dizilerden k-mer'lerin hazırlanması.

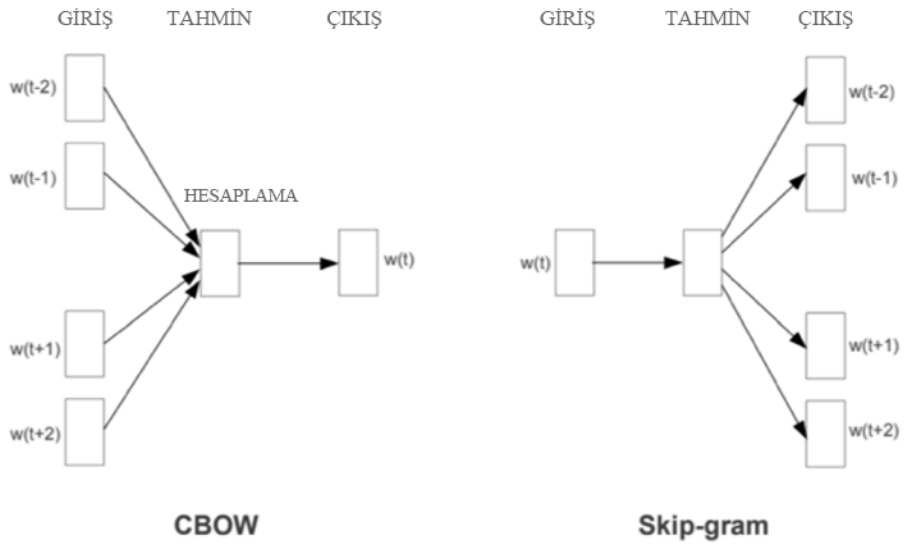
Şekil 4.10'da bir örneği gösterildiği gibi 3 karakterli kelimeler yerine 4, 5 veya 6 karakterli kelimeler oluşturulduğunda her diziden elde edilecek kelime sayısı, diziler kelime sayısı kadar kat çoğaldığından fazlaca artacaktır. Bu husus da daha uzun bir çalışma süresini ve daha fazla kaynak kullanımını beraberinde getirecektir. Tabii bu husus modellerin eğitim sürelerini de olumsuz etkileyecektir. Ayrıca her kelimenin eğitim veri setinde temsil sayısı daha az olacağından ve kelimelerin uzun olmasından ötürü bazı motifler kaçırılabileceğinden benzerlik değerlerinin etkisi daha az olacaktır. Bu husus da tabiatıyla başarıyı olumsuz yönde etkilemektedir.

Diziler kelimelere bölündüğü için dizilerin sabit uzunluklarının önceki metotların aksine 250 kelime ile sınırlandırılması yeterli olacaktır. Nitekim 200, 250 ve 300 kelime uzunlukları ile yapılan ön testler de tercihi doğrulamaktadır. Daha fazla sayıda kelime içeren diziler kullanılmak isteniyorsa, çok sayıda kısa dizinin sonu çok fazla 0 ile doldurulacak ve gereksiz bir işlem maliyeti olacaktır. Çok az sayıda kelime seçildiğinde de tam aksi olarak diziler çok kısalacak ve bu sebepten de proteinlerin sınıflarını, yani ailelerini belirleyen farklı bağların temsili olan motiflerde kayıp ve fazla kırpılma olabilecektir. Bu durumun da beraberinde başarıdan kaybı getirmesi kaçınılmazdır.

4.6.3.1. Word2Vec Modeli

Veri setindeki tüm diziler yukarıda belirtildiği şekilde kelimelere dönüştürüldükten sonra Word2Vec ile hazırlanan kelimelerin vektör gösterimi ve kelime dağarcığı

tamamlanmalıdır. Bu işlem için kullanılacak olan Word2Vec, vektör uzayında kelimeleri temsil eden denetimsiz, tahmin temelli bir modeldir. Word2Vec ile kelimeler bir vektör uzayında temsil edilirken benzerliğe göre puan verilir ve kelimelerin ağırlıkları oluşturulur. Bu ağırlıklar ile kelimelerin yakınlığı ve birbirlerinin önünde veya sonunda gelme olasılıkları belirlenir ve bu sayede gelecek veya önceki kelimelerin tahmini yapılır. Word2Vec modelinin iki farklı mimarisi vardır. Bu mimariler Sürekli Kelime Torbası (Continuous Bag-of-Words-CBOW) ve Sürekli Atlama-Gram (Continuous Skip-Gram) mimarileridir. Bu mimarilerden ilki olan CBOW mimarisinde (Şekil 4.11'in sol kısmı) tüm kelimeler aynı konumda yansıtılmaktadır, yani vektörlerin ortalaması alınmaktadır. CBOW, belirtilen pencere boyutu kadar önceki ve sonraki kelimeleri analiz ederek merkezdeki kelimeleri tahmin etme yapısı üzerinde çalışmaktadır. Dolayısıyla mevcut kelime, bağlama göre tahmin edilmiş olur. Skip-Gram mimarisi (Şekil 4.11'in sağ kısmı) de CBOW mimarisine benzer fakat Skip-Gram mimarisinde CBOW'dan farklı olarak mevcut kelimenin bağlama göre tahmin edilmesinin yerine, aynı cümledeki başka bir kelimeye göre bir kelimenin sınıflandırmasının en üst düzeye çıkarılması amaçlanır. Yani log-lineer sınıflandırıcıya girdi olarak mevcut her kelime kullanılır ve mevcut kelimeden önce ve sonra belirli bir aralıktaki kelimelerin tahmini yapılır. Skip-Gram mimarisinde geneli itibariyle ortadaki kelime girdi olarak alınır ve pencere boyutu kadar önceki ve sonraki kelimeler tahmin edilir. Skip-Gram'da, bağlama göre kelime tahmini yerine cümledeki diğer kelimelere dayalı kelime sınıflandırmasının en üst düzeye çıkarılması amaçlanır [72]. Aşağıda Şekil 4.11'de Word2Vec modelinin CBOW ve Skip-Gram mimarilerinin çalışma yapılarını gösteren şema verilmiştir. Bu şemada sol kısımda CBOW mimarisinin yapısı, sağ kısımda ise Skip-Gram mimarisinin yapısı açıklanmaktadır.

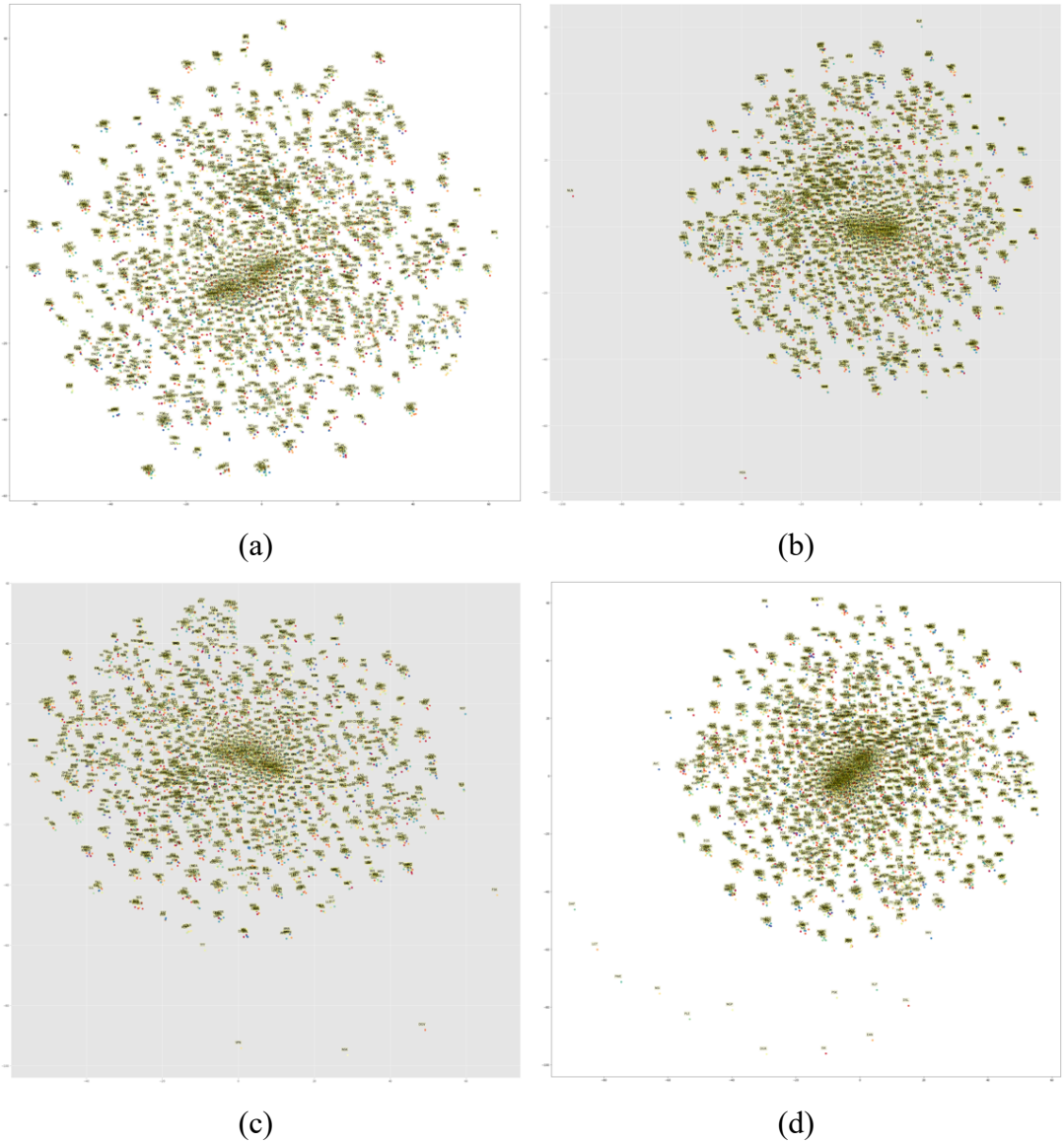


Şekil 4.11. Word2Vec'in CBOW ve Skip-Gram mimarilerinin yapısı [72].

Bu çalışmada yapısal tahmin kolaylığı, hız ve kaynak kullanımı açısından yapılan denemeler sonucunda CBOW mimarisi tercih edilmiştir. Modelde vektör boyutu 300 olarak ayarlanmıştır. Word2Vec modelinin pencere boyutu 4, 5, 7 ve 10 olarak seçilmiş, model bu boyutların her biri ile hazırlanarak incelenmiştir. Modellerin hazırlanmasında Python'un Gensim kütüphanesi kullanılmıştır. Gensim, yeni nesil makine öğrenmesi yöntemlerini kullanarak denetimsiz olarak metin indeksleme, benzerlik tespiti ve benzerliğe göre diğer işlemleri gerçekleştiren bir doğal dil işleme kütüphanesidir [73]. Bu çalışmada da Gensim'in Word2Vec aracı kullanılarak tüm işlemler gerçekleştirilmiştir. İncelenen modellerden en başarılı olan 4 ve 5 pencere boyutlu modellerin vektör temsilleri ve ağırlıkları derin öğrenme modellerinde test edilmiştir. Pencere boyutuna göre en iyi sonucu veren 4 pencere boyutu, çalışmada kullanılmak üzere belirlenmiştir.

Hazırlanan Python programı ile veri seti dosyası okunmuş, okunan dosyada her bir satırdaki dizilerden Şekil 4.10'da br örneği gösterildiği gibi üçer tane üçer uzunluktaki kelimelere bölünmüş diziler elde edilmiş ve bu yeni liste ile Word2Vec modeli çalıştırılmıştır. Yukarıda da belirtildiği gibi farklı pencere boyutlarında yapılan denemelerin sonucunda modellerin vektör boyutu "size" parametresi içerisine 300 olarak, pencere boyutu "window" parametresi içerisine 4 olarak verilmiştir. Ayrıca

üretilen tüm kelimelerin vektör temsilini hazırlamak ve ağırlıklarını kaydetmek için Word2Vec'in "min_count" parametresi 1 olarak verilmiş ve bu sayede tüm k-merler veri seti içerisinde bir defa dahi geçse kelime dağarcığına ve ağırlık dosyasına dahil edilmiştir. Word2Vec model testlerinin görselleştirilmiş olan yakınlık sonuçları Şekil 4.12'de verilmiştir.



Şekil 4.12. Çeşitli pencere boyutlarına göre Word2Vec sonuç grafikleri. Pencere boyutları a) 4, b) 5, c) 7, d) 10.

Derin öğrenme modellerinin giriş adımında Python'un Gensim modülünde Word2Vec ile hazırlanmış diziler Keras'ın gömme katmanına verilmiştir. Giriş ve çıkış boyutları olarak Word2Vec modelinin vektör boyutu verilmiştir. Aynı şekilde Word2Vec modelinin eğitimi sonucunda belirlenen ağırlıklar da ağırlık parametresine "weights=[wv.vectors]" ataması ile verilerek modelin bir adım önde eğitime başlaması sağlanmıştır. Gömme katmanının "eğitilebilir" parametresi de "trainable=True" ataması ile etkinleştirerek modellerin başarısı daha da artırılmıştır. Bu kısımdaki kelimelere bölme yapısı, Word2Vec vektör temsilinin ve ağırlıklarının kullanımı, bu tez kapsamında yapılmış olan çalışmanın literatüre kazandırmış olduğu yeniliklerin ön işleme ile ilgili olan kısmıdır.

Çizelge 4.3'te, Çift Katmanlı Çift Yönlü LSTM modelini kod sözlüğü ve Word2Vec ön işleme ile çalıştırmanın ilk epoch sonuçları sunulmuştur. Bu farklı önişleme yöntemleri denendiğinde Word2Vec modeli ile yapılan önişlemenin tüm modellerde en yüksek başarıyı sağladığı görülmüştür. Bu çalışma kapsamında önerilen hibrit modelde ve diğer tüm derin öğrenme modellerinde de bu ön işleme yöntemi kullanılmış ve hibrit yapı ile birlikte literatüre önemli bir yenilik ve katkı sağlanmıştır.

Çizelge 4.3. Çift Katmanlı Çift Yönlü LSTM modeli ile kod sözlüğü ile Word2Vec'in karşılaştırılması.

Önişleme Metodu	Eğitim Kayıp	Eğitim Doğruluk	Doğrulama Kayıp	Doğrulama Doğruluk
Kod Sözlüğü	2.8305	0.2519	2.2621	0.3770
Word2Vec	1.1427	0.7017	0.3897	0.8984

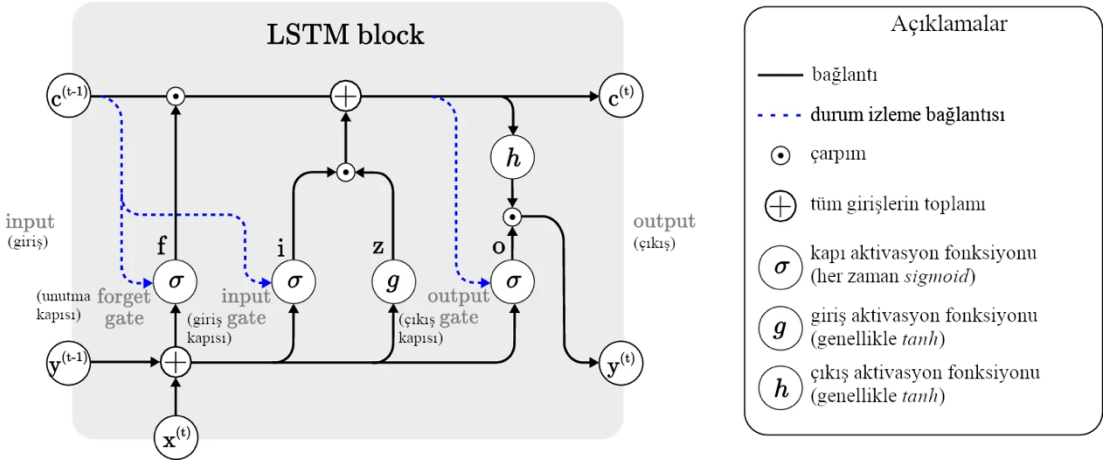
4.7. TEKRARLAMALI SİNİR AĞLARI

Konuşma tanıma ve analizi, dil ve çeviri işlemleri gibi sıralı ve uzun yapıda, tek boyutlu işlemlerde kullanılan Tekrarlayan Sinir Ağları (Recurrent Neural Networks-RNN), zamanla biyoinformatik temelli çalışmalarda da aranan bir araç olmuştur. RNN'lerde çıktı tahmin edilirken sadece andaki girişlere göre değil, geçmiş durumlarla beraber girişlere göre bir tahmin mekanizması çalıştırılmaktadır. Bu da geçmişteki tahminlerin, şimdiki ve gelecekteki tahminleri etkileyeceğini göstermektedir. RNN'ler, bünyesindeki gizli birimlerde verilerdeki sıralı bilgileri kullanmakta ve geçmiş durumları depolamakta, bu sayede tahminlerde bulunmaktadır.

Bununla birlikte, geri yayılan gradyanlar her adımda büyüdükçe veya küçüldükçe, Ufuk Gradyan veya Patlayan Gradyan sorunları ortaya çıkar ve geçmiş bilgiler hızla unutulmaktadır. Bu nedenle, bir sonraki kelime gibi yakın sıralama tahminlerde iyi sonuç verirken, uzun girişlerde sorunlar yaşanmaktadır. Uzun bağımlılıkları koruma sorunlarının üstesinden gelmek için literatürdeki çalışmalarda çeşitli yeni çözümler kullanılmaktadır [74]. Bu çözümlere Uzun Kısa-Vadeli Bellek Ağları ve Geçitli Tekrarlayan Ünite Ağları en önemli ve başarılı örnekler olarak verilebilir.

4.7.1. Uzun Kısa-Vadeli Bellek Ağları

RNN'lerin gradyanlardaki büyüme sonucu eski bilgileri unutma problemi sonrasında ortaya atılan fikirlerden biri de yapının içerisine bir karar verici mekanizma eklemek olmuştur. Bir dizideki uzun vadeli bağlamları unutmamak için RNN'lerde çeşitli değişiklikler yapılmıştır [75]. Ağın içerisine eklenen bir hafıza kapısı, insani türden bir doğal davranış olan önemli bilgilerin hatırlanıp, daha az öneme sahip olanların ise unutulmasını sağlamayı amaçlamıştır [74]. Çıkış noktası bu çözüm adımı olan Uzun Kısa-Vadeli Bellek (Long Short-Term Memory-LSTM) Ağları'nda tanh fonksiyonu ile RNN'lerde önceki gizli katmandaki ve mevcut gizli katmandaki bilgilerin işlenmesi prensibine dayalı olarak hücre yapısına bir unutma yapısı eklenmiştir [75]. İlk etapta LSTM içerisinde hata sinyallerini birimlerinin hücresinde tutan sabit hata döngüsü eklenerek önlenmişse de sonrasında ağın durumunu sıfırlayıp, unutma işlemini gerçekleştirerek başarıyı artırmak için Gers ve arkadaşları [76] tarafından unutma kapısı eklenmesi önerilmiştir [77]. Önceki adımlardan hangi bilgilerin unutulacağına ve hatırlanacağına karar vermek için her LSTM hücresine kapılar yerleştirilerek soruna bir çözüm üretilmiştir [75]. Bu kapılarda aktivasyn fonksiyonları olarak sigmoid aktivasyon fonksiyonları kullanılmaktadır [78]. Şekil 4.13'te tipik bir LSTM bloğunun tüm bölümleriyle beraber iç yapısı gösterilmiştir [77].



Şekil 4.13. Tipik bir LSTM bloğunun iç yapısı [77].

LSTM ağının yapısı Eşitlik [4.1-4.6]'da gösterilmiştir. Burada LSTM bloğunun girişi tüm blokların girişinin güncellenmesi için bulunmaktadır. Burada $x^{(t)}$ o andaki giriş, $y^{(t-1)}$ ise LSTM biriminin çıkışıdır. Güncelleme Eşitlik 4.1 ile yapılmaktadır [77]:

$$z^{(t)} = g\left(W_z x^{(t)} + R_z y^{(t-1)} + b_z\right) \tag{4.1}$$

Burada W_z ve R_z, $x^{(t)}$ ve $y^{(t-1)}$ ile ilişkili ağırlıklar, b_z ise bias ağırlık vektörüdür.

Giriş kapısı güncellemesi ise Eşitlik 4.2 ile yapılmaktadır. Burada $x^{(t)}$ o andaki giriş, $y^{(t-1)}$ ise LSTM biriminin çıkışıdır. $c^{(t-1)}$ de son iterasyondaki hücre değeridir [77].

$$i^{(t)} = \sigma\left(W_i x^{(t)} + R_i y^{(t-1)} + p_i \odot c^{(t-1)} + b_i\right) \tag{4.2}$$

Burada $\odot$ iki vektörün doğrusal çarpımını ifade etmektedir. W_i ve R_i ve p_i, $x^{(t)}$, $y^{(t-1)}$ ve $c^{(t-1)}$ ile ilişkili ağırlıklar, b_i ise bias ağırlık vektörüdür.

Hücre değeri olan $c^{(t)}$ vasıtasıyla LSTM hangi bilgilerin tutulacağını tayin edecek bilgiyi içermektedir. $z^{(t)}$, tutulmaya aday değerleri ve $i^{(t)}$ de aktivasyon değerlerini ifade etmektedir.

Unutma kapısında ise LSTM biriminin hangi bilgileri tutulacağı, hangi bilgileri ise unutacağı $c^{(t-1)}$ hücre durumu bilgisi ile belirlenmektedir. Unutma kapısının aktivasyon değeri olan $f^{(t)}$, mevcut giriş $x^{(t)}$, çıkış $y^{(t-1)}$ ve hücre durumu $c^{(t-1)}$ değerleri ile beraber hesaplanmaktadır. Bu hesaplamanın formülü Eşitlik 4.3'te gösterilmiştir [77].

$$f^{(t)} = \sigma\left(W_f x^{(t)} + R_f y^{(t-1)} + p_f \odot c^{(t-1)} + b_f\right) \tag{4.3}$$

Burada W_f ve R_f ve p_f, $x^{(t)}$, $y^{(t-1)}$ ve $c^{(t-1)}$ ile ilişkili ağırlıklar, b_f ise bias ağırlık vektörüdür.

Hücre değeri güncellemesi ise LSTM bloğunun girişi olan $z^{(t)}$, giriş kapısı $i^{(t)}$ ve unutma kapısı $f^{(t)}$ değeriyle güncellenmektedir. Unutma kapısından gelen değerle güncellenen hücre durum değeri, en doğru bilgilerin hatırlanmasını sağlayan esas yapıdır. Eşitlik 4.4'te formülize edilmiştir [77].

$$f^{(t)} = \sigma\left(W_f x^{(t)} + R_f y^{(t-1)} + p_f \odot c^{(t-1)} + b_f\right) \tag{4.4}$$

Çıkış kapısında mevcut giriş $x^{(t)}$, LSTM biriminin çıkışı olan $y^{(t-1)}$ ve son iterasyonun hücre durum değeri $c^{(t-1)}$ ile beraber çıkış hesaplaması yapılmaktadır. Bu hesaplama Eşitlik 4.5'te verilmiştir [77].

$$o^{(t)} = \sigma\left(W_o x^{(t)} + R_o y^{(t-1)} + p_o \odot c^{(t-1)} + b_o\right) \tag{4.5}$$

Burada W_o, R_o ve p_o, $x^{(t)}$, $y^{(t-1)}$ ve $c^{(t-1)}$ ile ilişkili ağırlıklar, b_o ise bias ağırlık vektörüdür.

Tüm LSTM bloğunun çıkışında mevcut hücre değeri olan $c^{(t)}$ ile mevcut çıkış kapısı değerini birleştiren blok çıkışı hesaplanmaktadır. Bu hesaplama Eşitlik 4.6'daki gibi yapılmaktadır [77].

$$y^{(t)} = g\left(c^{(t)}\right) \odot o^{(t)} \tag{4.6}$$

Denklemlerdeki σ, g ve h fonksiyonları, doğrusal olmayan aktivasyon fonksiyonlarıdır. Kapılarda aktivasyon fonksiyonu olarak lojistik sigmoid fonksiyonu $\sigma(x) = \frac{1}{1+e^{1-x}}$ kullanılmaktadır. Blok giriş ve çıkış aktivasyon fonksiyonları ise sıklıkla $g(x) = h(x) = \tanh(x)$ fonksiyonlarıdır [77].

4.7.2. Kapılı Tekrarlayan Birim Ağları

Kapılı Tekrarlayan Birim (Gated Recurrent Unit-GRU) Ağları, RNN'lerdeki problemler üzerine tasarlanan LSTM ağları ile benzer yapıda çalışmaktadır. Tasarımsal olarak da RNN'lere ve LSTM'lere benzemektedir. GRU ağları, RNN'lerde bulunan, gradyanlardaki büyüme sonucu eski bilgileri unutma problemine çözüm amaçlı olarak tıpkı LSTM'lerdeki mantıksal yapıya gibi bir tasarıma sahip ağlardır. Var olan uzun vadeli bağımlılıkları unutma problemi LSTM'de bir unutma kapısı vasıtası ile çözülürken GRU'da ise bu sorun bir unutma anahtarıyla çözülmüştür. GRU, içerisindeki unutma anahtarı ile anahtara bağlı kapılardaki değerleri uzun-kısa vadeli bağımlılıklara göre sıfırlamakta ve güncellemektedir [79]. GRU, performans açısından LSTM'lere benzer, ancak daha az karmaşıktır ve daha az kapıya sahip olduğundan çeşitli problemlerde LSTM'lerden daha hızlı olabilir. Bu çalışma kapsamında hazırlanan modellerde GRU ağları ile tasarlanan modellerin daha kısa bir eğitim süresine sahip olduğu açıkça görülmektedir. GRU'larda güncelleme kapısı, belleğe giren bilgileri kontrol etmektedir. Unutma kapısı, bellekten gelen bilgileri kontrol etmektedir. Bu kapılardaki ve anahtarlardaki işlemler Eşitlik [4.7-4.10] ile açıklanabilir.

Eşitlik 4.7'de t zamanında GRU'nun h_t aktivasyonu, önceki aktivasyon h_{t-1} ile GRU'nun $\tilde{h}_t$ aday aktivasyonu arasındaki doğrusal bir enterpolasyondur [79].

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \odot \tilde{h}_t \tag{4.7}$$

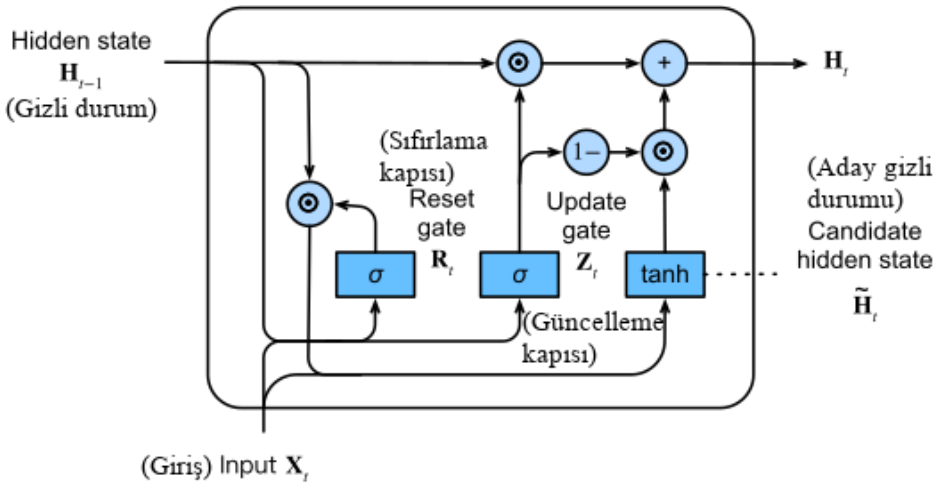
Burada z_t güncelleme kapısıdır. Birimin önceki adımdaki bilgileri ne kadar güncelleyip güncellemeyeceğine karar vermektedir. Eşitlik [4.8-4.10]'da ise her bir iterasyondaki güncelleme hesaplamaları belirtilmiştir [80].

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \odot \tilde{h}_t \tag{4.8}$$

$$r_t = \sigma_1(W_{hr} h_{t-1} + W_{xr} x_t + b_r) \tag{4.9}$$

$$\tilde{h}_t = \sigma_2(W_{chx} x_t + W_{chr}(r_t \odot h_{t-1}) + b_h) \tag{4.10}$$

Burada Eşitlik 4.8 ve 4.9 güncelleme kapısını ve sıfırlama kapısını ifade ederken, Eşitlik 4.10 ise aday aktivasyon fonksiyonunun güncellemesini ifade etmektedir.



Şekil 4.14. Tipik bir GRU bloğunun iç yapısı [80].

LSTM’in uzun metinlerdeki ve cümlelerdeki geçmiş gerekli bilgileri hafızada tutarak uzun vadeli bağımlılıkları yakalaması ve geksiz olan bilgileri unutarak bellek tasarrufu sağlaması, ayrıca önceki ve sonraki kelimeler arasındaki bağlamı başarılı bir şekilde tespit edebilmesi sebebi ile ve tüm bu özellikleri ek olarak GRU’nun eğitim ve çalışma sürelerinin daha görece kısa olması sebebiyle bu çalışmadaki tekli ve hibrit modellerin bir kısmında çift yönlü LSTM, GRU ve çift yönlü GRU mimarileri kullanılmıştır. Çünkü protein dizileri adeta uzun cümle grupları gibi değerlendirilebilir. Bu dizilerin içerisindeki TF protein ailelerini özelliklerini belirten motifler de tıpkı birer cümle veya kelime grubu gibi değerlendirilebilir. LSTM ve GRU mimarileri de yapıları itibaryle bu dizilerdeki karakterlerin ve k-merlerin aralarındaki uzun ve kısa vadeli bağımlılıkları tespit ederek motifleri yakalayabilecek ve bu sayede başarılı bir sınıflandırma işlemi gerçekleştirebileceklerdir.

LSTM ve GRU mimarilerindeki kapılar, anahtarlar ve hesaplamalar ile GRU'lar da LSTM'ler gibi bilgi akışını kontrol edebilir ve hangi uzun vadeli bağımlılıkların kalacağını ve hangilerinin unutulacağını belirleyebilir [81]. Çalışma prensipleri aynı olan LSTM ve GRU üzerinde yapılan testlerde iki ağın da başarılarının neredeyse aynı çıktığı, fakat GRU’da LSTM’e göre bir kapı eksik olduğundan eğitim ve çalışma süresinin daha hızlı olduğu görülmüştür. Bu sebepten de bu çalışma kapsamında önerilen derin öğrenme modelinde CNN ve çift yönlü GRU, yeni nesil veritabanının internet sitesinde çalışan modelde ise çift yönlü LSTM tabanlı katmanlar kullanılmıştır.

4.8. EVRİŞİMLİ SİNİR AĞLARI

Yapay zekâ çalışmalarının bir kolu da özellikle görüntüler, yani 2 boyutlu matrisler üzerine yönlenmiştir. Doğal bir görsel algılamayı kendine görev edinmiş olan Evrişimli Sinir Ağları (Convolutional Neural Networks-CNN), ilk olarak Hobel ve Wiesel’in [82] hayvan görsel korteksindeki hücrelerin, alıcı alanlardaki ışığı algılamaktan sorumlu olduğunu bulmasıyla beraber Fukushima, 1980'de CNN'nin atası olarak kabul edilebilecek Neocognitron [83]'u önermesiyle literatüre girmiştir. Yapay zekâ çalışmalarının ilk dönem yöntemlerinden olan çok katmanlı perceptron ağlarından sonra popülerliği artan CNN’lerin çığır açması ve güncel popülerliğini

kazanması ise LeCun ve ekibinin LeNet [84,85] isimli çalışması ile olmuştur [86]. Ardından da uluslararası alanda düzenlenen ImageNet yarışmasında AlexNet isimli CNN modeli ile beraber CNN'ler dünyaca tam bir tanınırlığa ve kullanıma sahip olmuştur. Bu sayede dünyadaki yapay zekâ çalışmalarının büyük bir çoğunluğuna temelinden yön vermiştir [87].

Yapay zekâ çalışmalarının önemli bir parçası olan CNN mimarisi ile beraber nesne analizi, örüntü tanıma, kanser tespiti, DNA ve protein tespiti, biyomedikal görüntü işleme çalışmaları, ses ve video işleme, koronavirüs temelli rahatsızlıkların tespiti gibi birçok çalışma yürütülmektedir. Bu alanlarda CNN temelli modellerin tercih edilmesinin en önemli sebepleri arasında bilhassa görüntüler üzerindeki işlem başarısı, eğitim ve çalışma hızı ve doğru ön işleme ile yüksek başarı oranı yer almaktadır. Bu çalışmalarda problemi tanımlayan, ifade eden özelliklerin çıkarılması ve bu özellikler vasıtası ile çalışmanın nevine göre sınıflandırma, regresyon analizi, kümeleme gibi birçok işlemi yapma prensibi uygulanmaktadır. CNN temelli çalışmalarda ham bir halde olan veri modelin giriş katmanından verilir, modelin içerisinde yer alan ve her biri farklı görevlerde yer alan katmanlar olan girdi katmanı, evrişim katmanı, havuz katmanı ve tam bağlantılı katman ile beraber özellik haritaları elde edilir ve devamında da probleme özgü sınıflandırma, regresyon analizi, kümeleme gibi işlemler gerçekleştirilerek sonuca ulaşılır [88].

4.8.1. Giriş Katmanı

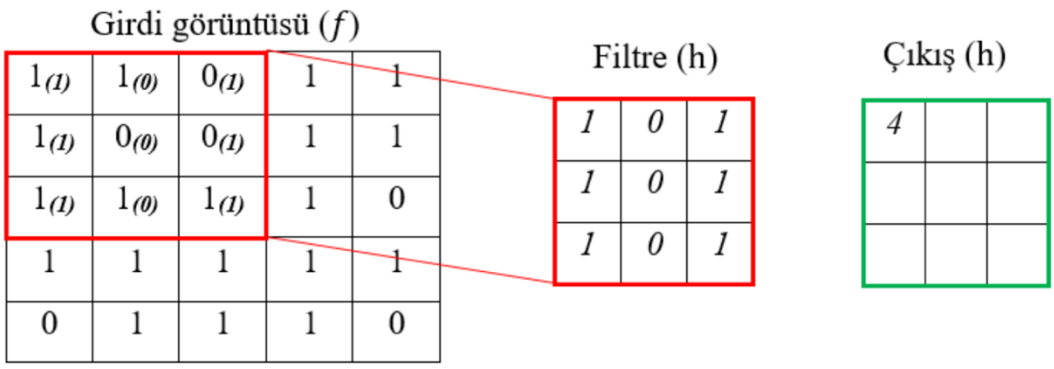
Giriş katmanı, CNN modeline işlenmemiş, ham bir vaziyette verilen bir giriş görüntüsü veya metnini ifade etmektedir. Bu katman, *d*-boyutlu yoğun bir vektör ile temsil edilen *n* giriş içermektedir. Temsili $d \times n$ boyutlu bir özellik haritası ile ifade edilmektedir [89]. Bu katman ile alınan ham veriler işlenerek görüntü hakkındaki detaylara sahip olmayı sağlar. Bu katmanın çıkışı sonraki katmana girdi olarak verilir. Boyutun küçük olması, modelin hızlı ve sağlıklı çalışması için önemlidir. Burada boyut eğer çok küçük seçilir ise verinin özelliklerinden kayıplar olabilecekken, çok büyük seçilir ise de gereken yoğun işlem gücü, çalışma süresini ve kaynak ihtiyacını oldukça artırır.

4.8.2. Evrişim (Convolution) Katmanı

Evrişim katmanı, giriş katmanından gelen verilerin ayırt edici özelliklerini tanımlayarak öğrenmeyi sağlamaktadır. Diğer bir ifadesi ise çekirdek katmanıdır. Doğrusal veya doğrusal olmayan işlemlerin bir kombinasyonundan oluşur ve özellik çıkarımı sağlar [90]. Bu katman içerisinde çeşitli filtreler görüntünün veya metnin giriş katmanı içerisinde matris formuna getirilmiş hali üzerinde gezdirilir. Bu işlem, matrisin sol üst köşesinden sağ alt köşesine kadar tüm matris üzerinde sırasıyla gerçekleştirilir. Her bir evrişim katmanında matrisin basit özelliklerinden derin özelliklerine doğru bir keşif söz konusudur. Burada filtreler kendi boyutları kadar matris bölümünü tek bir değere indirgeyerek matrisi yoğunlaştırır. Bu sayede hem özellikleri çıkarılmış ve belirginleştirmiş olur, hem de sonraki katmanların işlem gücü ihtiyacını ve işlem zamanını azaltmış olur [87]. Evrişim katmanı matematiksel olarak Eşitlik 4.11'deki gibi ifade edilir.

$$(f * h)[m,n] = \sum_{j}\sum_{k} h[j,k], f(m-j, n-k) \tag{4.11}$$

Burada f giriş katmanından gelen matrisi, h ise katmandaki çekirdeği ifade etmektedir. Çıkış matrisinin satır ve sütunu ise m ve n ile ifade edilir. j ve k ise bu katmanın çıkışında oluşacak olan yoğunlaşmış matrisin satır ve sütununu ifade eder [88]. Evrişim işleminin görselleştirilmiş hali Şekil 4.15'te verilmiştir.



Şekil 4.15. Tipik bir evrişim işlemi örneği [88].

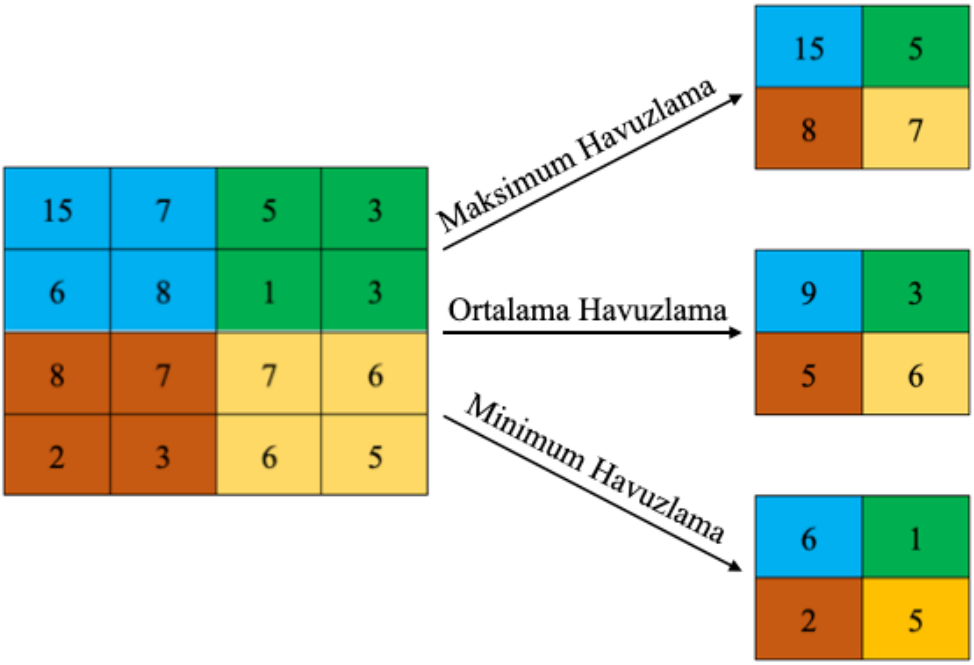
Şekil 4.15'te anlatılan işlemde giriş katmanından gelen matriste filtre soldan sağa ve yukarıdan aşağıya gezdirilerek her adımda Eşitlik 4.11 uygulanmış ve yoğun matris elde edilmiştir. Görselde sadece 3×3 boyutlu bir filtrenin işlemlerinin ilk adımı gösterilmiştir. Evrişim katmanının çalışması sonucu görüntünün geri kalanı da değişime uğrayacaktır. Ayrıca filtrenin matris üzerinde kayma sayısı (stride) arttıkça görüntü giderek küçülecektir. Çıkışın orantısız olarak küçülmesinin önüne geçmek için oluşan matrise doldurma işlemi (padding) uygulanır. Bu doldurma işlemi için literatüre bakıldığında çoğunlukla 0 rakamı ile doldurma işlemi yapıldığı görülmüştür [88]. Bu işlem tamamlandığında hem matrisin boyutu önemli ölçüde küçülecek, hem de özellikler başarılı bir şekilde çıkarılmış olacaktır. Yeni boyutun hesabı Eşitlik 4.12'de verilmiştir [91].

$$o = \frac{i - f + 2 * p}{s} + 1 \tag{4.12}$$

Burada o çıkış görüntü boyutunu, i giriş katmanından gelen matrisi (görüntü veya metin), f filtre boyutunu, p doldurma işleminin miktarını (satır ve sütun değeri) ve s ise filtrenin kayma miktarını ifade etmektedir [91].

4.8.3. Havuzlama (Pooling) Katmanı

Havuzlama katmanı, evrişim katmanının çıktısını kullanan bir örnekleme işlemidir. Bu katmanda, evrişim katmanından gelen matris boyutu pencere işlemi uygulanarak küçültülür [92]. Bu katmanda öğrenme işlemi yoktur fakat çok sayıda parametreden hesaplama karmaşıklığını azaltmak için bir kısmı atılmaktadır. Özellik kaybı olabileceğinden ötürü başarı azalma riski olduğundan dikkatli bir seçim yapılmalıdır [91]. Bu hesaplama düğümlerinin sayısını azaltır ve aşırı uyumu önler. Maksimum ve ortalama ve minimum olmak üzere üç tip havuzlama yöntemi vardır. Şekil 4.16'da bir matristen bu üç havuzlama yöntemine göre de oluşturulan matrislerin birer örneği verilmiştir.



Şekil 4.16. Adım sayısı 2, boyutu 2 olan bir havuzlama katmanı örneği.

4.8.4. Tam Bağlı (Fully Connected) Katman

Tam bağlantılı bir katmanda, son evrişim katmanının çıktısı tamamen düzleştirilir ve çıktını tek boyutlu bir vektöre dönüştürülmesi gerçekleştirilir. Bu katmanda sınıflandırma yapılır ve ağın nihai çıktılarına eşlenir. Önceki katmandaki tüm nöronlar bu katmanda yer alan nöronların tümü ile bağlıdır. Bu katman birden çok alt katmana sahip olabilir fakat bu katmanın son katmanındaki nöron sayısı, sınıflandırılacak olan veri setinin sınıf sayısı ile mutlaka eşit sayıda olmalıdır [93].

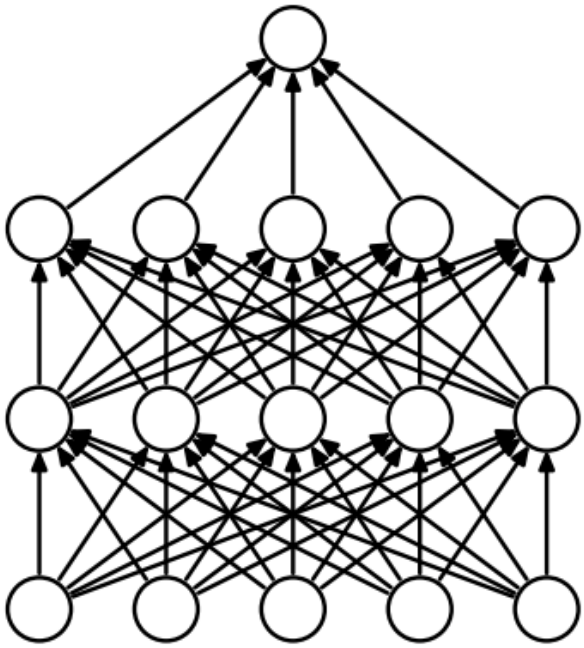
CNN'in evrişim katmanında uyguladığı filtreler ve yöntemler ile beraber görüntülerdeki, yani matrislerdeki özellik çıkarım başarısı aşikardır. Bu özellik çıkarım başarısı, metinlerin sayısallaştırılarak matrislere dönüştürülmesi sonrasında bu metinlerde de kullanılabilmektedir. Uzun protein dizilerinin Bölüm 4.6'da detaylı olarak anlatılan ön işleme metotlarıyla sayısallaştırılmış matris temsillerinin de CNN mimarisine sahip modeller ile sınıflandırma işlemine tabi tutulması, bu protein dizilerinin içerisinde yer alan motiflerin temsili için oldukça önemlidir. Çünkü protein dizileri içerisinde yer alan motifler kesintisiz, birleşik yapıdadır. Bu sebepten de bu motifler adeta bir görüntü içerisinde yer alan objelerin matris temsilleri içerisindeki

yapılarına benzemektedir. Bu sebeplerden de bu çalışma kapsamında hazırlanan tekli modellerin bir kısmında ve hibrit modellerin tümünde CNN mimarisi kullanılmış ve CNN'in özellik çıkarım performansı ile önerilen derin öğrenme modeli litertürdeki benzer veri setleri ile çalışan diğer yapay zekâ modelleri arasında en yüksek başarıyı göstererek literatüre katkı sağlamış ve ön plana çıkmayı başarmıştır.

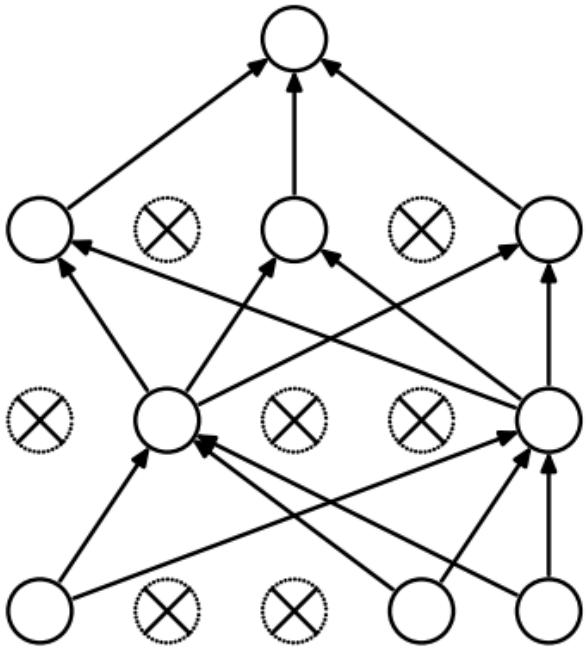
4.9. SEYRELTME (DROPOUT) KATMANI

Çok sayıda parametreye sahip derin sinir ağları, çok güçlü makine öğrenme sistemleridir. Bu sistemlerin gücü tasarımlarından ve barındırdıkları çok sayıdaki nörondan gelmektedir. Bu çok sayıda nöron bulunma durumu da bu ağlarda aşırı uyum yani ezberleme problemine yol açabilmektedir. Ezberleyen bir ağ eğitim esnasında yüksek bir doğruluk oranı verirken, sıra doğrulama ve test aşamalarına geldiğinde ise düşük doğruluk oranları ve yüksek kayıp oranları sergilemektedir. Diğer bir taraftan bu büyük ağların kullanımı da yavaştır. Bu sebepten de doğrulama ve test zamanında birçok farklı büyük sinir ağının tahminlerini birleştirerek fazla uydurma ile başa çıkmayı zorlaştırır. Seyreltme katmanı ise bu problemleri çözmek için Srivastava ve arkadaşları tarafından geliştirilmiş ve literatüre kazandırılmıştır.

Seyreltme işlemi, eğitim sırasında sinir ağındaki rastgele bazı nöronların bağlantıları ile birlikte koparılması ve devre dışı bırakılması işlemidir. Bu sayede ezberlemenin önüne geçilmektedir [94]. Seyreltme işleminde değişik oranlar kullanılmaktadır. Bu oranlar çalışmanın yapısına ve kullanılan veri setine göre seçilmektedir. Literatüre bakıldığında bu oranların genelde 0.1 ila 0.3 arasında değişmekte olduğu görülse de daha yüksek oranlarda seyreltme kullanımı da literatürde yer almaktadır [88]. Şekil 4.17'de sinir ağının seyreltme işlemi öncesi şeması, Şekil 4.18'de ise sinir ağının seyreltme işlemi sonrası şeması görülmektedir.



Şekil 4.17. Sinir ağının seyreltme öncesi şeması [94].



Şekil 4.18. Sinir ağının seyreltme sonrası şeması [94].

Seyreltme katmanının, sinir ağlarının görme, konuşma tanıma, belge sınıflandırma ve hesaplamalı biyolojideki denetimli öğrenme görevlerinde performans artırdığı hm literatürdeki çalışmalarda açıkça görülmektedir [94].

4.10. AKTİVASYON FONKSİYONU

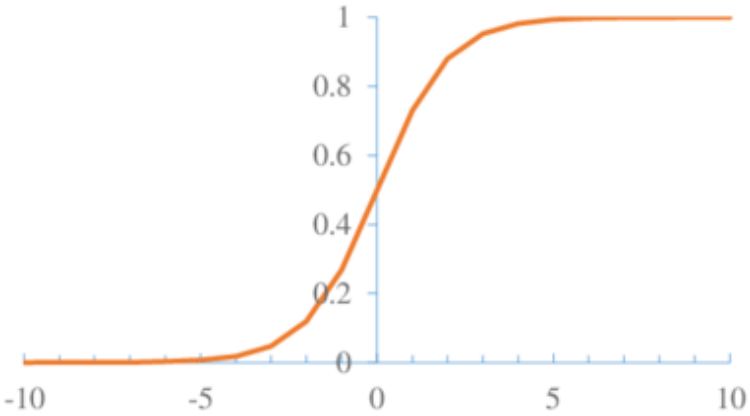
Sınıflandırma işleminin başarısı için etkili parametrelerden biri de aktivasyon fonksiyonudur. Sinir ağında her bir adımda giriş değeri, ağırlık değeri ve bias ile beraber yapılan bir hesaplama ile güncelleme işlemi yapımaktadır. Elde edilen sonuç, sonraki adımın girişi veya ağın çıkışı olabilir. Aktivasyon fonksiyonları ise ağın her adımında bu işlemler içerisinde bir dönüştürücü konumundadır. Bu güncelleme işleminden elde edilen sonuçlar bir doğrusal fonksiyon sonucudur ve doğrusal durumları ifade etmektedir. Derin öğrenme çalışmalarında ise sadece lineer regresyon gibi doğrusal işlemlerin yanı sıra hava durumu tahmini, kanser tespiti gibi doğrusal olmayan sonuçların da tahmini istenmektedir. Bu durumlarda da aktivasyon fonksiyonları devreye girmekte ve belli nöronların aktif veya pasif olma durumunu tespit etmektedir. Bu sayede ağın öğrenmesine ve doğrusal olmayan durumların da değerlendirilebilmesine olanak sağlanmış olur [95]. Aktivasyon fonksiyonlarına Basamak, Doğrusal, Sigmoid, Hiperbolik Tanjant, ReLU, Sızıntı ReLU, Softmax ve Swish aktivasyon fonksiyonları örnek verilebilir. Bunlardan literatürde sıklıkla kullanılanları ise Sigmoid, Hiperbolik Tanjant, ReLU ve Softmax aktivasyon fonksiyonlarıdır.

4.10.1. Sigmoid Aktivasyon Fonksiyonu

Türevi alınabilir bir fonksiyon olan Sigmoid, türevi alınabilmesinden ötürü öğrenmenin gerçekleştirilebildiği bir fonksiyon olarak göze çarpar ve çokça kullanılmaktadır [96]. Girdi olarak aldığı değerleri, 0 ila 1 arasındaki değerlere yerleştirir. Bu sayede gradyan patlamasının önüne geçilmiş olur. Sigmoid aktivasyon fonksiyonunun formülizasyonu Eşitlik 4.13'te verilmiştir [97].

$$\sigma = \frac{1}{1 + e^{-x}} \tag{4.13}$$

Sigmoid aktivasyon fonksiyonunun özet bir grafiği aşağıda Şekil 4.19'da verilmiştir.



Şekil 4.19. Sigmoid aktivasyon fonksiyonunun özet bir grafiği [98].

Sigmoid aktivasyon fonksiyonunda giriş değerleri çok küçüldüğünde veya çok büyüdüğünde çıkış değerlerindeki oynama çok küçük olacaktır. Bu sebepten de bu bölgelerde türev değerleri çok küçük olur, dolayısıyla da gradyan yok olması problemi ortaya çıkar [99].

4.10.2. Hiperbolik Tanjant Aktivasyon Fonksiyonu

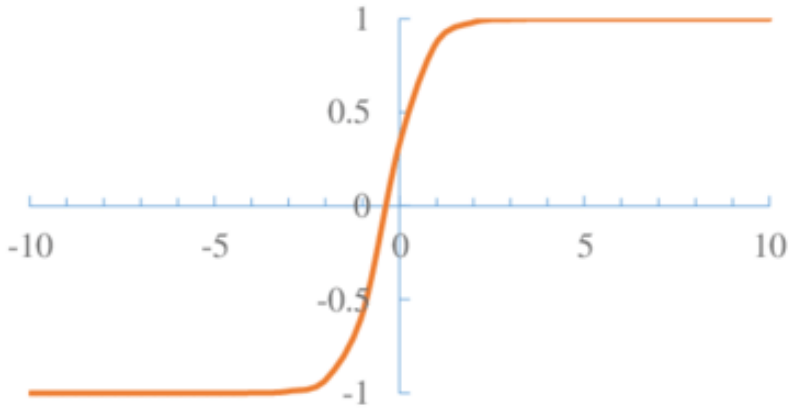
Sigmoid aktivasyon fonksiyonu ile aynı görevi üstlenen bir diğer fonksiyon olan hiperbolik tanjant aktivasyon fonksiyonu (tanh), sigmoid aktivasyon fonksiyonunundan farklı olarak -1 ila 1 aralığında çıkış değeri almaktadır. Değer aralığı geniş olduğundan ötürü daha başarılı bir öğrenme olacaktır. Hiperbolik tanjant fonksiyonun matematiksel temsili Eşitlik 4.14'te verilmiştir [100].

$$\tanh(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}} \tag{4.14}$$

Hiperbolik tanjant fonksiyonu, sigmoid aktivasyon fonksiyonu kullanılarak da elde edilebilmektedir. Eşitlik 4.15'te sigmoid aktivasyon fonksiyonu ile hiperbolik tanjant aktivasyon fonksiyonunun matematiksel temsili gösterilmiştir [100].

$$\tanh(x) = 2\sigma(2x) - 1 \tag{4.15}$$

Hiperbolik tanjant aktivasyon fonksiyonunun özet bir grafiği aşağıda Şekil 4.20'de verilmiştir.



Şekil 4.20. Hiperbolik tanjant aktivasyon fonksiyonunun özet bir grafiği [98].

Tıpkı sigmoid aktivasyon fonksiyonunda olduğu gibi giriş değerleri çok küçüldüğünde veya çok büyüdüğünde çıkış değerlerindeki oynama çok küçük olacaktır. Bu sebepten de hiperbolik tanjant fonksiyonunda da gradyan yok olması problemi olmaktadır [99].

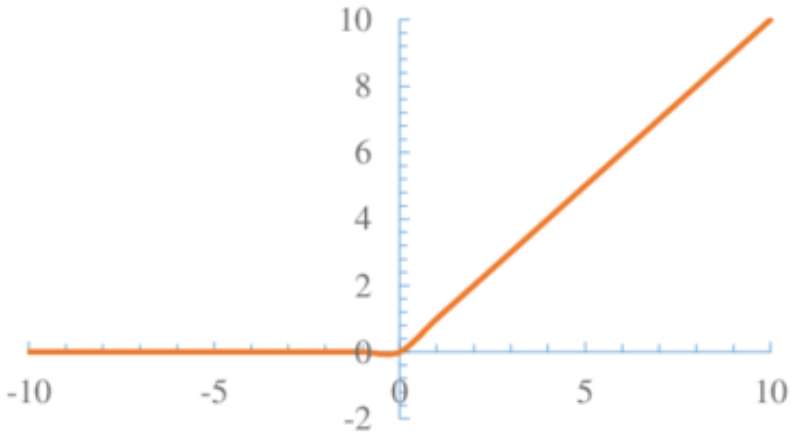
4.10.3. ReLU Aktivasyon Fonksiyonu

Doğrultulmuş Doğrusal Birim (Rectified Linear Unit-ReLU) aktivasyon fonksiyonu, girişler 0'ın altında ise çıkış olarak 0 üretmektedir. 0'dan büyük değerlerde ise girişi aynen çıkışa iletir. ReLU'da bazı nöronlar pasifize edildiğinden ve negatif değerlerde 0 çıkışı üretildiğinden daha az işlem yapılmış olur ve hesaplama maliyeti azalır. Ayrıca ReLU'nun gradyanı (veya eğimi), pozitif girdiler için 1 veya negatif olanlar için 0'dır, böylece gradyan yok olması sorunu çözülmüş olur. Ancak ağırlıkların küçük rastgele değerlere uygun şekilde başlatılmasının sağlamasına rağmen, büyük ağırlık güncellemeleriyle, ReLU aktivasyon fonksiyonuna toplam girdi her zaman negatiftir ('ölmekte olan ReLU' sorunu). Bu negatif değer, çıktıda bir sıfır değeri verir ve karşılık gelen düğümlerin sinir ağı üzerinde herhangi bir etkisi kalmaz, bu da yanlış

sınıflandırmaya yol açarak bir görüntüde yer alan bir patolojiyi tespit etme yeteneğinin olmamasına neden olabilir [101]. ReLU aktivasyon fonksiyonunun matematiksel temsili Eşitlik 4.16'da verilmiştir [101,102].

$$\tanh(x) = 2\sigma(2x) - 1 \tag{4.16}$$

ReLU aktivasyon fonksiyonunun özet bir grafiği aşağıda Şekil 4.21'de verilmiştir.



Şekil 4.21. ReLU aktivasyon fonksiyonunun özet bir grafiği [98].

4.10.4. Softmax Aktivasyon Fonksiyonu

Softmax aktivasyon fonksiyonu, ikili veya çok sınıflı verilerin sınıflandırılmasında kullanılan, doğrusal olmayan bir aktivasyon fonksiyonudur. Olasılık tabanlı sınıflandırma çalışmalarında sıkılıkla kullanılmaktadır. Bu fonksiyon bir dağılımı ifade etmekte ve sonuçları [0, 1] kapalı aralığı içerisinde temsil etmektedir [103]. Softmax aktivasyon fonksiyonu ile bir sınıfa ait olma olasılığı belirlendiğinden ötürü derin öğrenme modellerinin tam bağlantılı katmanlarının sınıflandırma alt katmanında Softmax aktivasyon fonksiyonu sıklıkla kullanılmaktadır. Softmax aktivasyon fonksiyonunun matematiksel temsili Eşitlik 4.17'de verilmiştir [101].

$$f(x) = \frac{e^{x_i}}{\sum_{j=1}^{J} e^{x_j}}, \forall\, i = [1, J] \tag{4.17}$$

4.11. MODEL PARAMETRELERİ

4.11.1. Batch Boyutu

Model eğitilirken veri seti ne kadar büyükse tüm veriyi aynı anda işlemek bir o kadar zorlaşır ve maliyetli olur. Dolayısıyla veri seti tümüyle değil parça parça işlendiği takdirde daha düşük maliyetli ve kolay hesaplı bir eğitim süreci gerçekleştirilmiş olur. Bu işlemin yapılabilmesi için batch size parametresine istenilen değerler verilmelidir. Bu değerler, hafızanın boyut tasarımı ele alındığında 2'nin kuvvetleri şeklinde seçilmelidir. Bu değer, modelin birim zamanda işleyeceği verinin miktarını belirlemektedir. Bu çalışmadaki derin öğrenme modellerinin geliştirilmesinde kullanılan Keras kütüphanesi içerisindeki mimarilerde batch boyutu varsayılan olarak 32 belirlenmiş olmasına karşın bu çalışmada kullanılan bitki TF protein veri setindeki protein dizileri uzun olduğundan ve metin yapısındaki dizilerin işlenmesi, görüntülerin işlenmesine göre görece daha az maliyetli olduğundan yapılan testler sonucunda bu çalışmadaki modellerdde 128 ve 256 batch boyutları kullanılmıştır. Yeni nesil bHLH TF faktör veritabanı içerisinde kullanılan çift yönlü LSTM modelinde ve bitki TF protein veri setini sınıflandırmak üzere önerilen hibrit modelde batch boyutu 256 olarak tespit edilmiştir.

4.11.2. Epoch

Derin öğrenme modelleri de tıpkı insan beyni gibi belli bir süreç ve eğitim ile öğrenmeyi sağlarlar. Onun için de bu modellerin veri setleri üzerinde birden çok kere geçerek tekrar yapmaları esastır. Bu tekrarın sayısının belirlendiği parametre de epoch parametresidir. Veri seti üzerinde epoch parametresine verilen değer kadar tekrar yapılarak öğrenme gerçekleştirilmektedir. Fakat modellerde belli bir tekrar sayısının üstünde öğrenmenin yavaşladığı görülebilmektedir [104]. Tekrar sayısı daha da arttığında modelde ezberleme olacak, dolayısıyla da doğrulama ve test başarısı gittikçe

düşecektir. Bu sebepten de bazı çalışmalara bir erken durdurma (early stop) mekanizması eklenmiş ve her bir tekrarın hata durumu kontrol edilerek modelin eğitiminin belirlenen epoch değerine ulaşılmasa bile belli bir noktada kesilmesi ve tamamlanması sağlanmıştır [105]. Bu sayede daha az deneme ile en iyi başarıya ulaşılabilmektedir.

Bu tez kapsamında gerçekleştirilen derin öğrenme modellerinin her birinde en doğru epoch sayısını tespit etmek ve bu tespit süresinde de deneme yanılma yaparak zaman ve kaynak kaybının önüne geçmek için bir erken durdurma mekanizması tüm modellere eklenmiş ve en doğru epoch değerinin tespit edilmesi sağlanmıştır. Bu erken durdurma mekanizması eğitim esnasında modelin doğulama kayıp değerini takip etmekte ve gelen değer ard arda üç defa yükselme trendine geçerse model eğitimi kesmekte ve trend içerisinde en yüksek başarı ile eğitimi tamamlamaktadır.

4.11.3. Öğrenme Oranı

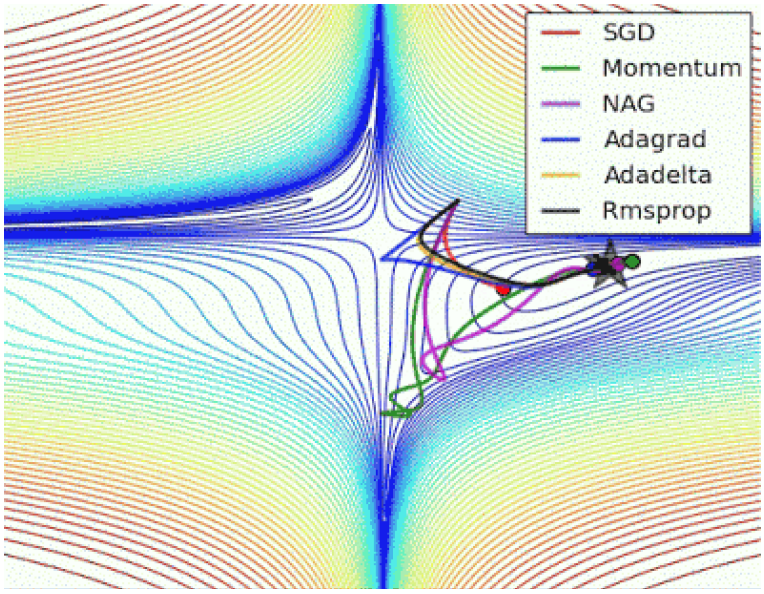
Modelin her adımında geri yayılım esasına göre geriye doğru türev alma işlemi yapılır ve gerçek değer ile tahmin edilen değer arasındaki fark bulunarak öğrenme oranı ile çarpılır ve en düşük hataya yaklaşmak amaçlanır. Öğrenme katsayısının seçimi bu sebepten oldukça büyük önem arz etmektedir. Büyük seçilen bir öğrenme oranı global minimum hata değerine yaklaşmayı engeller ve belli değerler arasında salınıma yol açar. Küçük bir öğrenme katsayısı seçildiğinde de global minimum hata değerine yaklaşmak çok uzun zaman alacak ya da lokal minimum değerin aşılamayarak genel minimum hata değerine hiçbir zaman ulaşılamamasına sebep olacaktır [106,107].

Bu çalışma kapsamında öğrenme katsayıları, LSTM ve GRU ağlarında 0.01, CNN ağlarında 0.001 ve hibrit modellerde 0.01 varsayılan değerler olarak belirlenmiştir.

4.12. OPTİMİZASYON ALGORİTMALARI

Optimizasyon algoritmaları, model parametrelerinin güncellenmeye devam etmesini ve kayıp fonksiyonunun değerinin en aza indirilmesini sağlayan algoritmalardır. Derin öğrenme modellerinin eğitimleri kullanılan veri setlerine ve modellerin derinliklerine

ve katmanlarına göre saatler, günler ve hatta haftalar alabilmektedir. Optimizasyon algoritmasının performansı, modelin eğitim verimliliğini doğrudan etkilemektedir [80]. Uygun optimizasyonu seçildiğinde hem model daha başarılı hem de daha hızlı bir öğrenme gerçekleştirir. Popüler optimizasyon algoritmaları olarak Gradyan Azalma, Adagrad, Adadelta, RMSProp ve ADAM algoritmaları literatüre bakılarak söylenebilir [88]. Optimizasyon algoritmaları aynı görevleri üstlenseler de aralarında hız performans farklılıkları mevcuttur. Optimuma ulaşma açısından örnek bir grafik Şekil 4.22'de gösterilmiştir.



Şekil 4.22. Optimizasyon algoritmalarının optimuma ulaşma performansları [108].

4.12.1. Uyarlanabilir Moment Tahmini Algoritması

Uyarlanabilir Moment Tahmini (Adaptive Moment Estimation-ADAM) algoritması, modelin her adımında öğrenme katsayısını güncellemektedir [109]. ADAM'ın temel bileşenlerinden biri hem momentumun hem de gradyanın ikinci momentinin bir tahminini elde etmek için üstel ağırlıklı hareketli ortalamaları kullanmasıdır [80]. ADAM, seyrek eğimlerde iyi çalışan Adagrad algoritması ve sabit öğrenme katsayılı RMSProp algoritmasının bir birleşimi gibi çalışmaktadır [88]. RMSProp'tan farklı olarak gradyan değerinin kendini değil, momentumu günceller. ADAM algoritmasının matematiksel temsili Eşitlik [4.18–4.22]'de verilmiştir.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{4.18}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{4.19}$$

$$\acute{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{4.20}$$

$$\tilde{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{4.21}$$

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\tilde{v}_t + \epsilon}} \cdot \acute{m}_t \tag{4.22}$$

Burada β_1 ve β_2 parametreleri hiper parametreler, m_t, θ_t önceki gradyanların üssel ortalamaları, v_t ise gradyanların karelerini ifade eder [109].

Bu çalışma kapsamında hazırlanan modellerde optimizasyon başarısı ve optimuma ulaşma hızı sebebiyle aktivasyon fonksiyonu olarak en çok kullanılan optimizasyon algoritmalarından olan ADAM algoritması kullanılmıştır.

4.13. KAYIP FONKSİYONU

Kayıp fonksiyonu (loss function), ileri yayılım tarafından verilen tam referans etiketlerini kontrol ederek ağın tahmin edilen çıktılarının doğruluğunu kontrol eder. Çok sınıflı veri setleri üzerinde çalışan modellerde en yaygın olarak kullanılan kayıp fonksiyonu, kategorik çapraz entropi (categorical crossentropy) fonksiyonudur [90].

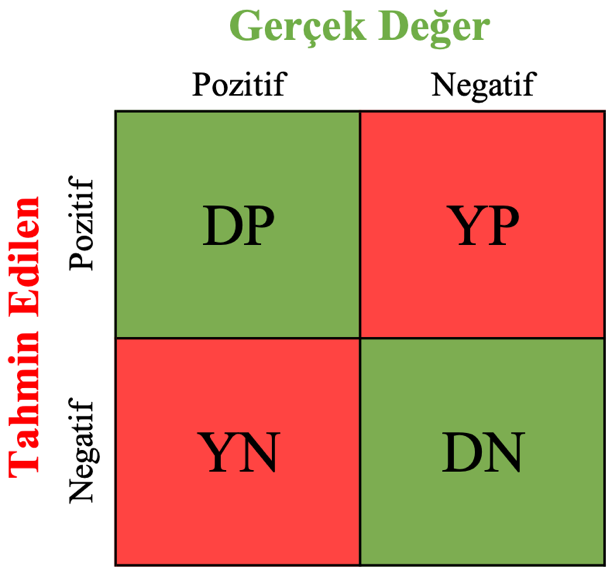
Bu sebepten de çok sınıflı bir veri seti ile hazırlanmış olan bu çalışma kapsamında hazırlanan modellerde kategorik çapraz entropi fonksiyonu kayıp fonksiyonu olarak kullanılmıştır.

4.14. PERFORMANS DEĞERLENDİRME KISTASLARI

Bu çalışmada da olduğu gibi sınıflandırma problemlerinde çeşitli performans değerlendirici araçlar kullanılarak modelin standart performans değerlendirmeleri ve kıyaslamaları yapılmaktadır. Bu araçlara örnek olarak karmaşıklık matrisi (confusion matrix), bu matristeki verilerden elde edilen ölçütler, Alıcı İşlem Karakteristiği eğrisi ve k-katlı çapraz doğrulama verilebilir.

4.14.1. Karmaşıklık Matrisi

Karmaşıklık matrisi (confusion matrix), iki veya daha fazla sınıfa sahip modellerin sonuçlarını analiz etmek ve performans ölçütlerine göre kıyas yapabilmek için üretilen bir araçtır. Bu kare matris içerisinde yatay pozisyon gerçek değerleri, dikey pozisyon da tahmin edilen değerleri ifade eder ve bunların kesişim değerlerine göre çeşitli analizler yapılır [110]. Oluşan matris, adeta bir performans değerlendirme özetidir. Şekil 4.23'te iki sınıflı bir derin öğrenme modeli için üretilmiş olan temsili bir karmaşıklık matrisi verilmiştir.



Şekil 4.23. Karmaşıklık matrisi.

Şekil 4.23'te bir örneği belirtilen karmaşıklık matrisinden yola çıkılarak standart performans değerlendirme metrikleri elde edilmektedir. Bu metriklere doğruluk

(accuracy), duyarlılık-hassasiyet (sensivity-recall), özgünlük (specifity), kesinlik (precision), f-skor (f-score) ve ortalama doğruluk (average-precision) örnek verilebilir. Bu metriklerin metematiksel temsilleri Eşitlik [4.23-4.28]'de verilmiştir [111].

$$Doğruluk\ (Acc) = \frac{DP + DN}{DP + YP + YN + DN} \tag{4.23}$$

$$Duyarlılık\ (Se) = \frac{DP}{DP + YN} \tag{4.24}$$

$$Özgünlük\ (Sp) = \frac{DN}{DN + YP} \tag{4.25}$$

$$Kesinlik\ (Pre) = \frac{DP}{DP + YP} \tag{4.26}$$

$$F - skor = \frac{2 * DP}{2 * DP + YP + YN} \tag{4.27}$$

$$Ortalama - Doğruluk\ (AP) = \frac{DP}{DP + DN} \tag{4.28}$$

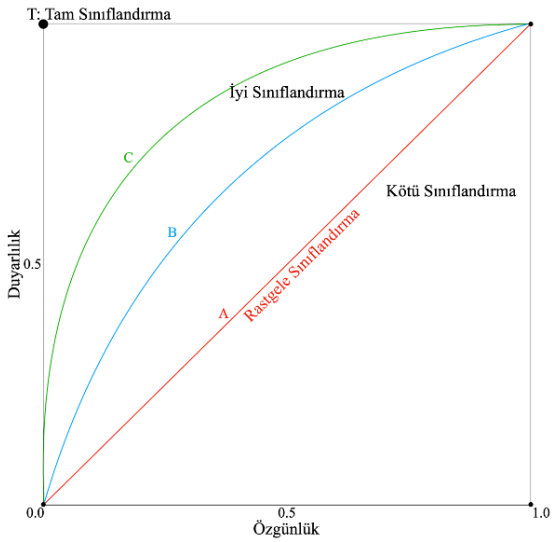
Burada DP doğru pozitif, DN doğru negatif, YP yanlış pozitif ve YN ise yanlış negatif değerleri ifade etmektedir. Modelin eğitim ve doğrulama adımlarından sonra oluşan karmaşıklık matrisinden bu değerler elde edilmeltedir. Akabinde de bu değerlere göre belirtilen performans metrikleri hesaplanabilmektedir.

Bu çalışma kapsamında hazırlanan modellerin başarılarını ölçebilmek ve kıyaslayabilmek adına her bir model için bu değerlerden doğruluk, kesinlik, hassasiyet ve f-skor değerleri hesaplanmış, ayrıca ROC grafikleri için de bu değerlerden duyarlılık ve özgünlük değerleri kullanılmıştır.

4.14.2. Alıcı İşlem Karakteristiği

Alıcı işlem karakteristiği (Receiver Operating Characteristic Curve-ROC Curve), hazırlanmış ve eğitilmiş olan modelin veri setindeki her bir sınıfı tahmin etme doğruluğunu ortaya koyan bir grafik yöntemidir. Tüm olası sınıf eşik değerleri üzerinden sınıfların duyarlılığı ve özgünlüğü arasındaki dengeyi göstermektedir. Bu işlemi, karmaşıklık matrisinden elde edilen DP, DN, YP ve YN değerlerinden hesaplanan duyarlılık ve özgünlük metrikleri ile gerçekleştirmektedir.

Bu grafiklerde x ekseni özgünlük, y ekseni ise duyarlılık değerlerini barındırır [112]. Bir ROC eğrisinin altında kalan alan 1'e ne kadar yakınsa yani her bir sınıfa ait eğriler grafiğin sol üst köşesine ne kadar yakın olarak toplandıysa modelin o kadar başarılı olduğu kanaatine varılmaktadır. Bu eğri altındaki alan için en büyük değer 1 (tüm sınıflandırmalar doğru), en küçük değer ise 0.5 (rastgele sınıflandırma)'tir [113]. Tanımlayıcı bir ROC grafiği Şekil 4.24'te verilmiştir. Grafikteki T noktası tüm sınıfların doğru tahmin edildiği 1 doğruluk oranına sahip durumu ifade etmektedir. A eğrisi 0.5 değerine sahip olan rastgele sınıflandırma eğrisidir. Eğrinin altı daha kötü sınıflandırmayı, üstü de daha iyi sınıflandırmayı ifade etmektedir. B eğrisi görece iyi bir sınıflandırmayı, C eğrisi ise daha iyi bir sınıflandırmayı ifade etmektedir. Bu eğriye göre model A sınıfını rastgele sınıflandırmış, B ve C sınıflarını görece daha iyi sınıflandırmış ve T sınıfını ise tam doğrulukla sınıflandırmıştır.

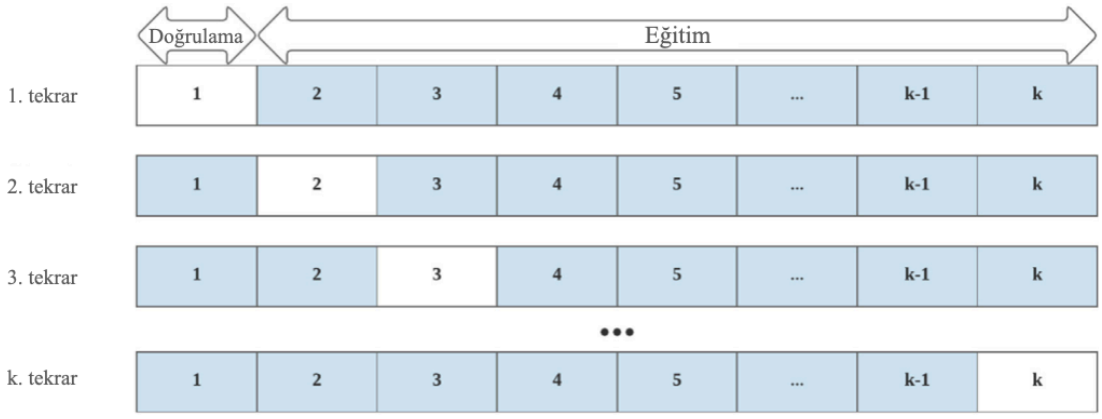


Şekil 4.24. Çeşitli durumları gösteren örnek bir ROC eğrisi.

4.14.3. K-Katlı Çapraz Doğrulama

Modeller geliştirilirken kullanılacak olan veri setleri hem eğitimde hem de testte kullanılmalıdır. Model eğitilirken veri setinin bir kısmı ile eğitim yapılırken, modele hiç verilmemiş olan bir kısmı ile de test yapılır [114]. Bu sayede modelin daha önce hiç karşılaşmadığı durumlara göre tepkisi ve gerçek çalışmasındaki başarı durumu tespit edilmiş olur. K-katlı çapraz doğrulama yönteminde ise geliştirilen modelin performansı üzerinde istatistiksel analizler yapılır. Bu sayede model hakkında bir genelleme yapılmış olur [115]. K-katlı çapraz doğrulama yönteminde modeller olabilecek birçok kombinasyonda eğitim ve test süreçlerine tabi tutularak modelin hem ezberlemenin önüne geçilerek başarısına hem de en doğru değerlendirmesine ulaşılması amaçlanmıştır.

Bu işlem yapılırken veri seti k parçaya bölünür. Ardından yöntemin her çalışmasında veri setinin farklı k-1 parçası eğitim, 1 parçası ise doğrulama için kullanılır. Bu yöntem k defa çalışır ve her çalışmasında veri setinin farklı parçaları eğitim ve doğrulama için kullanılır [116]. Yapılan k adet işlemin aritmetik ortalaması alınarak modelin nihai başarısı elde edilir. Şekil 4.25'te k-katlı çapraz doğrulama yönteminin yapısı gösterilmiştir.



Şekil 4.25. k-katlı çapraz doğrulama [116].

4.15. VERİTABANI VE İNTERNET SİTESİ ARAÇLARI

Bu çalışma kapsamında hazırlanan yeni nesil veritabanı için çeşitli araçlar kullanılmıştır. Bu yeni nesil veritabanı bir derin öğrenme modelini de bünyesinde barındırdığı için bütün yapının uyum içerisinde çalışmasını sağlamak amacıyla SQLite veritabanı yönetim sistemi, Python programlama dili ve Python programlama dili için hazırlanmış olan yüksek seviyeli bir web çatısı olan Django kullanılmıştır [117]. Tüm bu araçlarla hazırlanan veritabanı ve internet sitesi internet üzerinde yayınlanmıştır ve tüm araştırmacılar için kullanıma hazırdır.

4.15.1. SQLite Veritabanı Yönetim Sistemi

SQLite, bağımsız, sunucusuz, yapılandırma ihtiyacı olmayan, etkileşimli bir SQL veritabanı yönetim sistemidir. SQLite açık kaynak kodludur ve bu nedenle ticari veya özel herhangi bir amaç için kullanım için ücretsizdir. SQLite dünyanın en yaygın olarak kullanılan veritabanıdır [118]. SQLite, gömülü bir SQL veritabanı motorudur. Diğer SQL veritabanlarının çoğundan farklı olarak, SQLite ayrı bir sunucu işlemine sahip değildir. SQLite, doğrudan sıradan disk dosyalarını okur ve yazar. Birden çok tablo, dizin, tetikleyici ve görünüm içeren eksiksiz bir SQL veritabanı, tek bir disk dosyasında bulunur. Veritabanı dosya formatı platformlar arası aktarılabilir bir yapıdadır [119].

Bu çalışma kapsamında SQLite veritabanı yönetim sisteminin hafif ve tasarruflu yapısı, ayrıca Django web çatısı içerisinde varsayılan veritabanı yönetim sistemi olması ve veritabanının verilerinin dizilerden ve dizilerin belli tanımlayıcı bilgilerinden oluşan sade bir yapıda olması sebebiyle karmaşık ve maliyetli yapılardan uzak durmak ve daha az bellek ve depolama alanı kullanmak için SQLite veritabanı yönetim sistemi kullanılmıştır.

4.15.2. Django Web Çatısı

Django, hızlı geliştirmeyi ve temiz, pragmatik tasarımı teşvik eden üst düzey bir Python web çerçevesidir. Yeni nesil bir web geliştirme mimarisi olan Model-View

Controller (MVC) mimarisini kullanan bir web çatısıdır [120]. Python programlama dili ile geliştirme yapılmaktadır. Tüm modelleri birer Python fonksiyonu olarak çatı mimarinin altında çalışmakta, bu sayede çalışma uyumu ve kolaylığı sağlanmaktadır.

Bu çalışma kapsamında hazırlanan ve yeni nesil veritabanının internet sitesine yerleştirilen derin öğrenme modeli Python programlama dili ile hazırlanmıştır. Django'nun yapısı itibariyle bu modelin site içerisinde bir Python fonksiyonu olarak direkt çalıştırılabilmesi ve ayrıca Linux dağıtımlarında uçbirim üzerinde kolay ve hızlı bir şekilde çalışan HMMER analiz aracının Python ile bir alt süreç olarak sitenin depolandığı sunucuda hızlı bir şekilde kullanılabilmesinden ötürü Django web çatısı bu çalışmada tercih edilmiştir. Bu çalışmadaki derin öğrenme modelleri gibi tüm araçlar ve programlar Django içerisinde, dolayısıyla da hazırlanan internet sitesi içerisinde direkt olarak kullanılabilmektedir.

BÖLÜM 5

DENEYSEL ÇALIŞMALAR

Bu tez çalışması kapsamında bitki TF proteinlerinin sınıflandırılması için çeşitli derin öğrenme modelleri tasarlanmıştır. Bu kapsamda Word2Vec kelime vektör yapısı, Çift Yönlü LSTM, GRU, Çift Yönlü GRU ve CNN modelleri kullanılmaktadır. Tekli modellerin yanı sıra literatüre katkı sağlayacak Word2Vec + CNN + Çift Yönlü LSTM, Word2Vec + CNN + GRU ve Word2Vec + CNN + Çift Yönlü GRU hibrit modelleri oluşturulmuştur. Tek seferlik çalışma sonuçları ve grafikleri ile 10 kat çapraz doğrulama sonuçları hazırlanmıştır. ADAM, optimizasyon fonksiyonu olarak kullanılmıştır. Ağın sağlıklı bir eğitim sürecine sahip olması için erken durdurma aracı eklenmiş ve her modele uygun epoch sayısı belirlenmiştir. Tüm bu çalışmalar Python programlama dili ile gerçekleştirilmiştir.

Ayrıca bHLH TF proteinleri için referans bir yeni nesil veritabanı oluşturulmuştur. Bu yeni nesil veritabanı içerisinde bHLH TF proteinleri ile ilgili genel bilgiler, bu proteinlerin gen, transkripsiyon ve NCBI tanımlayıcı numaraları ile DNA ve protein dizileri yer almaktadır. Literatürde kabul gören HMM ve BLAST analiz araçları ile beraber yeni bir çift yönlü LSTM temelli derin öğrenme modeli yerleştirilmiştir. Bu çalışmalar da Python programlama dili ve Django web çatısı ile gerçekleştirilmiş, bir uzak sunucuda çalışması sağlanmıştır ve aktif olarak http://www.bhlhdb.org/ alan adına bağlı internet sitesinde kullanıma açıktır.

5.1. KULLANILAN VERİ SETİ

Veri seti oluşturulurken PlantTFDB'den alınan verilerin hazırlanmasıyla 132330 satırda 58 farklı TF protein sınıfının dizi ve sınıf bilgileri elde edilmişti. Bu diziler öncelikle kod sözlüğü yöntemiyle sayısallaştırıldıktan sonra 450 karakter uzunluğunda

olacak şekilde kırpılarak veya dizilerin sonundan itibaren eksik karakter sayısında 0 ile doldurularak hazır hale getirilmiştir. Şekil 5.1'de belirlenen uzunluktan uzun (Şekil 5.1 (a)) ve kısa (Şekil 5.1 (b)) olan diziler için birer kısaltılmış örnek ön işleme sonucu gösterilmiştir. Bu işlemler sonucunda protein dizileri sınıflandırma yapmak üzere modelin katmanlarına verilmeye hazır hale getirilmiştir.

```
Dizi         : MGKRKLKLELIKNNSTRKNCLRVRKGGLIK

Kod Sözlüğü : [11, 6, 9, 15, 9, 10, 9, 10, 4,
               10, 8, 9, 12, 12, 16, 17, 15,
               9, 12, 2, 10, 15, 18, 15, 9, 6,
               6, 10, 8, 9]

k-mer        : ['MGK', 'GKR', 'KRK', 'RKL', 'KLK',
               'LKL', 'KLE', 'LEL', 'ELI', 'LIK']

Vektör       : [1293, 108, 1, 64, 9876, 454, 12,
               3297, 564, 5]
```

(a)

```
Dizi         : MTEHKRRPASLEPAVSLSCGTRQK

Kod Sözlüğü : [11, 17, 4, 7, 9, 15, 15, 13, 1, 16,
               10, 4, 13, 1, 18, 16, 10, 16, 2, 6,
               17, 15, 14, 9, 0, 0, 0, 0, 0, 0]

k-mer        : ['MTE', 'HKR', 'RPA', 'SLE', 'PAV',
               'SLS', 'CGT', 'RQK']

Vektör       : [1001, 44, 123, 56, 908, 2, 13,
               502, 0, 0]
```

(b)

Şekil 5.1. Diziler için örnek ön işleme. a) uzun dizi, b) kısa dizi.

Şekil 5.1'deki ön işlemelerde "Dizi" dizinin ham halini, "Kod Sözlüğü" kod sözlüğü yöntemine göre sayısallaştırılmış halini, "k-mer" dizinin kelimelere bölünmüş halini ve "Vektör" de dizinin kelimelere bölündükten sonra Word2Vec vektör temsili yöntemi ile sayısallaştırılmış halini göstermektedir. (a) şeklinde bir dizinin tam veya fazla uzunlukta olması temsil edilmiştir. Görüldüğü gibi temsiller kod sözlüğü ve Word2Vec dönüşümlerinde tam olarak doludur, listelerde 0 yoktur. (b) şeklinde ise kısa diziler ele alınmış olup, kod sözlüğü kısmında eksik olan 6 amino asit yerine 0 doldurulması yapılmış, Word2Vec kısmında ise eksik olan 2 kelime vektörü yerine 0 doldurulması yapılmıştır. Bu sayede tüm diziler sayısallaştırılmış, aynı boyuta

getirilmiş ve modele vermeye hazırdır. Tüm bu işlemeler Python programlama dili ile hazırlanan fonksiyonlar ile yapılmış olup, fonksiyonlarda yer alan döngüler vasıtasıyla döngünün her bir adımında bir adet dizi işlenerek liste türündeki yeni değişkenine eklenerek hazırlanmış ve derin öğrenme modellerinin giriş katmanlarına verilmeye hazır hale getirilmiştir.

5.2. ÜRETİLEN DERİN ÖĞRENME MODELLERİ

Bu çalışmada kapsamında bitki TF proteinlerinin hangi TF protein ailesine ait olduğunu sınıflandırabilmek için yalnızca protein dizilerini kullanarak çalışan bir dizi derin öğrenme yöntemi üretilmiştir. Modellerin denemeleri hem kod sözlüğü yöntemi ile yapılan ön işleme adımı ile, hem de Word2Vec kelime temsili ile yapılan ön işleme adımı ile test edilmiştir. Her bir model birçok parametre değişikliği ile test edilerek her bir modelin en başarılı versiyonu belirlenerek kaydedilmiştir. Ayrıca her bir modelin en başarılı versiyonu 10 kat çapraz doğrulama yöntemi ile tekrar test edilmiş ve en başarılı model bu çalışma kapsamında önerilerek literatüre katkıda bulunulmuştur.

Tüm modellerin gerçeklenmesinde Python programlama dilinin TensorFlow ve Keras yapay zekâ kütüphaneleri ve bu kütüphanelerin alt modülleri kullanılmıştır. Veri setinde yer alan 58 farklı sınıfa ait 132330 adet dizi Python'un scikit-learn kütüphanesindeki train_test_split aracı ile %80 eğitim, %10 doğrulama ve %10 test veri seti olarak ayrılmıştır. Bu işlem sonrası 105864 dizi eğitim için, 13233 dizi doğrulama için ve yine 13233 dizi de test için ayrılmıştır. Bu ayırma işleminden sonra tüm modeller aynı ayrılmış veri seti parçaları üzerinde çalışmış, bu sayede tüm modeller veri seti örneklerinden bağımsız bir şekilde test edilebilmiştir. Modellerin geliştirilme sürecinde sürekli farklı veri örnekleriyle eğitim, doğrulama ve test yapılabilmesi için tüm modellerin baştan ona her bir çalışmasında veri seti tekrar ve rastgele ayrılma işlemine tabi tutulmuştur.

Örnek bir tek katmanlı çift yönlü LSTM modelinde Keras'ın gömme (embedding) katmanındaki "mask_zero" ve "trainable" parametreleri aktif (True) ve pasif (False) olarak test edilmiş ve bu parametrelerden "mask_zero" parametresinin pasif ve "trainable" parametresinin aktif olduğu durum en başarılı sonucu vermiştir. Bu

sebepten de tüm modellerde “mask_zero” parametresi pasif ve “trainable” parametresi aktif olarak kullanılmıştır. Burada “mask_zero” parametresi veri setindeki 0’ları yok sayma işlemini ve “trainable” parametresi de modelin kendi içinde her bir epochta eğitilebilir olması işlemini gerçekleştirmektedir. Yine gömme katmanında yer alan başlangıç ağırlıkları parametresine (weights) Word2Vec modelinin üretmiş olduğu ağırlıklar verilerek modelin eğitiminin daha yüksek başarılı seviyeden başlaması sağlanmıştır. Nitekim bu işlemin başarısı Çizelge 4.3’te ifade edilmiştir. Tüm modellerde optimizasyon fonksiyonu olarak ADAM en başarılı sonuçları vermiştir ve tüm modellerde kullanılmıştır. Ayrıca tüm modeller her bir katmandaki nöron sayısı, batch boyutu ve epoch değerleri ile eğitilmiş, eğitim, doğrulama ve test kısımlarında en düşük hataya ve en yüksek başarıya sahip olanlar seçilmiş, hem tekli çalışma sonuçlarıyla, hem de 10 kat çapraz doğrulama sonuçlarıyla kıyaslanmış ve önerilen derin öğrenme modeli belirlenmiştir.

5.2.1. Kod Sözlüğü Ön İşlemeli Modeller

Eğitim, doğrulama ve test olarak üç parçaya ayrılan veri seti, Bölüm 4.5.1’de detayları ile anlatılmış olan kod sözlüğü ön işleme yöntemi ile sayısallaştırılmış ve ön işleme süreci tamamlanmıştır.

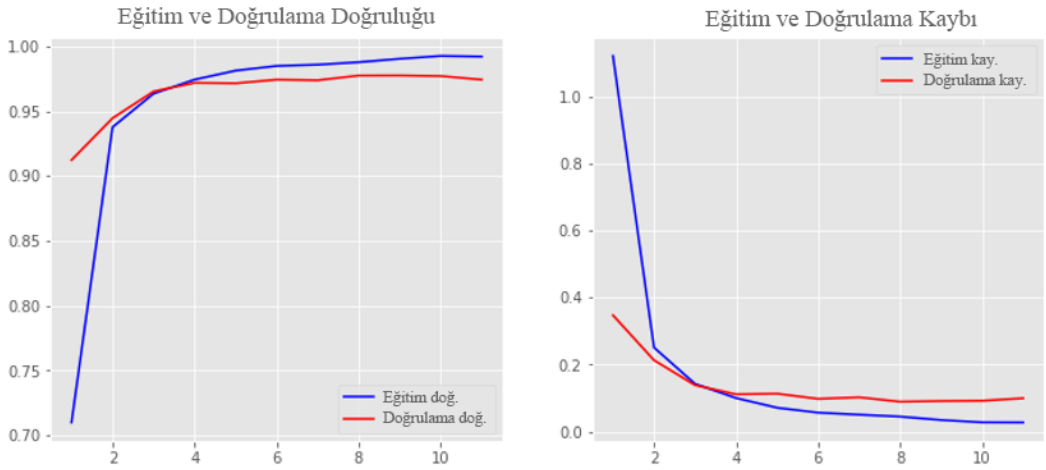
5.2.1.1. LSTM Modelleri

Kod sözlüğü ile yapılan ön işleme adımından sonra hazırlanan sabit 450 uzunluğa sahip diziler önce gömme katmanına, ardından her biri 256 nörona sahip iki katmanlı çift yönlü LSTM modeline verilmiş ve tam bağlı sınıflandırma katmanından sonra sonuç elde edilmiştir. Burada tam bağlı katmanda aktivasyon fonksiyonu olarak softmax, derleme kısmında optimizasyon fonksiyonu olarak ADAM, kayıp fonksiyonu olarak kategorik çapraz entropi seçilmiş, batch boyutu 256 ve epoch değeri de 50 olarak belirlenmiştir. Ayrıca modele bir erken durdurma aracı eklenerek en başarılı durumda modelin sonlandırılarak kaydedilmesi amaçlanmıştır. Modelin yapısı Çizelge 5.1’de verilmiştir.

Çizelge 5.1. Çift katmanlı, çift yönlü LSTM modelinin yapısı.

Katman	Çıkış Boyutu	Param #
Embedding	(None, 450, 300)	2784600
Bidirectional	(None, 450, 512)	1140736
Bidirectional	(None, 512)	1574912
Dense	(None, 58)	29754

Modelin eğitim başarısı %98.78 ve doğrulama başarısı %97.32'dir. Modelin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.2'de verilmiştir.



(a) (b)

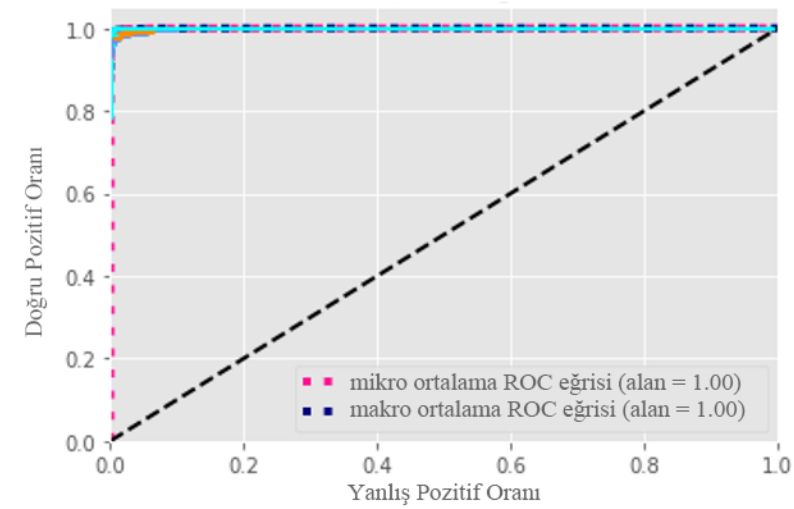
Şekil 5.2. Çift katmanlı, çift yönlü LSTM modelinin eğitim ve doğrulama sonuç grafikleri.

Eğitimini 13 epochta tamamlayan bu modelin test doğruluk, kesinlik, hassasiyet ve f-skor değerleri Çizelge 5.2'de verilmiştir.

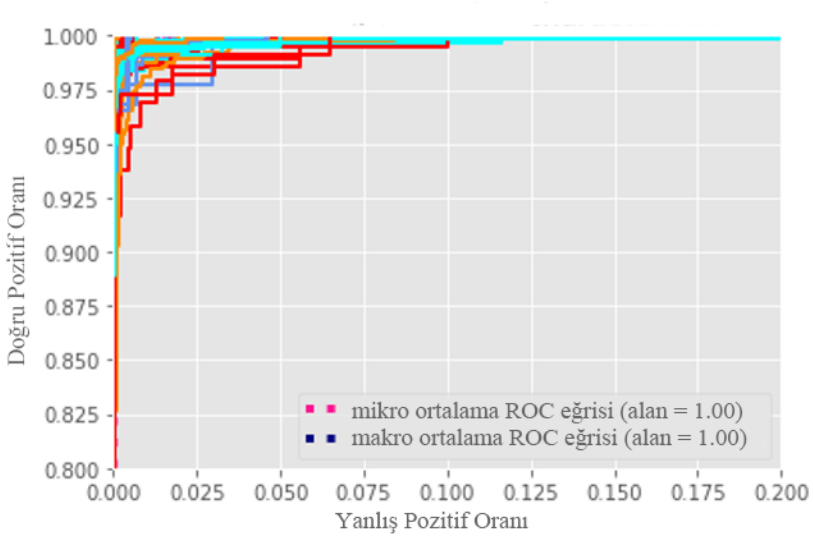
Çizelge 5.2. Çift katmanlı, çift yönlü LSTM modelinin test sonuçları.

Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F-Skor (%)
Çift Katmanlı Çift Yönlü LSTM	97.54	95.42	95.53	95.32

Çizelge 5.2'ye göre 58 farklı sınıfı da %97.54 doğruluk oranı ile başarılı bir şekilde sınıflandıran modelin ROC grafikleri Şekil 5.3'te verilmiştir.



(a) ROC grafiği.



(b) ROC grafiğinin yakınlaştırılmış hali.

Şekil 5.3. Çift katmanlı, çift yönlü LSTM modelinin a) ROC grafiği, b) ROC grafiğinin yakınlaştırılmış hali.

Şekil 5.3'ün (a) şeklinde de görüldüğü gibi model 58 sınıfın tümünü de oldukça başarılı bir şekilde sınıflandırmıştır. 58 sınıf için ayrı ayrı olan 58 eğrinin de 1.0 değerine çok yakın olduğu, modelin tüm sınıfları başarılı bir şekilde sınıflandırdığının göstergesidir. (b) şeklinde verilen, (a) şeklindeki grafiğin yakınlaştırılmış versiyonunda her bir sınıfın eğrileri daha rahat bir şekilde görünmektedir.

Bu model, hazırlanmış olan yeni nesil bHLH TF protein veritabanında derin öğrenme analiz aracı olarak kullanılmış ve literatürdeki benzer örneklerinin önüne geçerek literatüre katkı sağlamıştır [117].

5.2.2. k-mer ve Word2Vec Ön İşlemeli Modeller

Eğitim, doğrulama ve test olarak üç parçaya ayrılan veri seti, bölüm 4.5.3'te detayları ile anlatılmış olan k-mer ve Word2Vec yöntemi ile vektörize edilerek sayısallaştırılmış ve ön işleme süreci tamamlanmıştır. Bölüm 4.5.3'te belirtildiği gibi Word2Vec CBOW modelinde pencere boyutu (window size) parametresi, varsayılan "w = 5" değeri ile ve "w = 4" değeri ile test edilmiş ve "w = 4" pencere değerinin daha başarılı olduğu yapılan testlerin sonucunda açıkça görülmüştür. Bu çalışma kapsamında hazırlanan 9 farklı model hem 4, hem de 5 pencere boyutu ile çalıştırılmış ve modelleri farklı parametrelerle çalışan versiyonlarının büyük bir çoğunluğu 4 pencere boyutlu Word2Vec vektörleri ile daha yüksek başarı göstermiştir. Çizelge 5.3'te örnek bir tek katlı çift yönlü LSTM modelinde Word2Vec pencere boyutuna göre model başarıları verilmiştir.

Çizelge 5.3. Çift katmanlı, çift yönlü LSTM modelinin Word2Vec pencere boyutuna göre sonuçları.

Model	Word2Vec Pencere Boyutu	Eğitim (%)	Doğrulama (%)	Test (%)
Tek Katmanlı Çift Yönlü LSTM	4	99.71	97.29	97.45
	5	99.63	97.16	97.43

Çizelge 5.3'teki sonuçlar ışığında çalışmadaki tüm k-mer ve Word2Vec ön işlemeli modellerde Word2Vec pencere boyutu "w = 4" olarak seçilmiştir.

5.2.2.1. LSTM Modelleri

k-mer ve Word2Vec ile yapılan ön işleme adımından sonra hazırlanan sabit 250 uzunluğa sahip diziler önce gömme katmanına, ardından tek katmanlı çift yönlü LSTM ve çift katmanlı çift yönlü LSTM modellerine verilmiş ve tam bağlı sınıflandırma katmanından sonra sonuçlar elde edilmiştir.

Tek Katmanlı Çift Yönlü LSTM Modelleri

Tek katmanlı, çift yönlü LSTM modelleri bir adet gömme katmanı, bir adet çift yönlü LSTM katmanı ve bir adet de tam bağlı katman ile oluşturulmuş ve eğitim gerçekleştirilmiştir. Modellerin yapısı Çizelge 5.4'te verilmiştir.

Çizelge 5.4. Tek katmanlı, çift yönlü LSTM modellerinin yapıları.

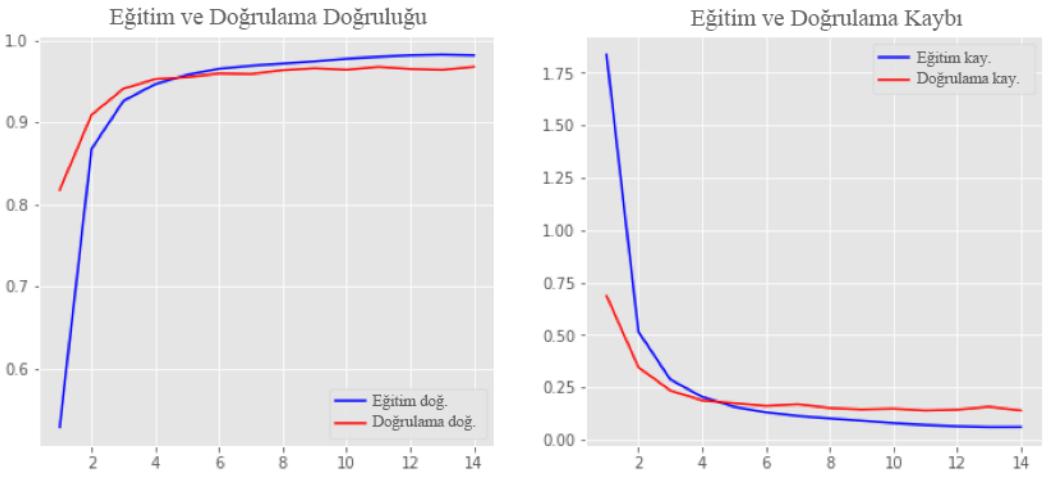
Model	**Katman**	**Model Kodu**	**LSTM**	**Batch Boyutu**
Tek Katmanlı Çift Yönlü LSTM	Embedding, Bidirectional, Dense (sınıflandırma)	M-1-1	128	128
		M-1-2	128	256
		M-1-3	256	128
		M-1-4	256	256

Çizelge 5.4'teki parametrelere göre eğitilen modellerin sonuçları Çizelge 5.5'te verilmiştir.

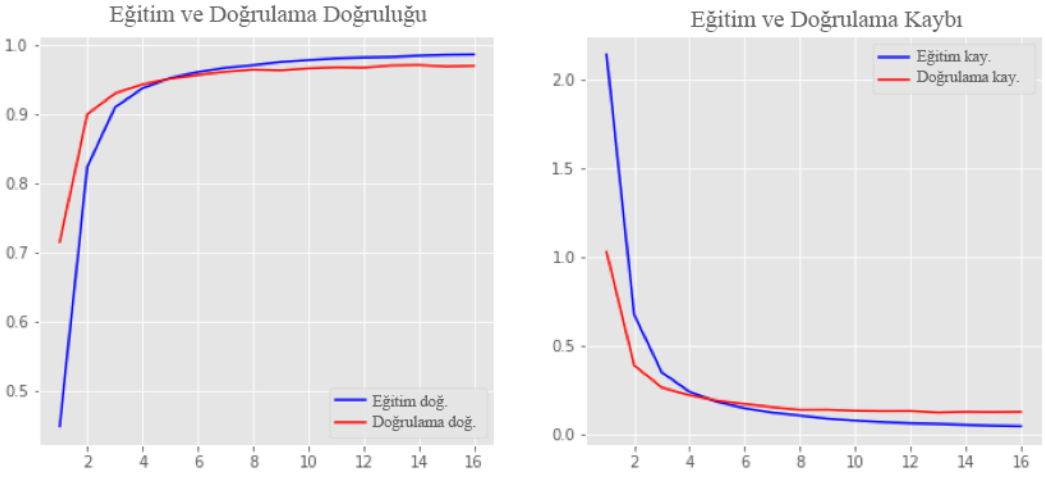
Çizelge 5.5. Tek katmanlı, çift yönlü LSTM modellerinin sonuçları.

Model Kodu	**Epoch**	**Eğitim (%)**	**Doğrulama (%)**	**Test (%)**
M-1-1	14	99.44	96.74	97.02
M-1-2	16	99.68	97.05	97.36
M-1-3	**10**	**99.71**	**97.29**	**97.42**
M-1-4	8	99.22	96.85	97.06

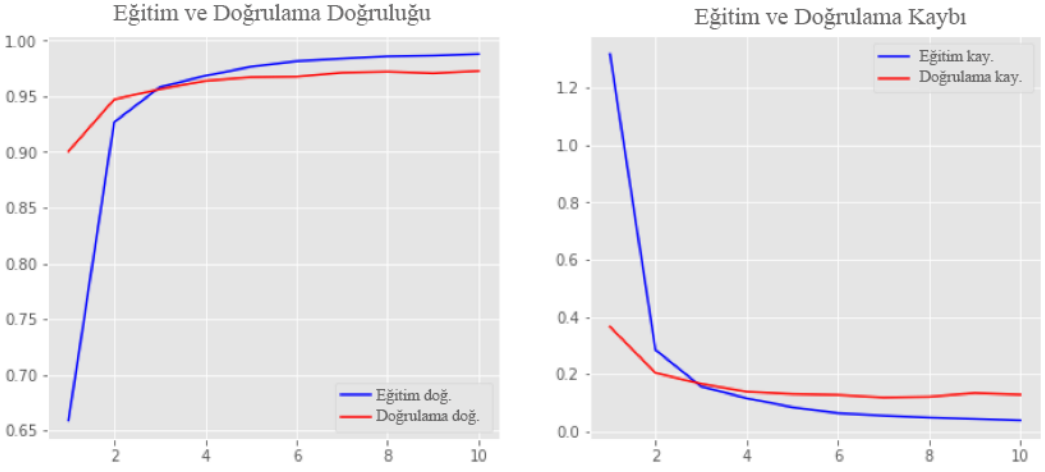
Modellerin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.4'te verilmiştir.



(a) M-1-1 modelinin eğitim ve doğrulama sonuç grafikleri.

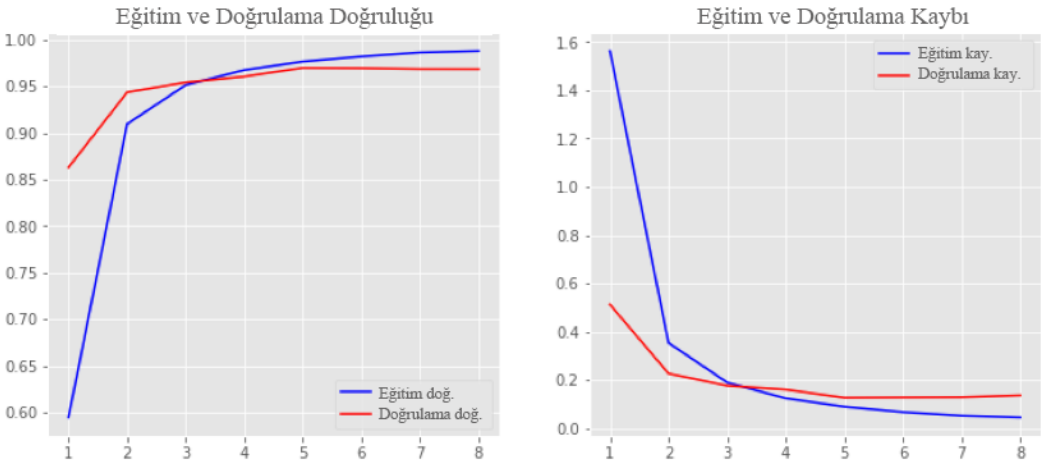


(b) M-1-2 modelinin eğitim ve doğrulama sonuç grafikleri.



(c) M-1-3 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.4. Tek katmanlı, çift yönlü LSTM modellerinin eğitim ve doğrulama sonuç grafikleri. Model kodları a) M-1-1, b) M-1-2, c) M-1-3, d) M-1-4.



(d) M-1-4 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.4. (Devam ediyor).

Gerçekleştirilen eğitim, doğrulama ve test sürecinden sonra M-1-3 kodlu model tek katmanlı çift yönlü LSTM modelleri içerisinde en başarılı model olarak belirlenmiştir.

Çift Katmanlı Çift Yönlü LSTM Modelleri

Çift katmanlı, çift yönlü LSTM modelleri bir adet gömme katmanı, iki adet çift yönlü LSTM katmanı ve bir adet de tam bağlı katman ile oluşturulmuş ve eğitim gerçekleştirilmiştir. Modellerin yapısı Çizelge 5.6'da verilmiştir.

Çizelge 5.6. Çift katmanlı, çift yönlü LSTM modellerinin yapıları.

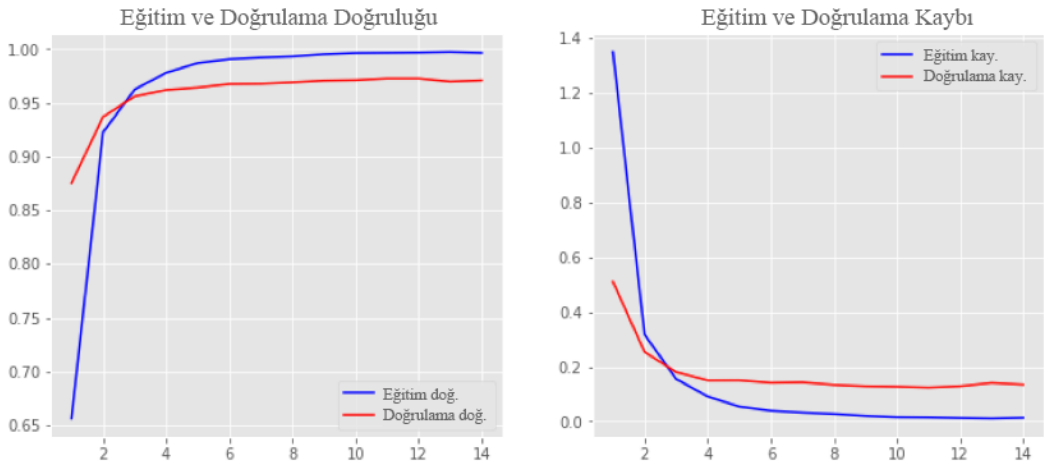
Model	Katman	Model Kodu	LSTM 1-2	Batch Boyutu
Çift Katmanlı Çift Yönlü LSTM	Embedding, Bidirectional, Bidirectional, Dense (sınıflandırma)	M-2-1	128	256
			128	
		M-2-2	128	256
			256	
		M-2-3	256	256
			128	
		M-2-4	256	256
			256	

Çizelge 5.6'daki parametrelere göre eğitilen modellerin sonuçları Çizelge 5.7'de verilmiştir.

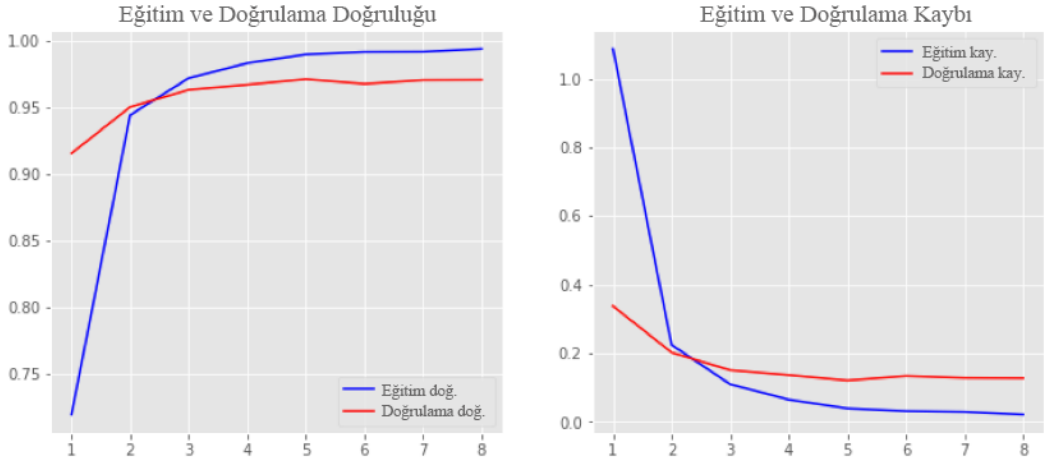
Çizelge 5.7. Çift katmanlı, çift yönlü LSTM modellerinin sonuçları.

Model Kodu	Epoch	Eğitim (%)	Doğrulama (%)	Test (%)
M-2-1	14	99.82	97.08	97.43
M-2-2	8	99.69	97.08	97.30
M-2-3	9	99.45	96.80	97.26
M-2-4	**12**	**99.79**	**97.51**	**97.72**

Modellerin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.5'te verilmiştir.

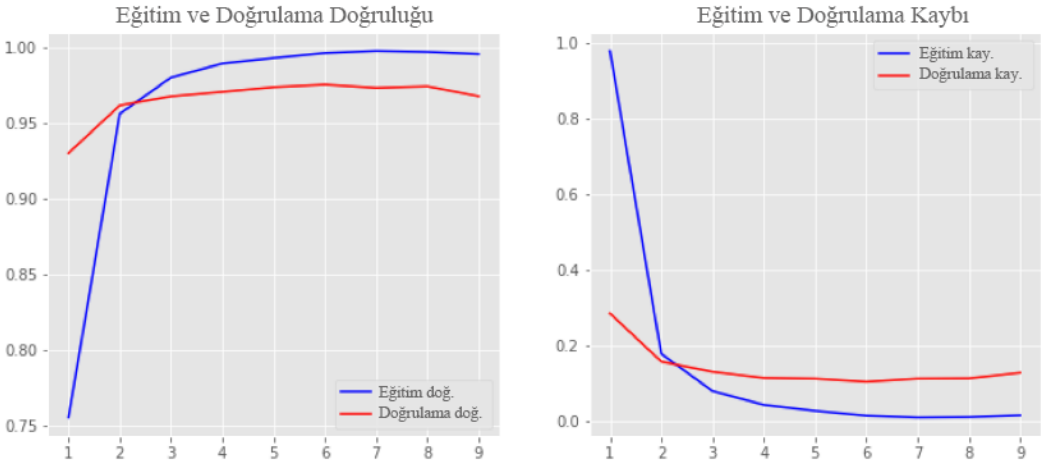


(a) M-2-1 modelinin eğitim ve doğrulama sonuç grafikleri.

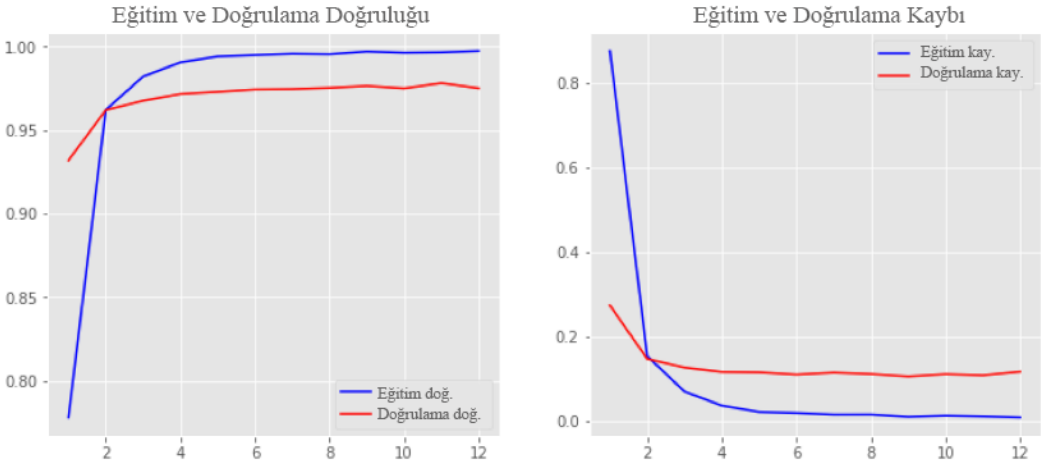


(b) M-2-2 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.5. Çift katmanlı, çift yönlü LSTM modellerinin eğitim ve doğrulama sonuç grafikleri. Model kodları a) M-2-1, b) M-2-2, c) M-2-3, d) M-2-4.



(c) M-2-3 modelinin eğitim ve doğrulama sonuç grafikleri.



(d) M-2-4 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.5. (Devam ediyor).

Gerçekleştirilen eğitim, doğrulama ve test sürecinden sonra M-2-4 kodlu model çift katmanlı çift yönlü LSTM modelleri içerisinde en başarılı model olarak belirlenmiştir.

5.2.2.2. GRU Modelleri

k-mer ve Word2Vec ile yapılan ön işleme adımından sonra hazırlanan sabit 250 uzunluğa sahip diziler önce gömme katmanına, ardından tek katmanlı çift yönlü GRU ve çift katmanlı çift yönlü GRU modellerine verilmiş ve tam bağlı sınıflandırma katmanından sonra sonuçlar elde edilmiştir.

Tek Katmanlı Çift Yönlü GRU Modelleri

Tek katmanlı, çift yönlü GRU modelleri bir adet gömme katmanı, bir adet çift yönlü GRU katmanı ve bir adet de tam bağlı katman ile oluşturulmuş ve eğitim gerçekleştirilmiştir. Modellerin yapısı Çizelge 5.8'te verilmiştir.

Çizelge 5.8. Tek katmanlı, çift yönlü GRU modellerinin yapıları.

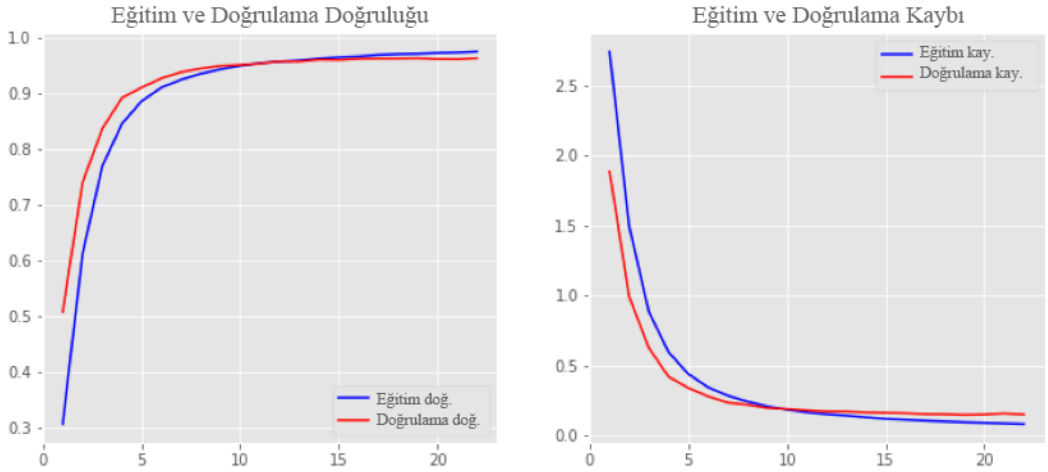
Model	Katman	Model Kodu	GRU	Batch Boyutu
Tek Katmanlı Çift Yönlü GRU	Embedding, Bidirectional, Dense (sınıflandırma)	M-3-1	64	128
		M-3-2	64	256
		M-3-3	128	128
		M-3-4	128	256
		M-3-5	256	128
		M-3-6	256	256

Çizelge 5.8'teki parametrelere göre eğitilen modellerin sonuçları Çizelge 5.9'te verilmiştir.

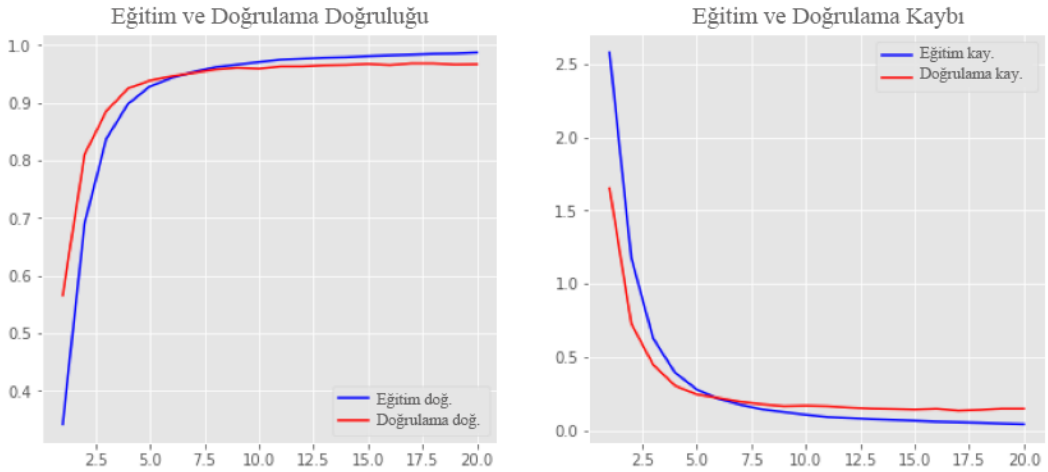
Çizelge 5.9. Tek katmanlı, çift yönlü GRU modellerinin sonuçları.

Model Kodu	Epoch	Eğitim (%)	Doğrulama (%)	Test (%)
M-3-1	22	99.45	96.39	96.73
M-3-2	20	99.71	96.71	97.15
M-3-3	18	99.64	96.88	96.98
M-3-4	14	99.36	96.53	96.60
M-3-5	11	99.74	97.13	97.29
M-3-6	**13**	**99.78**	**97.32**	**97.39**

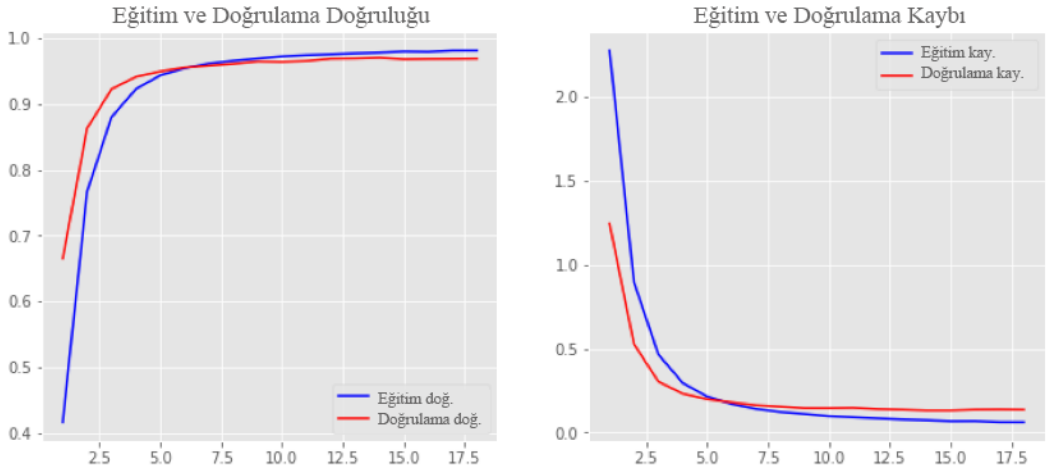
Modellerin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.6'da verilmiştir.



(a) M-3-1 modelinin eğitim ve doğrulama sonuç grafikleri.

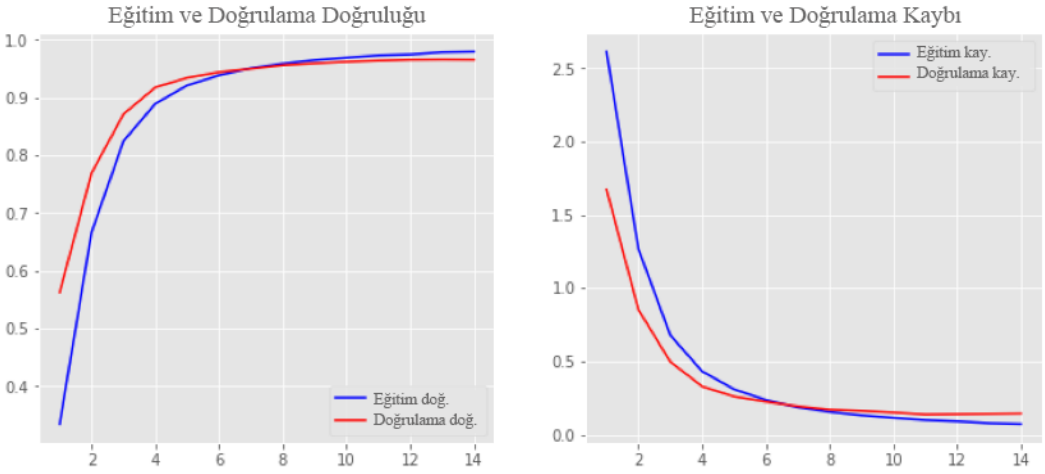


(b) M-3-2 modelinin eğitim ve doğrulama sonuç grafikleri.

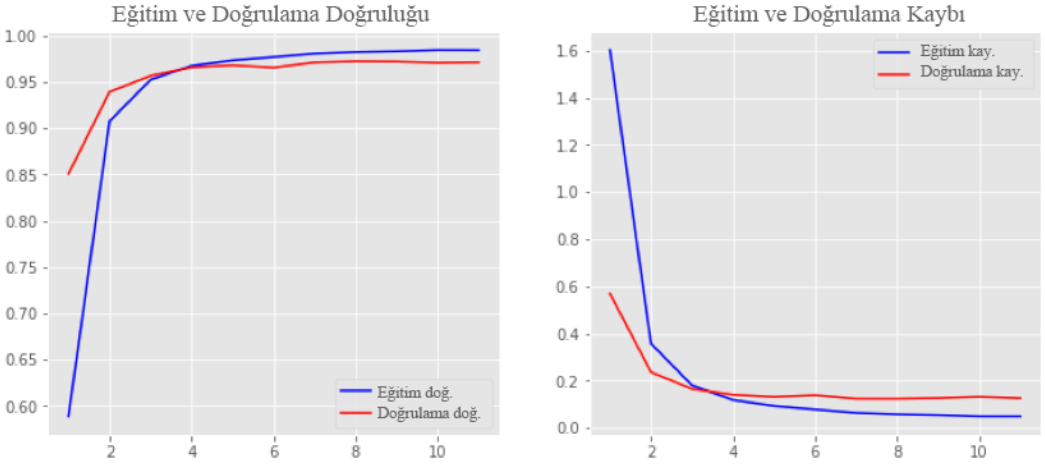


(c) M-3-3 modelinin eğitim ve doğrulama sonuç grafikleri.

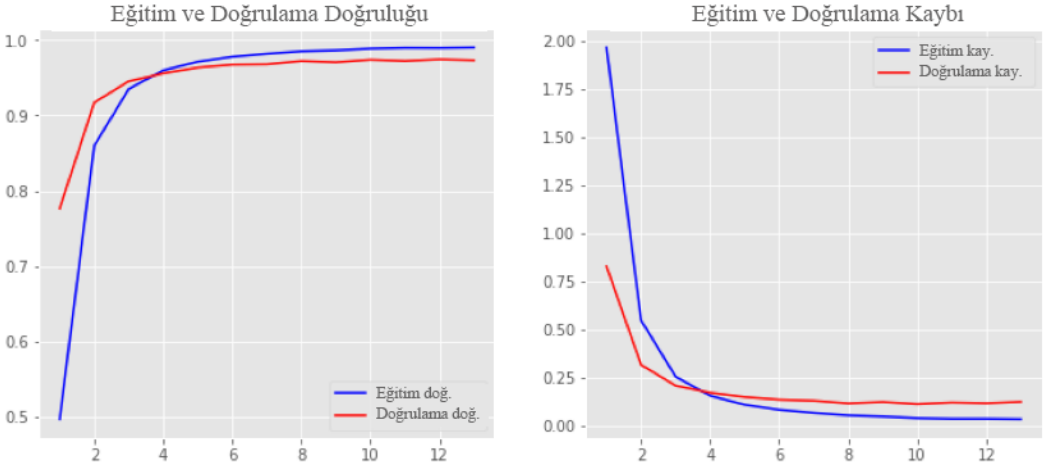
Şekil 5.6. Tek katmanlı, çift yönlü GRU modellerinin eğitim ve doğrulama sonuç grafikleri. Model kodları a) M-3-1, b) M-3-2, c) M-3-3, d) M-3-4, e) M-3-5, f) M-3-6.



(d) M-3-4 modelinin eğitim ve doğrulama sonuç grafikleri.



(e) M-3-5 modelinin eğitim ve doğrulama sonuç grafikleri.



(f) M-3-6 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.6. (Devam ediyor).

Gerçekleştirilen eğitim, doğrulama ve test sürecinden sonra M-3-6 kodlu model tek katmanlı çift yönlü GRU modelleri içerisinde en başarılı model olarak belirlenmiştir.

Çift Katmanlı Çift Yönlü GRU Modelleri

Çift katmanlı, çift yönlü GRU modelleri bir adet gömme katmanı, iki adet çift yönlü GRU katmanı ve bir adet de tam bağlı katman ile oluşturulmuş ve eğitim gerçekleştirilmiştir. Modellerin yapısı Çizelge 5.10'da verilmiştir.

Çizelge 5.10. Çift katmanlı, çift yönlü GRU modellerinin yapıları.

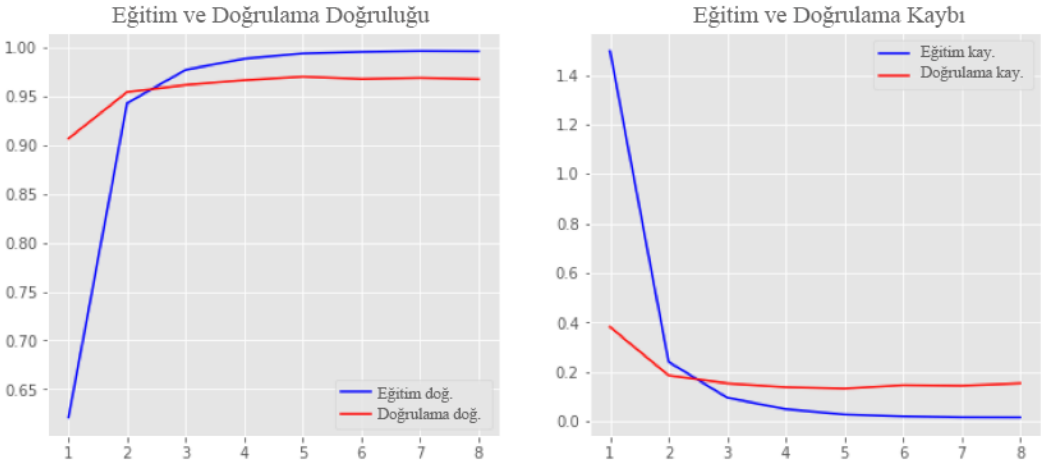
Model	Katman	Model Kodu	GRU	Batch Boyutu
Çift Katmanlı Çift Yönlü GRU	Embedding, Bidirectional, Bidirectional, Dense (sınıflandırma)	M-4-1	128	256
			128	
		M-4-2	128	256
			256	
		M-4-3	256	256
			128	
		M-4-4	256	256
			256	

Çizelge 5.10'daki parametrelere göre eğitilen modellerin sonuçları Çizelge 5.11'de verilmiştir.

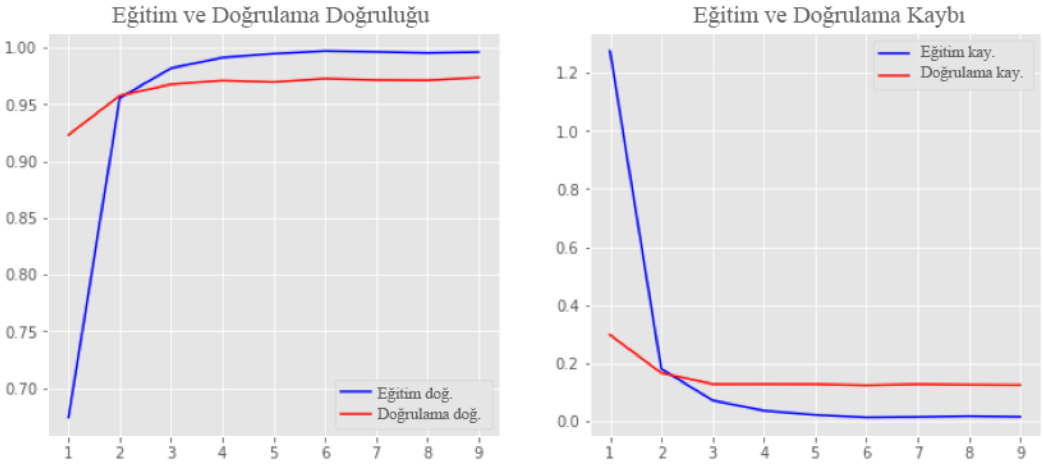
Çizelge 5.11. Çift katmanlı, çift yönlü GRU modellerinin sonuçları.

Model Kodu	Epoch	Eğitim (%)	Doğrulama (%)	Test (%)
M-4-1	8	99.77	96.75	96.83
M-4-2	9	99.82	97.38	97.49
M-4-3	6	99.66	96.86	96.85
M-4-4	**7**	**99.76**	**97.38**	**97.54**

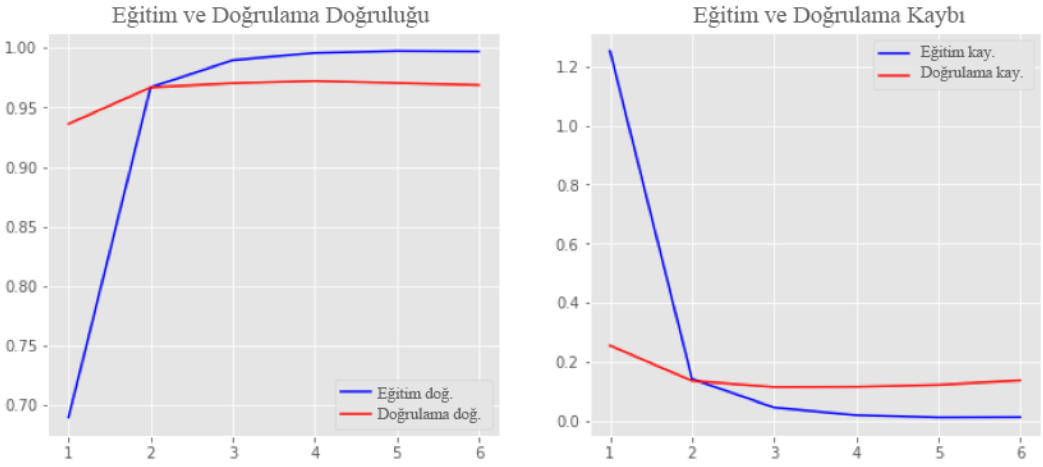
Modellerin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.7'de verilmiştir.



(a) M-4-1 modelinin eğitim ve doğrulama sonuç grafikleri.

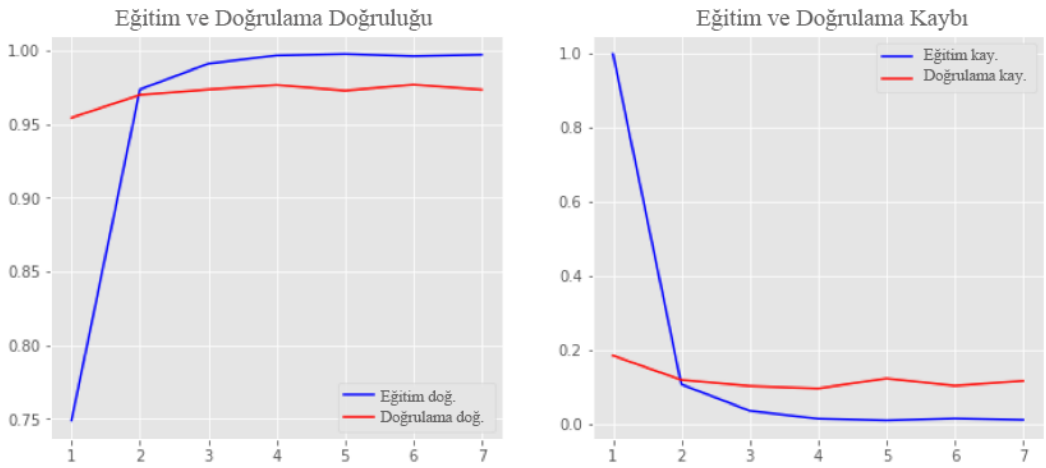


(b) M-4-2 modelinin eğitim ve doğrulama sonuç grafikleri.



(c) M-4-3 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.7. Çift katmanlı, çift yönlü GRU modellerinin eğitim ve doğrulama sonuç grafikleri. Model kodları a) M-4-1, b) M-4-2, c) M-4-3, d) M-4-4.



(d) M-4-4 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.7. (Devam ediyor).

Gerçekleştirilen eğitim, doğrulama ve test sürecinden sonra M-4-4 kodlu model çift katmanlı çift yönlü GRU modelleri içerisinde en başarılı model olarak belirlenmiştir.

5.2.2.3. CNN Modelleri

Görüntüleri çok iyi sınıflandıran evrişimli sinir ağı modellerinin başarısı artık doğal dil işlemeye dayalı çalışmalarda da kullanılmaktadır [121]. Burada da CNN mimarisinin öznitelik çıkarmadaki başarısından yararlanmak için k-mer ve Word2Vec ile yapılan ön işleme adımından sonra hazırlanan sabit 250 uzunluğa sahip diziler önce gömme katmanına, ardından üç evrişim ve üç maksimum havuzlama katmanına, seyreltme katmanına ve bir tam bağlı sınıflandırma katmanına verilerek modeller oluşturulmuş ve eğitim gerçekleştirilmiştir. Modellerin yapısı Çizelge 5.12'de verilmiştir.

Çizelge 5.12. CNN modellerinin yapıları.

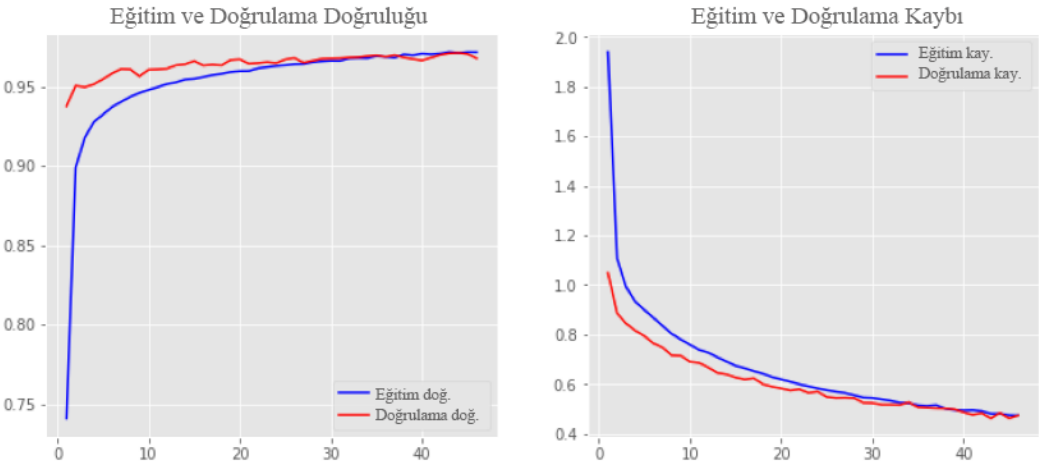
Model	Katman	Model Kodu	CONV	Batch Boyutu
CNN	Embedding, Conv2D x 3, MaxPool2D x 3, Dropout, Dense (sınıflandırma)	M-5-1	128	128
		M-5-2	128	256
		M-5-3	256	128
		M-5-4	256	256

Çizelge 5.12'deki parametrelere göre eğitilen modellerin sonuçları Çizelge 5.13'te verilmiştir.

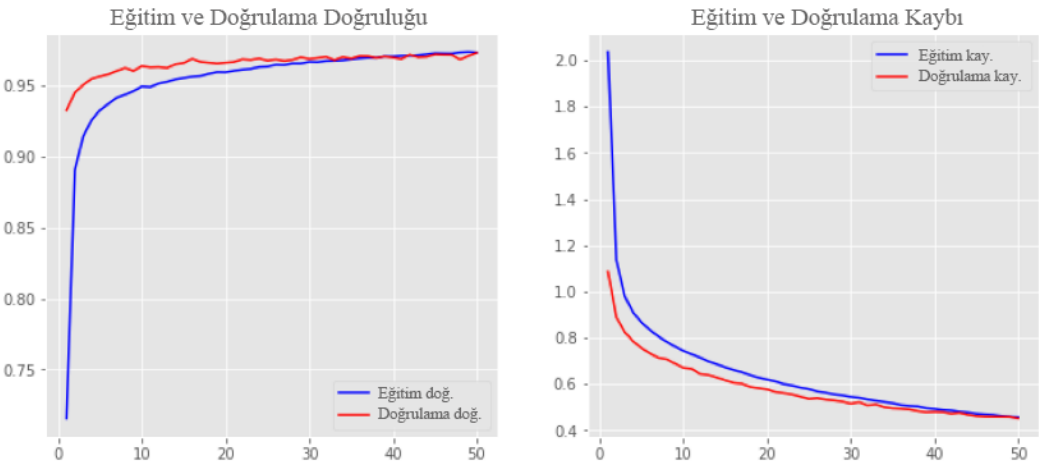
Çizelge 5.13. CNN modellerinin sonuçları.

Model Kodu	Epoch	Eğitim (%)	Doğrulama (%)	Test (%)
M-5-1	46	98.20	96.77	97.09
M-5-2	**50**	**98.45**	**97.28**	**97.33**
M-5-3	45	98.28	96.82	96.99
M-5-4	50	98.42	97.19	97.20

Modellerin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.8'de verilmiştir.

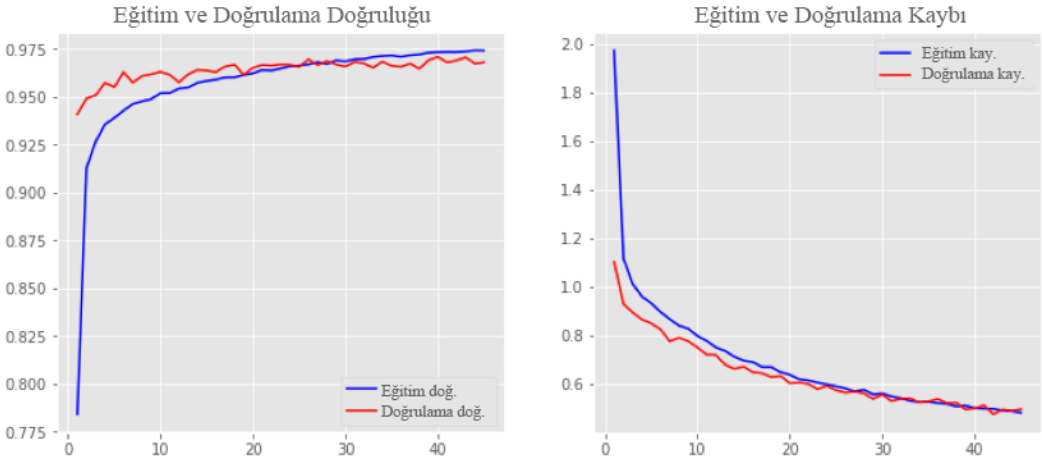


(a) M-5-1 modelinin eğitim ve doğrulama sonuç grafikleri.

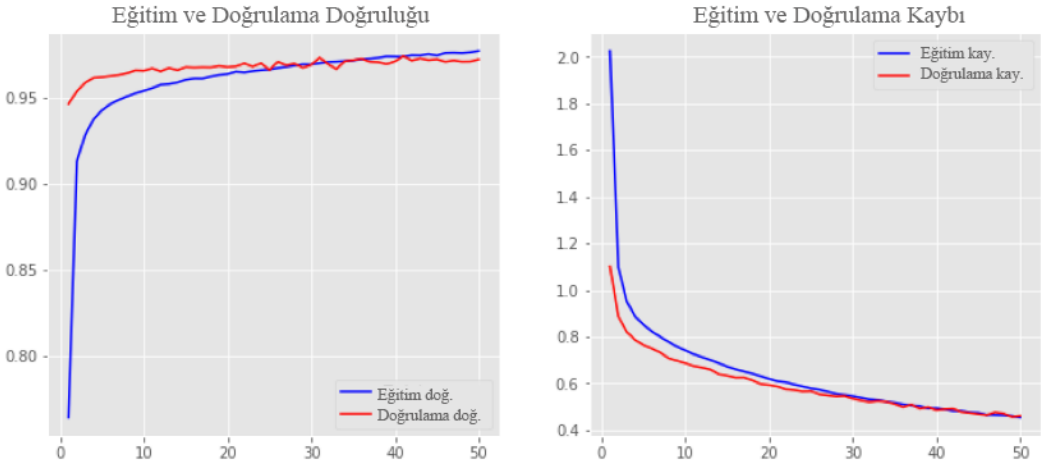


(b) M-5-2 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.8. CNN modellerinin eğitim ve doğrulama sonuç grafikleri. Model kodları a) M-5-1, b) M-5-2, c) M-5-3, d) M-5-4.



(c) M-5-3 modelinin eğitim ve doğrulama sonuç grafikleri.



(d) M-5-4 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.8. (Devam ediyor).

Gerçekleştirilen eğitim, doğrulama ve test sürecinden sonra M-5-2 kodlu model CNN modelleri içerisinde en başarılı model olarak belirlenmiştir.

5.2.2.4. Hibrit Derin Öğrenme Modelleri

Yukarıda bahsedildiği gibi, öznitelik çıkarmada CNN mimarilerinin başarısı açıktır. Ayrıca LSTM, GRU ve benzeri mimarilerle uzun süreli bağımlılıklardaki başarı klasik RNN mimarisine göre oldukça yüksektir [122]. Bu iki farklı mimarinin birlikte çalışması ve eğitim öncesi bir Word2Vec modelinden kelimelerin yakınlığını ifade eden ağırlıkları alması ile birlikte bu üçlüden kaçınılmaz olarak yüksek bir başarı elde

edilecektir. Bu sebepten de bu çalışma kapsamında farklı hibrit modeller tasarlanmış ve eğitilmiştir. Bu modellerde CNN mimarisinin öznitelik çıkarmadaki başarısından yararlanmak için k-mer ve Word2Vec ile yapılan ön işleme adımından sonra hazırlanan sabit 250 uzunluğa sahip diziler önce gömme katmanına, ardından üç adet evrişim, maksimum havuzlama ve seyreltme katmanına, bir LSTM/GRU (tek ve çift yönlü) katmanına, bir adet seyreltme katmanına, bir adet tam bağlı katmana ve bir adet de tam bağlı sınıflandırma katmanına verilerek hibrit modeller oluşturulmuş ve eğitim gerçekleştirilmiştir. Ayrıca modellerde LSTM yerine LSTM kadar başarılı olan GRU mimarisinin kullanılması, hücrelerinin tasarımı açısından daha hızlı sonuç verebilmektedir [79]. Bu sayede hem yüksek başarı hem de daha hızlı sonuçlar amaçlanmaktadır.

CNN LSTM Modelleri

CNN LSTM modelleri bir adet gömme katmanı, üç adet evrişim, maksimum havuzlama ve seyreltme katmanı, bir adet LSTM katmanı, bir adet seyreltme katmanı, bir adet tam bağlı katman ve bir adet de tam bağlı sınıflandırma katmanı ile oluşturulmuş ve eğitim gerçekleştirilmiştir. Modellerin yapısı Çizelge 5.14'te verilmiştir.

Çizelge 5.14. CNN LSTM modellerinin yapıları.

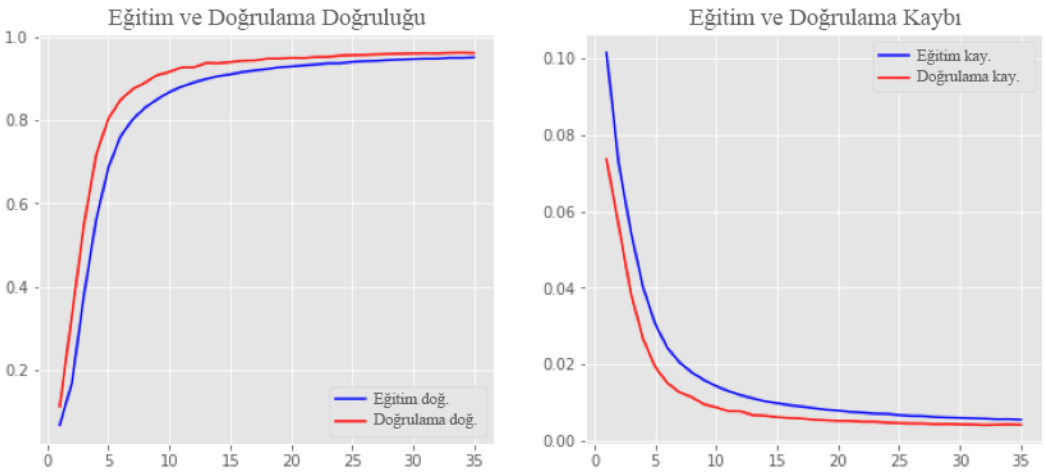
Model	Katman	Model Kodu	CONV1	CONV2	CONV3	LSTM	Dense	Batch Boyutu
CNN LSTM	Embedding, (Conv1D, MaxPool1D, Dropout) x 3, LSTM, Dropout, Dense, Dense (sınıflandırma)	M-6-1	32	64	128	64	128	256
		M-6-2	32	64	128	128	128	256
		M-6-3	64	128	256	128	128	256
		M-6-4	64	128	256	256	128	256
		M-6-5	64	128	256	256	256	256

Çizelge 5.14'teki parametrelere göre eğitilen modellerin sonuçları Çizelge 5.15'te verilmiştir.

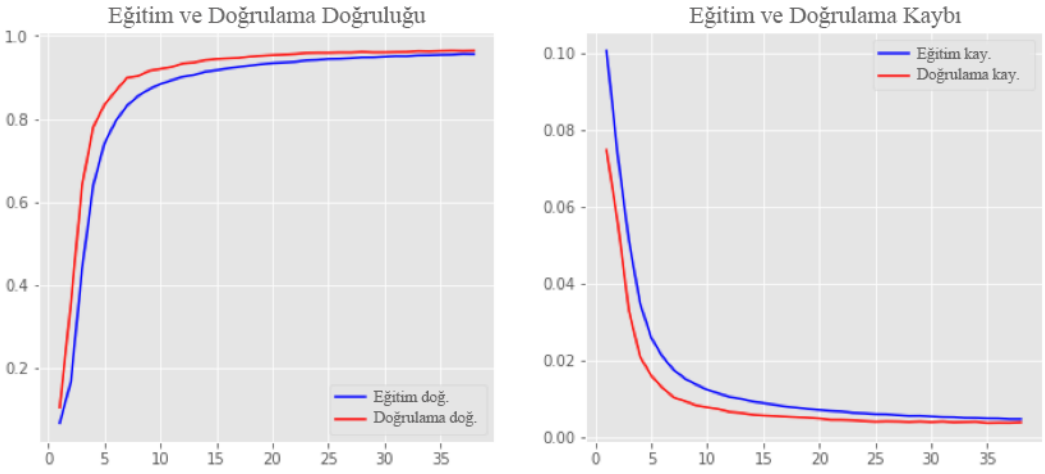
Çizelge 5.15. CNN LSTM modellerinin sonuçları.

Model Kodu	Epoch	Eğitim (%)	Doğrulama (%)	Test (%)
M-6-1	35	98.19	96.10	96.46
M-6-2	38	98.53	96.40	96.68
M-6-3	**29**	**99.12**	**97.48**	**97.83**
M-6-4	24	98.95	97.31	97.57
M-6-5	21	99.03	97.39	97.72

Modellerin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.9'da verilmiştir.

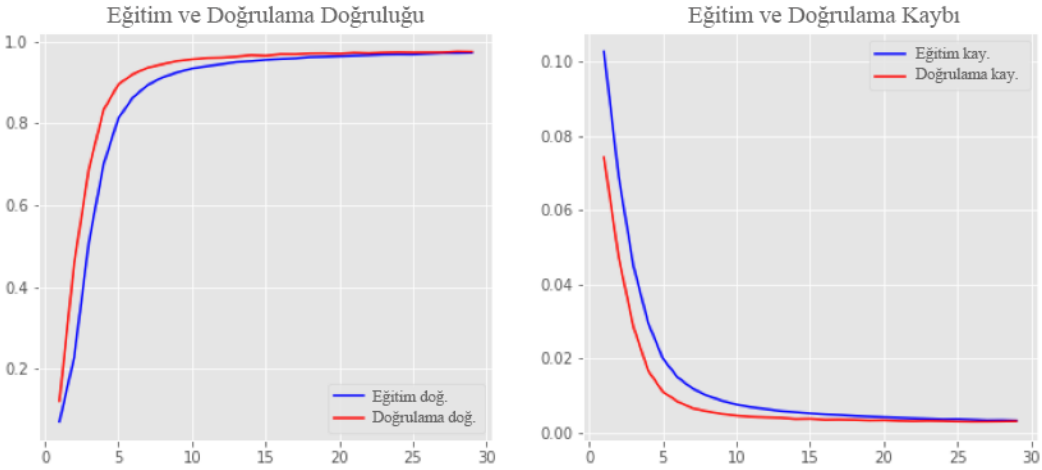


(a) M-6-1 modelinin eğitim ve doğrulama sonuç grafikleri.

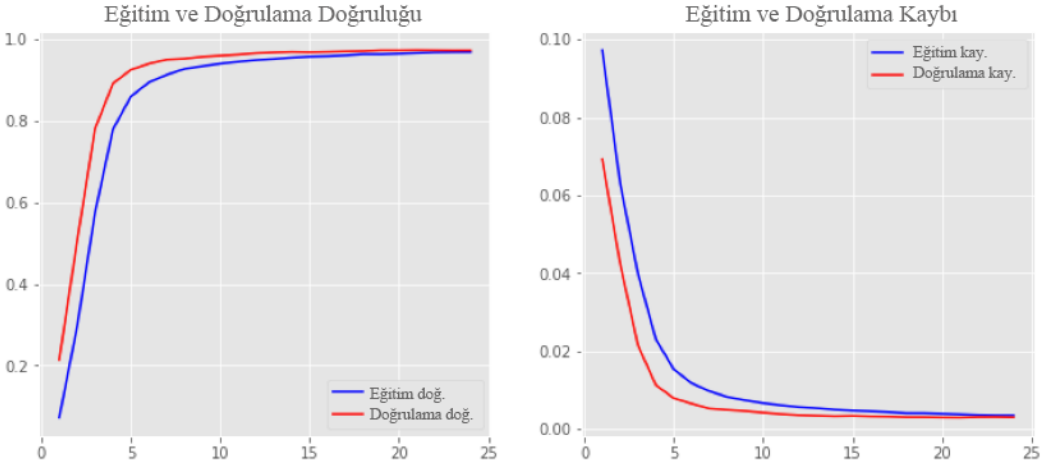


(b) M-6-2 modelinin eğitim ve doğrulama sonuç grafikleri.

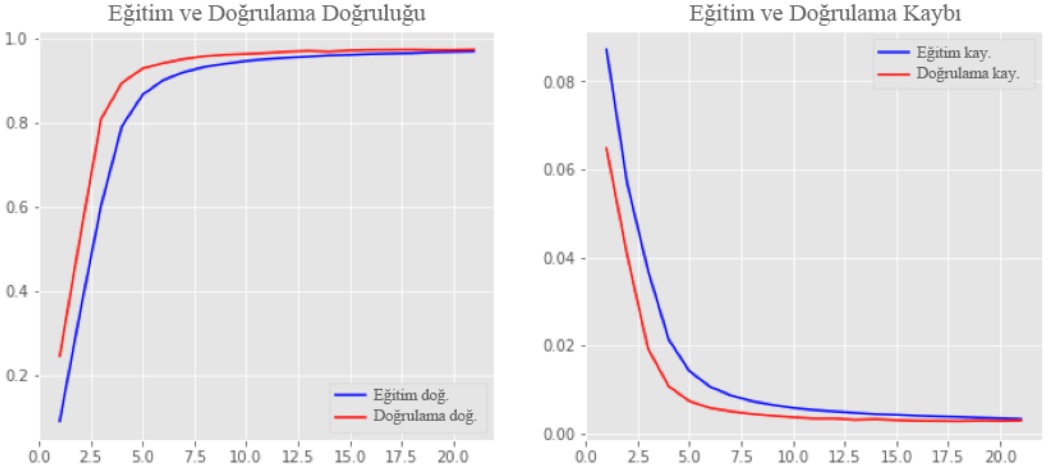
Şekil 5.9. CNN LSTM modellerinin eğitim ve doğrulama sonuç grafikleri. Model kodları a) M-6-1, b) M-6-2, c) M-6-3, d) M-6-4, e) M-6-5.



(c) M-6-3 modelinin eğitim ve doğrulama sonuç grafikleri.



(d) M-6-4 modelinin eğitim ve doğrulama sonuç grafikleri.



(e) M-6-5 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.9. (Devam ediyor).

Gerçekleştirilen eğitim, doğrulama ve test sürecinden sonra M-6-3 kodlu model CNN LSTM modelleri içerisinde en başarılı model olarak belirlenmiştir.

CNN GRU Modelleri

CNN GRU modelleri bir adet gömme katmanı, üç adet evrişim, maksimum havuzlama ve seyreltme katmanı, bir adet GRU katmanı, bir adet seyreltme katmanı, bir adet tam bağlı katman ve bir adet de tam bağlı sınıflandırma katmanı ile oluşturulmuş ve eğitim gerçekleştirilmiştir. Modellerin yapısı Çizelge 5.16'da verilmiştir.

Çizelge 5.16. CNN GRU modellerinin yapıları.

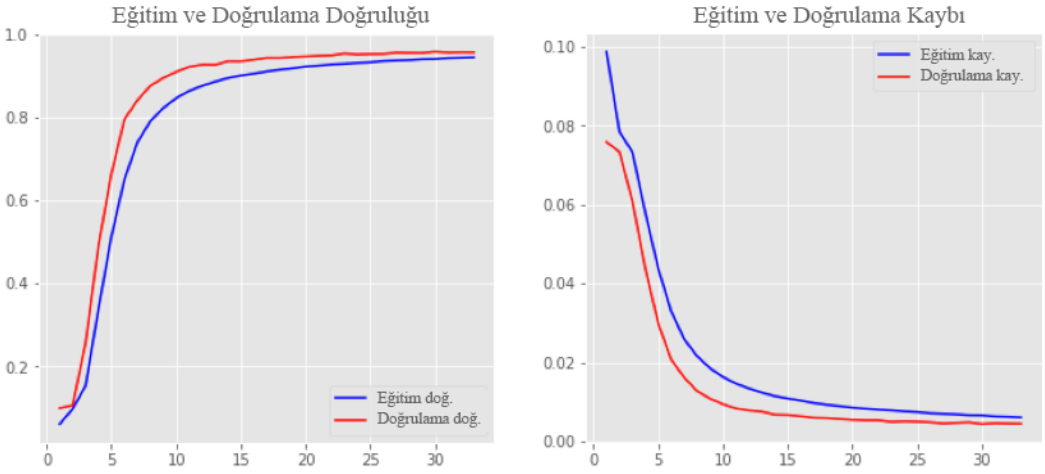
Model	Katman	Model Kodu	CONV1	CONV2	CONV3	GRU	Dense	Batch Boyutu
CNN GRU	Embedding, (Conv1D, MaxPool1D, Dropout) x 3, GRU, Dropout, Dense, Dense (sınıflandırma)	M-7-1	32	64	128	64	128	256
		M-7-2	32	64	128	128	128	256
		M-7-3	64	128	256	128	128	256
		M-7-4	64	128	256	256	128	256
		M-7-5	64	128	256	256	256	256

Çizelge 5.16'daki parametrelere göre eğitilen modellerin sonuçları Çizelge 5.17'de verilmiştir.

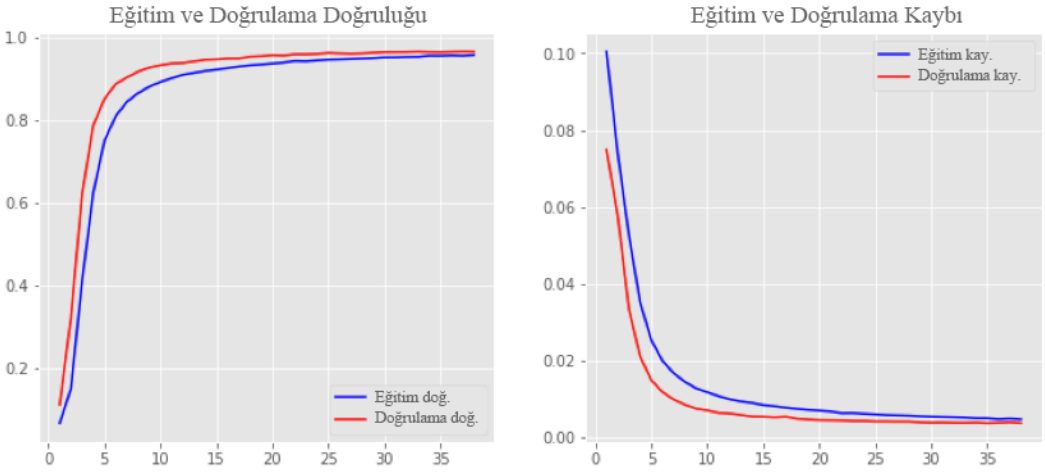
Çizelge 5.17. CNN GRU modellerinin sonuçları.

Model Kodu	Epoch	Eğitim (%)	Doğrulama (%)	Test (%)
M-7-1	33	97.94	95.74	96.24
M-7-2	38	98.60	96.47	96.83
M-7-3	**38**	**99.32**	**97.70**	**97.97**
M-7-4	24	99.13	97.73	97.90
M-7-5	18	98.83	97.23	97.50

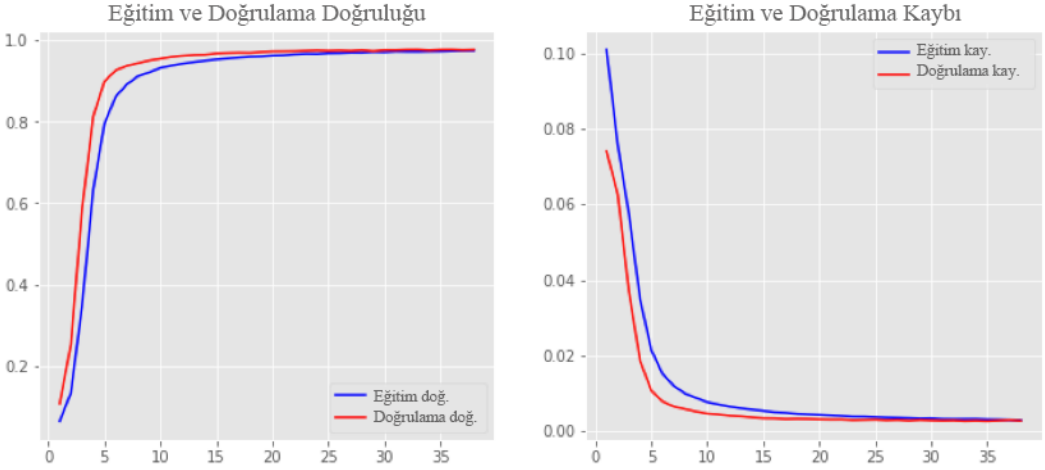
Modellerin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.10'da verilmiştir.



(a) M-7-1 modelinin eğitim ve doğrulama sonuç grafikleri.

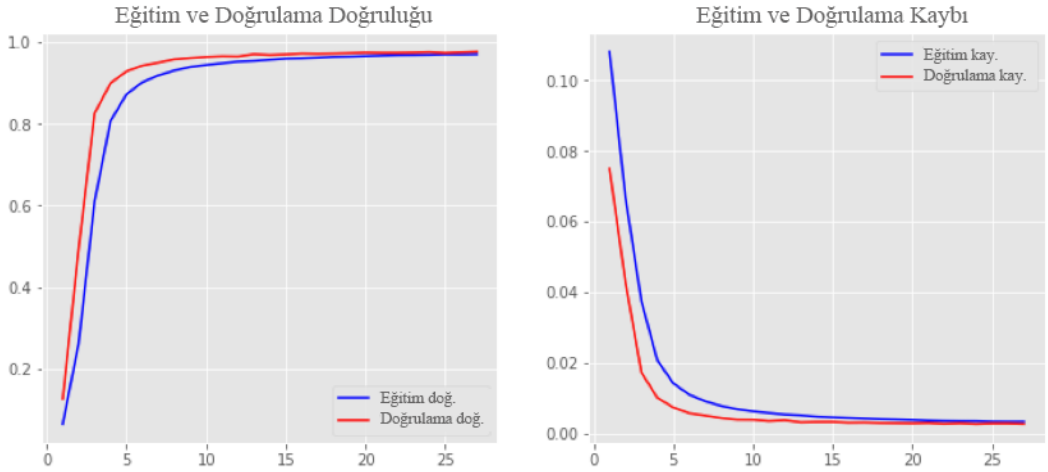


(b) M-7-2 modelinin eğitim ve doğrulama sonuç grafikleri.

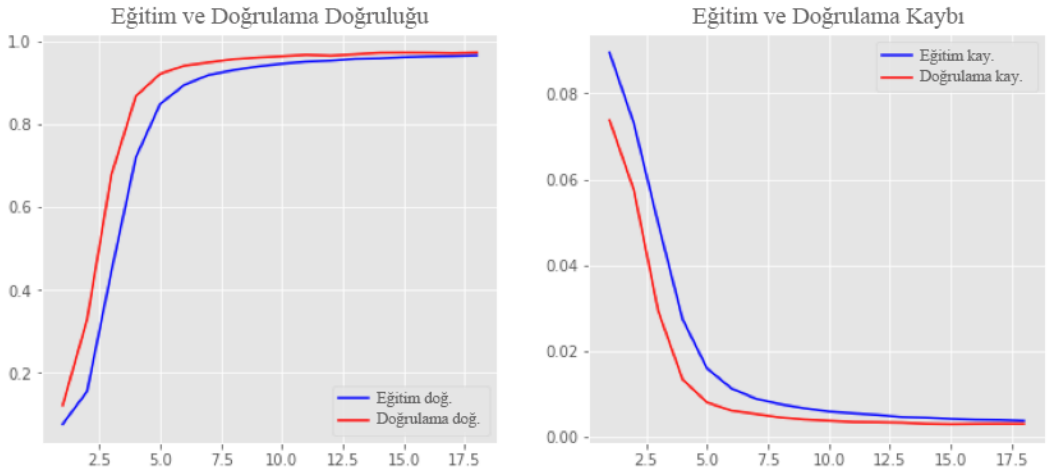


(c) M-7-3 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.10. CNN GRU modellerinin eğitim ve doğrulama sonuç grafikleri. Model kodları a) M-7-1, b) M-7-2, c) M-7-3, d) M-7-4, e) M-7-5.



(d) M-7-4 modelinin eğitim ve doğrulama sonuç grafikleri.



(e) M-7-5 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.10. (Devam ediyor).

Gerçekleştirilen eğitim, doğrulama ve test sürecinden sonra M-7-3 kodlu model CNN GRU modelleri içerisinde en başarılı model olarak belirlenmiştir.

Hibrit modellerin tek yönlü LSTM ve GRU katmanları ile dahi yüksek başarısı görüldüğünden ötürü bu iki hibrit modelin çift yönlü versiyonları da farklı parametrelerle beraber test edilmiştir.

CNN Çift Yönlü LSTM Modelleri

CNN LSTM mimarisinin yüksek başarısı üzerine bu modellerden en başarılı olanı çift yönlü LSTM katmanı ile beraber tekrar oluşturulmuş ve test edilmiştir. Bu model bir

adet gömme katmanı, üç adet evrişim, maksimum havuzlama ve seyreltme katmanı, bir adet çift yönlü LSTM katmanı, bir adet seyreltme katmanı, bir adet tam bağlı katman ve bir adet de tam bağlı sınıflandırma katmanı ile oluşturulmuş ve eğitim gerçekleştirilmiştir. Modelin yapısı Çizelge 5.18'de verilmiştir.

Çizelge 5.18. CNN Çift Yönlü LSTM modelinin yapısı.

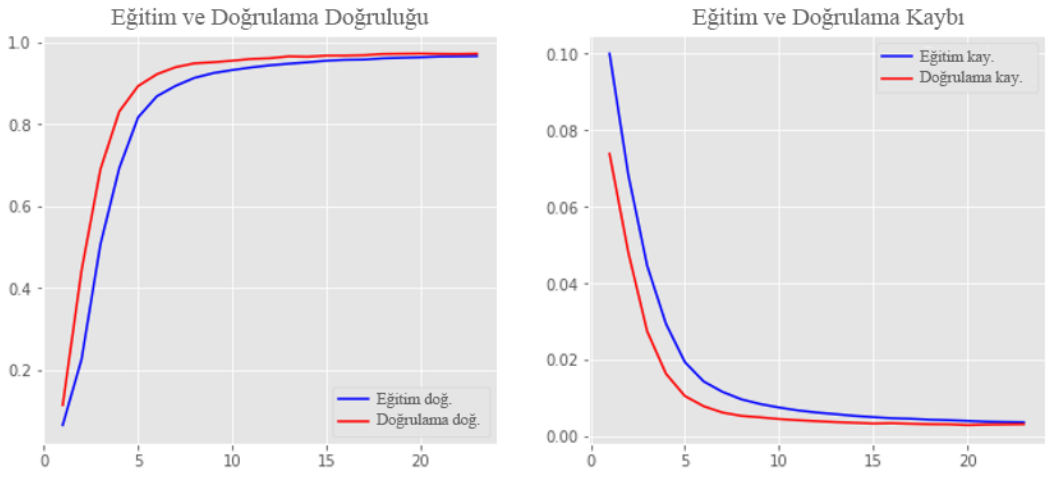
Model	**Katman**	**Model Kodu**	**CONV1**	**CONV2**	**CONV3**	**LSTM**	**Dense**	**Batch Boyutu**
CNN Çift Yönlü LSTM	Embedding, (Conv1D, MaxPool1D, Dropout) x 3, Bidirectional LSTM, Dropout, Dense, Dense (sınıflandırma)	M-8-1	64	128	256	128	128	256

Çizelge 5.18'deki parametrelere göre eğitilen modelin sonuçları Çizelge 5.19'da verilmiştir.

Çizelge 5.19. CNN Çift Yönlü LSTM modelinin sonuçları.

Model Kodu	**Epoch**	**Eğitim (%)**	**Doğrulama (%)**	**Test (%)**
M-8-1	23	98.89	97.23	97.45

Modelin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.11'de verilmiştir.



Şekil 5.11. CNN Çift Yönlü LSTM modelinin eğitim ve doğrulama sonuç grafikleri.

CNN Çift Yönlü GRU Modelleri

CNN GRU mimarisinin yüksek başarısı üzerine bu modellerden en başarılı olanı da dahil olmak üzere başarılı bir kısmı çift yönlü GRU katmanı ile beraber tekrar oluşturulmuş ve test edilmiştir. Bu modeller bir adet gömme katmanı, üç adet evrişim, maksimum havuzlama ve seyreltme katmanı, bir adet çift yönlü GRU katmanı, bir adet seyreltme katmanı, bir adet tam bağlı katman ve bir adet de tam bağlı sınıflandırma katmanı ile oluşturulmuş ve eğitim gerçekleştirilmiştir. Modellerin yapısı Çizelge 5.20'de verilmiştir.

Çizelge 5.20. CNN Çift Yönlü GRU modellerinin yapıları.

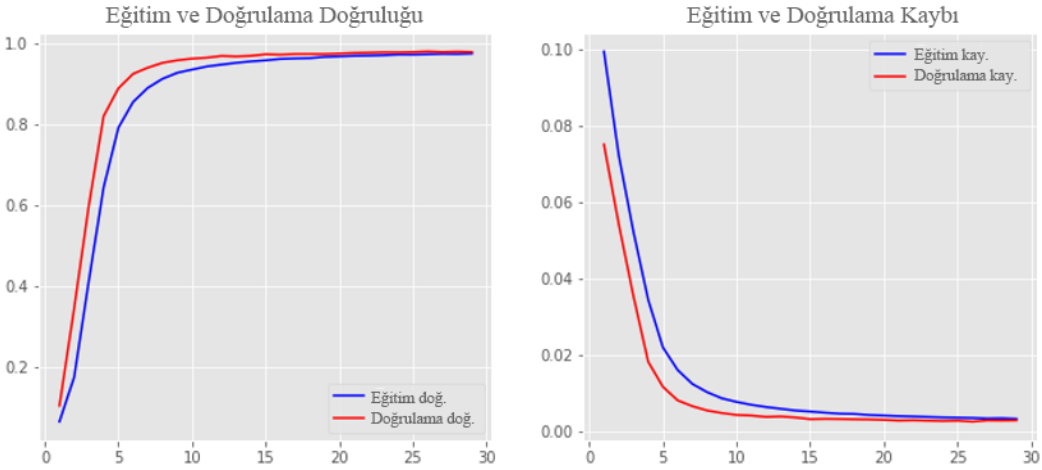
Model	Katman	Model Kodu	CONV1	CONV2	CONV3	GRU	Dense	Batch Boyutu
CNN Çift Yönlü GRU	Embedding, (Conv1D, MaxPool1D, Dropout) x 3, Bidirectional GRU, Dropout, Dense, Dense (sınıflandırma)	M-9-1	64	128	256	128	128	256
		M-9-2	64	128	256	128	256	256
		M-9-3	128	256	256	256	128	256
		M-9-4	128	256	256	384	128	256
		M-9-5	128	256	256	384	256	256

Çizelge 5.20'deki parametrelere göre eğitilen modellerin sonuçları Çizelge 5.21'de verilmiştir.

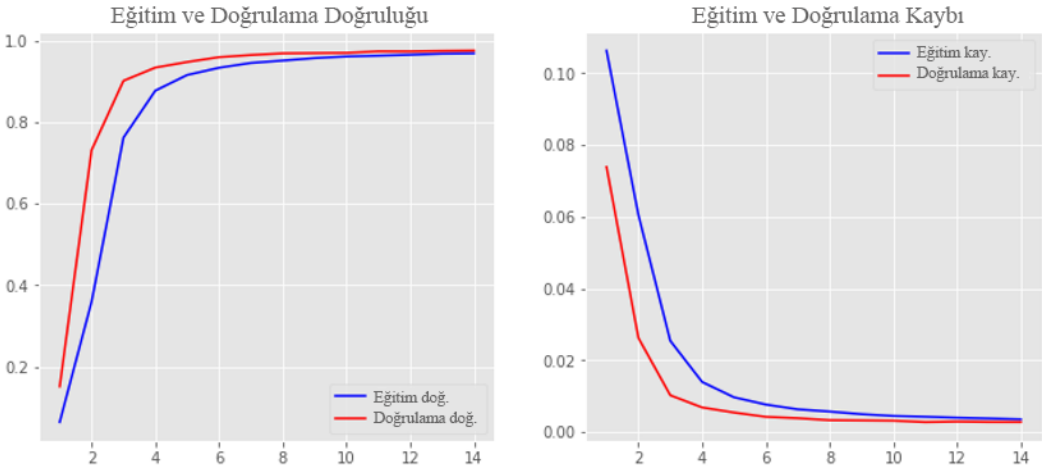
Çizelge 5.21. CNN Çift Yönlü GRU modellerinin sonuçları.

Model Kodu	Epoch	Eğitim (%)	Doğrulama (%)	Test (%)
M-9-1	29	99.16	97.70	98.00
M-9-2	14	98.94	97.57	97.76
M-9-3	27	99.29	97.72	98.08
M-9-4	**23**	**99.33**	**98.00**	**98.23**
M-9-5	17	99.16	98.06	98.05

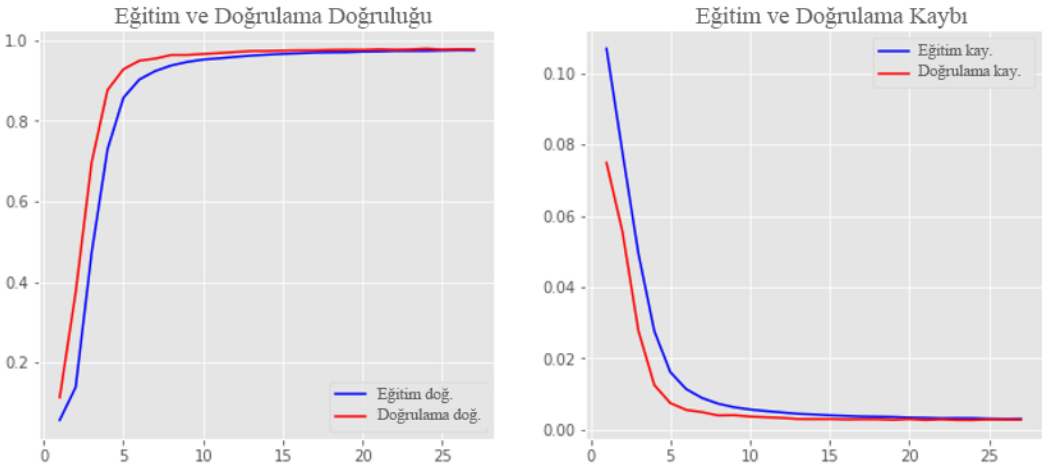
Modellerin her bir epocha göre eğitim ve test için doğruluk ve kayıp grafikleri Şekil 5.12'de verilmiştir.



(a) M-9-1 modelinin eğitim ve doğrulama sonuç grafikleri.

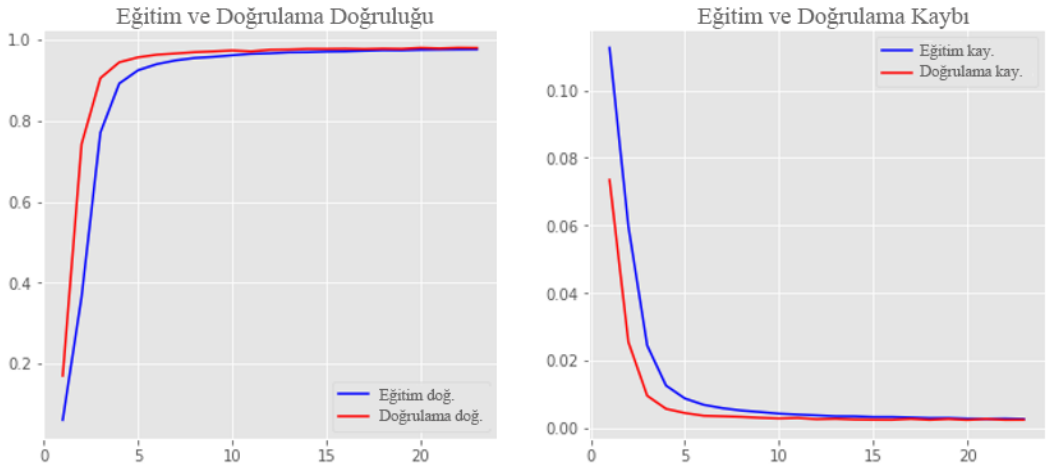


(b) M-9-2 modelinin eğitim ve doğrulama sonuç grafikleri.

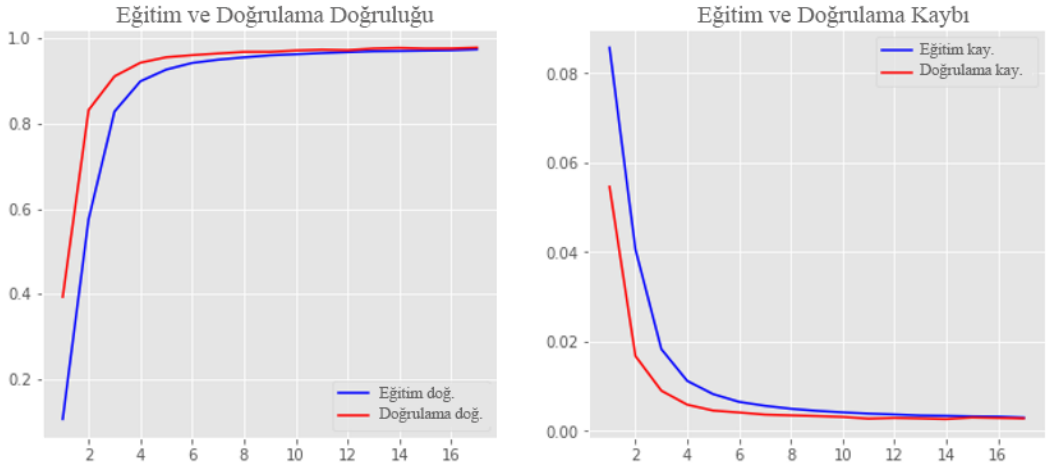


(c) M-9-3 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.12. CNN Çift Yönlü GRU modelinin eğitim ve doğrulama sonuç grafikleri. Model kodları a) M-9-1, b) M-9-2, c) M-9-3, d) M-9-4, e) M-9-5.



(d) M-9-4 modelinin eğitim ve doğrulama sonuç grafikleri.



(e) M-9-5 modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 5.12. (Devam ediyor).

Gerçekleştirilen eğitim, doğrulama ve test sürecinden sonra M-9-4 kodlu model CNN Çift Yönlü GRU modelleri içerisinde en başarılı model olarak belirlenmiştir. Ayrıca modellerdeki evrişimli katman sayılarına bakılırsa katman sayısı belli bir düzeyin üstüne çıktığında ezberlemeye yatkınlık artmış, eğitim başarısı arttığı halde doğrulama ve bilhassa test başarısı aşağıya düşmüştür. Benzer durum diğer katmanlardaki sayılar için de kısmi olarak geçerlidir.

Genel duruma bakılacak olursa tüm modeller içerisinde katman sayıları arttıkça daha az epoch sayısında öğrenme gerçekleşmekte, bilhassa tam bağlı katmandaki katman sayısındaki fazla artış modelin doğrulama ve test başarısını aşağı yöne çekmiş ve modeli ezberleme noktasına yaklaştırmıştır. Bu sebepten de erken durdurma aracı ile

beraber modellerin en uygun epoch sayıları belirlenmiş, ayrıca yapılan testlerle beraber tüm modellerin en başarılı versiyonları kaydedilmiştir. Burada kaydedilen en başarılı modeller arasından CNN Çift Yönlü GRU modeli tüm modeller arasında en yüksek doğruluk değerine sahiptir. Ayrıca bu model literatürde yer alan ve benzer veri setleri ile çalışan modeller arasında da en yüksek doğruluk değerine sahiptir. Hem %98.23 doğruluk oranı ile en yüksek başarıya sahip olması hem de tasarlanan üçlü hibrit yapısı ile literatüre katılan önemli bir yenilik olarak işlenmektedir. Ayrıca önerilen diğer bir model olan Çift Katmanlı Çift Yönlü LSTM modeli de %96.60 f-skor değeri ile hazırlanan tüm modeller arasında en yksek f-skor değerine ve başarıya sahiptir.

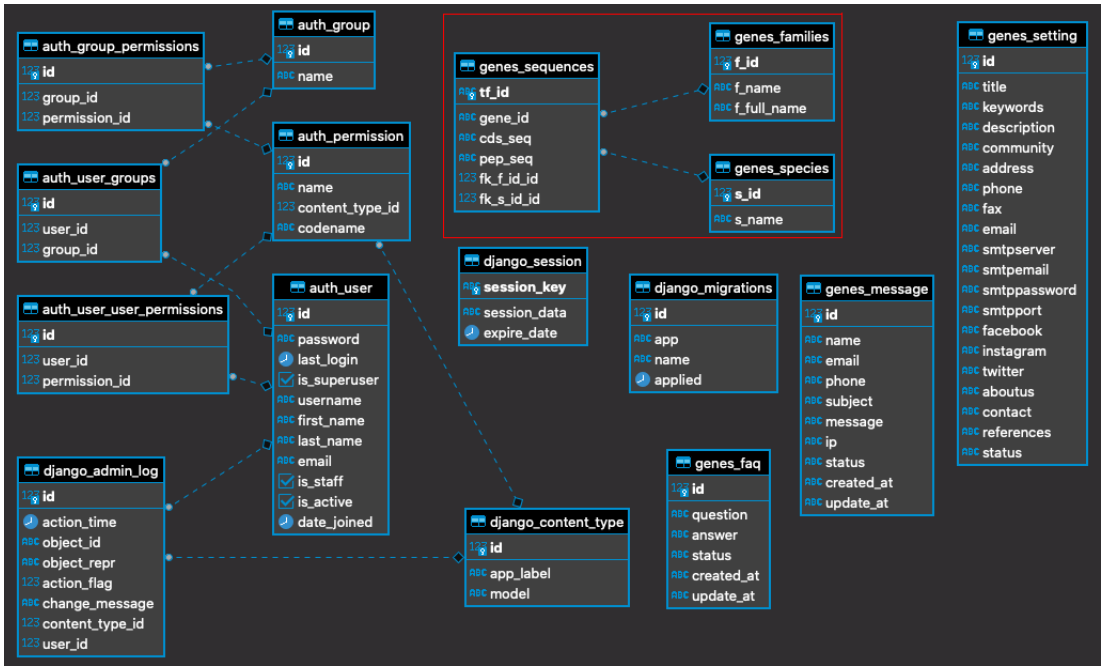
5.3. VERİTABANININ İNTERNET SİTESİ

Halihazırda umumi erişimine açık olan veri tabanları incelendiğinde hem bHLH TF proteinlerine özel hem de derin öğrenme tabanlı bir sınıflandırma yapan herhangi bir veri tabanının olmadığı görülmektedir. Bu nedenle bu çalışma kapsamında hazırlanmış olan ve içerisinde proteinlerin derin öğrenme tabanlı sınıflandırmasını da bulunduran bu veritabanı "yeni nesil" olarak tanımlanmakta olup, sınıfının ilk örneklerindendir.

5.3.1. Veritabanının Yapısı

Literatürde eksikliği görülmüş olan bHLH TF proteinleri veritabanını oluşturmak için öncelikle bitki genomik kaynakları üzerindeki bHLH proteinleri araştırılmıştır. Bu kaynağa erişim, Phytozome [59]'un internet sitesi ve bağlı sunucusu ile gerçekleştirilmiştir. Elde edilen genomik bilgiden bHLH proteinlerini ayırmak için, bHLH proteinlerinin korunan alanları, CLC Genomics Workbench v11 yazılımı tarafından gerçekleştirilen PFAM Etki Alanı Araması ile elde edilmiştir. Ardından protein dizileri ve DNA dizileri ayrıntılı ve büyük miktarlarda bHLH dizileri içeren PlantTFDB [60] veritabanından alınmıştır. Bu ham veriler Bölüm 4.5'te detayları ile belirtilen Python programlama dili ile hazırlanmış betiklerle alınmıştır. bHLH TF protein ailesi için hazırlanan yeni nesil arama ve analiz modelleri barındıran özel biyoinformatik veritabanı için CDS ve PEP dizilerini barındıran bu dosyalar, içerisindeki tanımlayıcı numaralar ve bitki tür isimleri ile beraber bir Python betiği

vasıtasıyla Bölüm 4.5'te açıklanan detayları ile tasnif edilmiş ve NCBI'dan elde edilen protein dizileri bilgilerinden NCBI tanımlayıcı numaraları yine bir Python betiği ile otomatik olarak çıkarılarak ve tasnif edilen dosyaya eklenmiştir. Hazırlanan bu dosya ile beraber tüm diziler ve bu dizilein ek tanmlayıcı bilgileri ve bHLH TF proteinleri ile ilgili genel ve teknik bilgiler ilişkisel olarak tasarlanmış olan veritabanına yüklenerek yayınlanmaya hazır hale getirilmiştir [117]. Veritabanının varlık ilişki (entity relationship-ER) diyagramı Şekil 5.13'te verilmiştir.



Şekil 5.13. Veritabanının ER diyagramı [117].

Veritabanının ER diyagramında kırmızı kutu içerisine alınan kısım veritabanının bHLH dizileri ile ilgili olan kısmıdır. Diğer tablolar ise internet sitesinin ve internet sitesindeki diğer verilerin tutulduğu, internet sitesinin Django içe oluşturulan çatısı ile beraber otomatik oluşturulan tablolardır. Kırmızı kutu içerisindeki tabloların detayı şu şekildedir:

- **genes_families:** Veritabanında yer alan gen ailelerinin bulunduğu tablodur. İçerisinde, veritabanı içerisinde yer alan aileler yer alır. İçerisinde ailenin tam adı ve kısa adı bulunur.

- **genes_sequences:** Veritabanında yer alan dizilerin bulunduğu tablodur. İçerisinde gen ID'si, DNA (CDS) dizisi ve protein (PEP) dizisi bulunur.

- **genes_species:** Veritabanında yer alan türlerin bulunduğu tablodur. İçerisinde her bir dizinin türünü barındırır.

Veritabanında yer alan diğer tablolardan ismi "auth" ile başlayanlar veritabanı ve internet sitesi için gereken yetkilendirme bilgilerini barındırırken, ismi "django" ile başlayanlar ise internet sitesinin diğer hizmetleri için gereken bilgileri barındırır.

Hazırlanan bu yeni nesil veritabanı 166 farklı bitki türüne ait 28698 adet bHLH TF protein dizisi ve DNA dizisi içerir. Her bit bitki türünü ve bHLH dizisi sayısını gösterir tablo Çizelge 5.22'de verilmiştir.

Çizelge 5.22. Bitki türlerine göre bHLH dizilerinin sayısı [117].

No	Bitki Türü	bHLH Sayısı	No	Bitki Türü	bHLH Sayısı
1	*Actinidia chinensis*	181	84	*Mimulus guttatus*	161
2	*Aegilops tauschii*	131	85	*Monoraphidium neglectum*	3
3	*Aethionema arabicum*	125	86	*Morus notabilis*	116
4	*Amaranthus hypochondriacus*	120	87	*Musa acuminata*	298
5	*Amborella trichopoda*	84	88	*Nelumbo nucifera*	127
6	*Ananas comosus*	121	89	*Nicotiana benthamiana*	277
7	*Aquilegia coerulea*	178	90	*Nicotiana sylvestris*	240
8	*Arabidopsis halleri*	142	91	*Nicotiana tabacum*	435
9	*Arabidopsis lyrata*	155	92	*Nicotiana tomentosiformis*	245
10	*Arabidopsis thaliana*	225	93	*Ocimum tenuiflorum*	168
11	*Arabis alpina*	111	94	*Oropetium thomaeum*	130
12	*Arachis duranensis*	156	95	*Oryza barthii*	170
13	*Arachis hypogaea*	72	96	*Oryza brachyantha*	132
14	*Arachis ipaensis*	160	97	*Oryza glaberrima*	147
15	*Artemisia annua*	45	98	*Oryza glumaepatula*	197
16	*Auxenochlorella protothecoides*	3	99	*Oryza longistaminata*	117
17	*Azadirachta indica*	207	100	*Oryza meridionalis*	175
18	*Bathycoccus prasinos*	1	101	*Oryza nivara*	199
19	*Beta vulgaris*	121	102	*Oryza punctata*	192
20	*Boechera stricta*	170	103	*Oryza rufipogon*	176
21	*Brachypodium distachyon*	247	104	*Oryza sativa subsp. indica*	169

Çizelge 5.22. (Devam ediyor).

22	*Brachypodium stacei*	177	105	*Oryza sativa subsp. japonica*	211
23	*Brassica napus*	553	106	*Ostreococcus lucimarinus*	1
24	*Brassica oleracea*	393	107	*Ostreococcus sp. RCC809*	4
25	*Brassica rapa*	371	108	*Ostreococcus tauri*	1
26	*Cajanus cajan*	174	109	*Panicum hallii*	258
27	*Camelina sativa*	467	110	*Panicum virgatum*	559
28	*Cannabis sativa*	99	111	*Petunia axillaris*	175
29	*Capsella grandiflora*	157	112	*Petunia inflata*	182
30	*Capsella rubella*	163	113	*Phalaenopsis equestris*	96
31	*Capsicum annuum*	129	114	*Phaseolus vulgaris*	203
32	*Carica papaya*	105	115	*Phoenix dactylifera*	128
33	*Castanea mollissima*	101	116	*Phyllostachys heterocycla*	168
34	*Catharanthus roseus*	218	117	*Physcomitrella patens*	462
35	*Chlamydomonas reinhardtii*	12	118	*Picea abies*	107
36	*Chlorella variabilis NC64A*	2	119	*Picea glauca*	42
37	*Cicer arietinum*	197	120	*Picea sitchensis*	34
38	*Citrullus lanatus*	126	121	*Picochlorum sp. SENEW3*	1
39	*Citrus clementina*	163	122	*Pinus taeda*	31
40	*Citrus sinensis*	194	123	*Populus euphratica*	178
41	*Coccomyxa subellipsoidea C-169*	5	124	*Populus trichocarpa*	379
42	*Coffea canephora*	122	125	*Prunus mume*	165
43	*Cucumis melo*	178	126	*Prunus persica*	282
44	*Cucumis sativus*	192	127	*Pseudotsuga menziesii*	230
45	*Daucus carota*	171	128	*Pyrus bretschneideri*	197
46	*Dianthus caryophyllus*	128	129	*Raphanus raphanistrum*	230
47	*Dichanthelium oligosanthes*	157	130	*Raphanus sativus*	254
48	*Dorcoceras hygrometricum*	109	131	*Ricinus communis*	121
49	*Dunaliella salina*	6	132	*Saccharum officinarum*	48
50	*Elaeis guineensis*	251	133	*Salix purpurea*	393
51	*Eragrostis tef*	159	134	*Salvia miltiorrhiza*	151
52	*Eucalyptus camaldulensis*	133	135	*Selaginella moellendorffii*	55
53	*Eucalyptus grandis*	178	136	*Sesamum indicum*	219
54	*Eutrema salsugineum*	173	137	*Setaria italica*	233
55	*Fragaria vesca*	112	138	*Setaria viridis*	271
56	*Fragaria x ananassa*	94	139	*Sisymbrium irio*	156
57	*Genlisea aurea*	80	140	*Solanum lycopersicum*	161
58	*Glycine max*	548	141	*Solanum melongena*	121
59	*Glycine soja*	342	142	*Solanum pennellii*	172
60	*Gonium pectorale*	5	143	*Solanum pimpinellifolium*	158
61	*Gossypium arboreum*	215	144	*Solanum tuberosum*	206

Çizelge 5.22. (Devam ediyor).

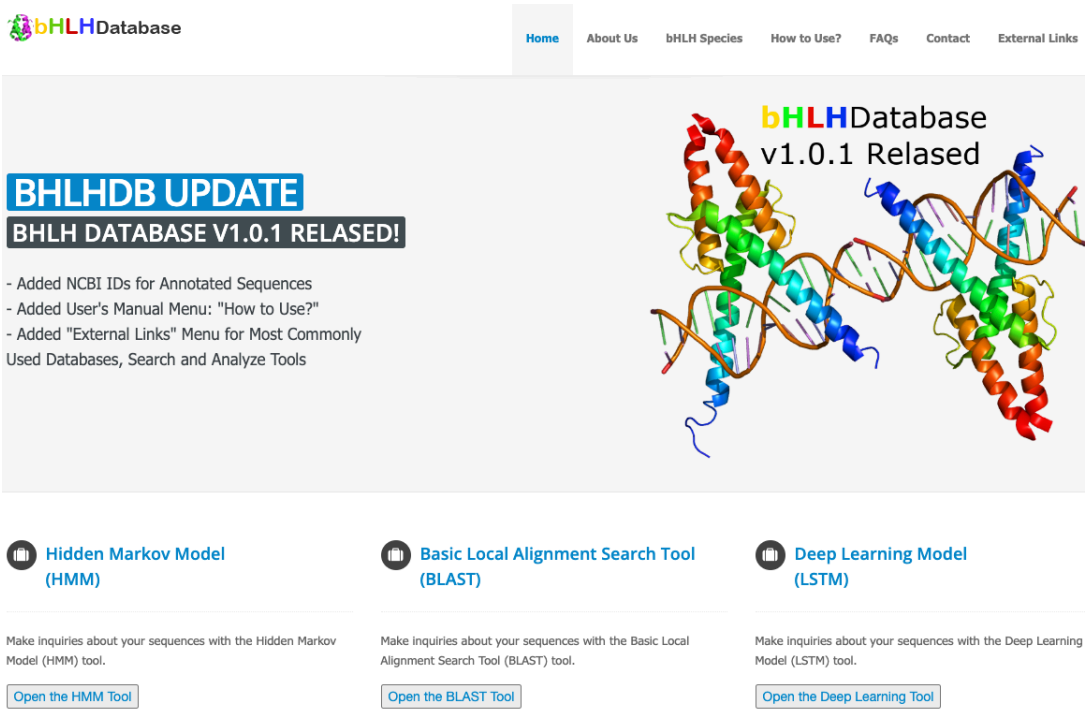
62	*Gossypium hirsutum*	427	145	*Sorghum bicolor*	297
63	*Gossypium raimondii*	426	146	*Sphagnum fallax*	165
64	*Helianthus annuus*	24	147	*Spinacia oleracea*	111
65	*Helicosporidium*	1	148	*Spirodela polyrhiza*	94
66	*Hordeum vulgare*	266	149	*Tarenaya hassleriana*	335
67	*Humulus lupulus*	103	150	*Thellungiella parvula*	154
68	*Ipomoea trifida*	171	151	*Theobroma cacao*	200
69	*Jatropha curcas*	113	152	*Trifolium pratense*	162
70	*Juglans regia*	125	153	*Triticum aestivum*	324
71	*Kalanchoe laxiflora*	247	154	*Triticum urartu*	111
72	*Kalanchoe marnieriana*	393	155	*Utricularia gibba*	132
73	*Klebsormidium flaccidum*	10	156	*Vigna angularis*	192
74	*Lactuca sativa*	90	157	*Vigna radiata*	153
75	*Leersia perrieri*	194	158	*Vigna unguiculata*	61
76	*Linum usitatissimum*	195	159	*Vitis vinifera*	115
77	*Lotus japonicus*	177	160	*Volvox carteri*	3
78	*Malus domestica*	250	161	*Zea mays*	308
79	*Manihot esculenta*	224	162	*Ziziphus jujuba*	185
80	*Marchantia polymorpha*	105	163	*Zostera marina*	103
81	*Medicago truncatula*	259	164	*Zoysia japonica*	216
82	*Micromonas pusilla CCMP1545*	2	165	*Zoysia matrella*	386
83	*Micromonas sp. RCC299*	2	166	*Zoysia pacifica*	256

5.3.2. İnternet Sitesi

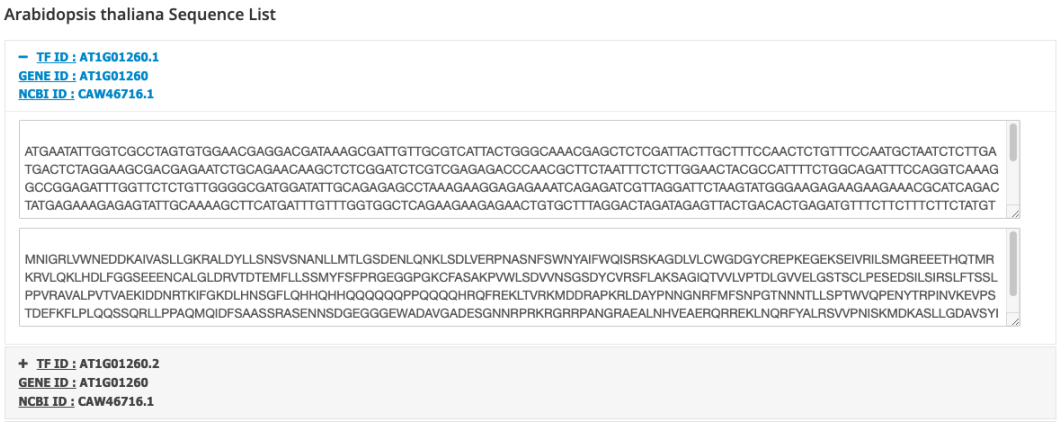
Hazırlanan veritabanı, Python Programlama ve Django web çatısı ile hazırlanmıştır ve aktif olarak http://www.bhlhdb.org/ alan adına sahip internet sitesinde kullanıma açılmıştır. Bu internet sitesi içerisinde bHLH TF proteinlerinin tanımlayıcı numaraları ve dizileri, bHLH TF proteinleri ile ilgili çeşitli bilgiler, çeşitli analiz ve sınıflandırma araçları yer almaktadır.

Siteye literatürde aktif kullanımı olan HMM ve BLAST arama araçları eklenmiştir. Bu araçların her biri kendi resmi dokümanına göre internet sitesine yerleştirilmiştir. HMM analiz aracı, HMMER [123]'in kendi internet sitesinde ve resmi dokümanında yer alan şekli ile bHLH TF proteinlerini analiz etmek üzere hazırlanmıştır. BLAST [124,125] aracı ise yine kendi internet sitesinde ve resmi dokümanında yer alan şekli ile protein

hizalamaları ve analizi yapmak üzere hazırlanmıştır. Şekil 5.14'te (a) internet sitesinin genel görünümü ve arama araçları ve (b) diziler ekranı verilmiştir.



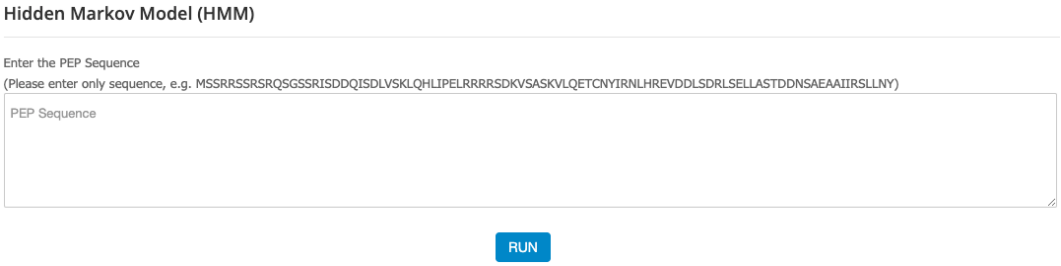
(a) Genel görünüm ve arama araçları.



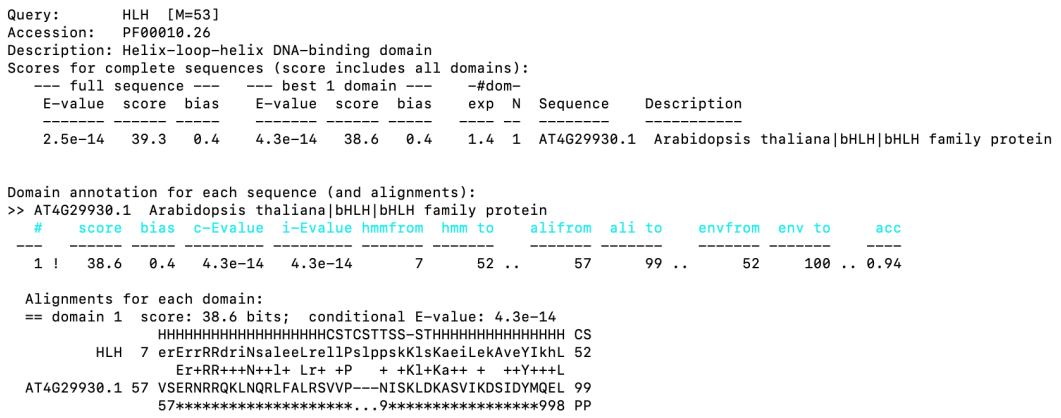
(b) Diziler ekranı.

Şekil 5.14. İnternet sitesinin a) genel görünümü ve arama araçları, b) diziler ekranı.

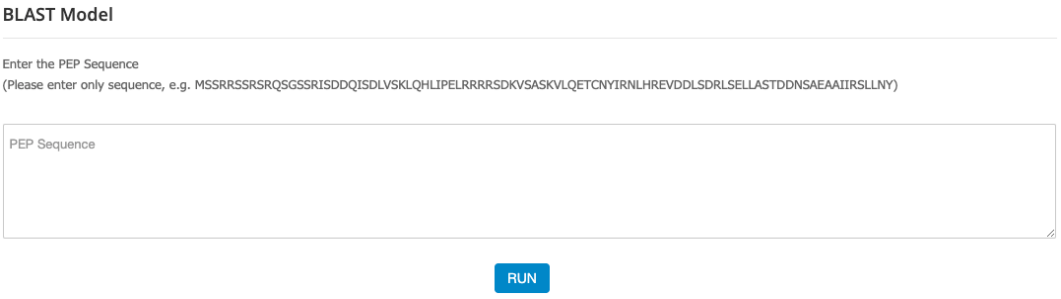
Şekil 5.14'teki ekranlar yardımıyla hem diziler hakkında bilgiler alınabilir (b), hem de analiz araçlarına rahatlıkla erişilebilir (a). Şekil 5.15'te ise analiz araçlarının sorgu ve sonuç ekranları verilmiştir.



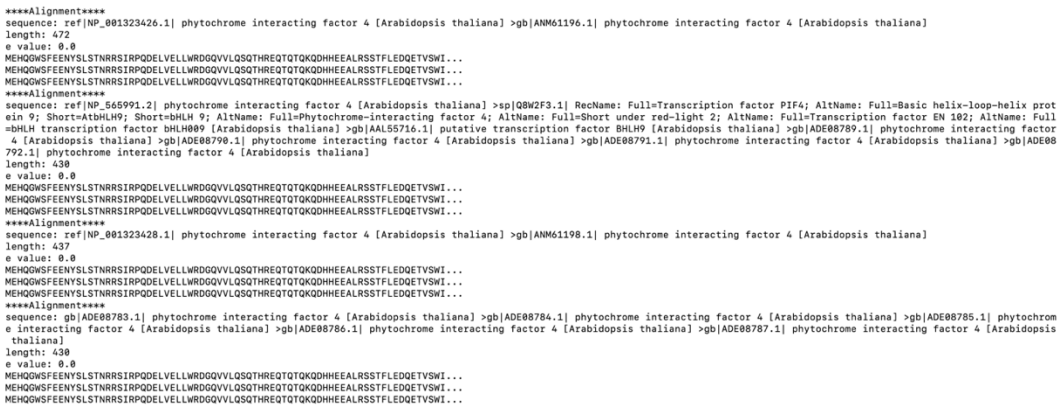
(a) HMM sorgu ekranı.



(b) HMM sonuç ekranı.

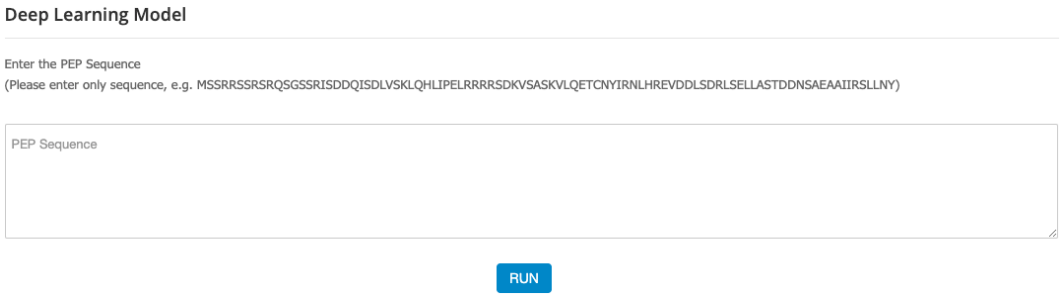


(c) BLAST sorgu ekranı.

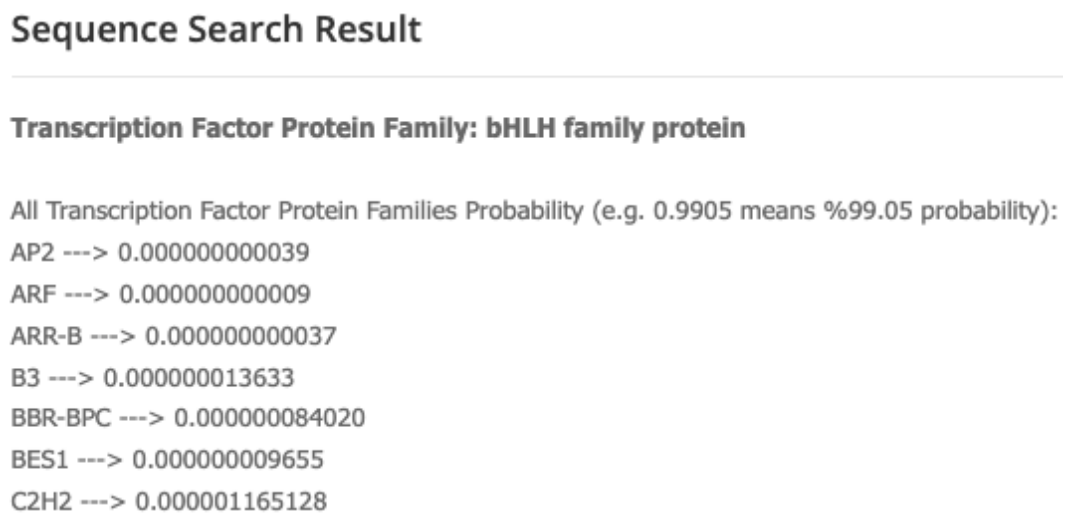


(d) BLAST sonuç ekranı.

Şekil 5.15. İnternet sitesindeki analiz araçlarının sorugu ve sonuç ekranları. a) HMM sorgu, b) HMM sonuç, c) BLAST sorgu, d), BLAST sonuç, e) derin öğrenme sorgu, f) derin öğrenme sonuç ekranı.



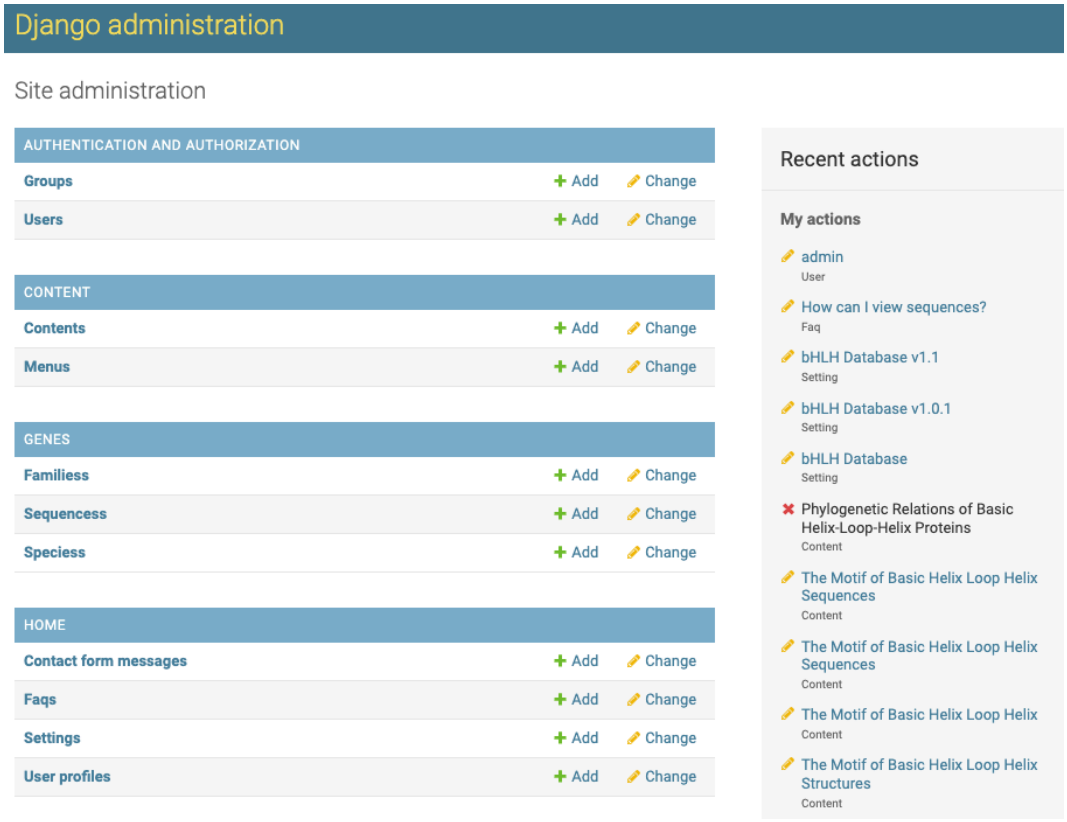
(e) Derin öğrenme sorgu ekranı.



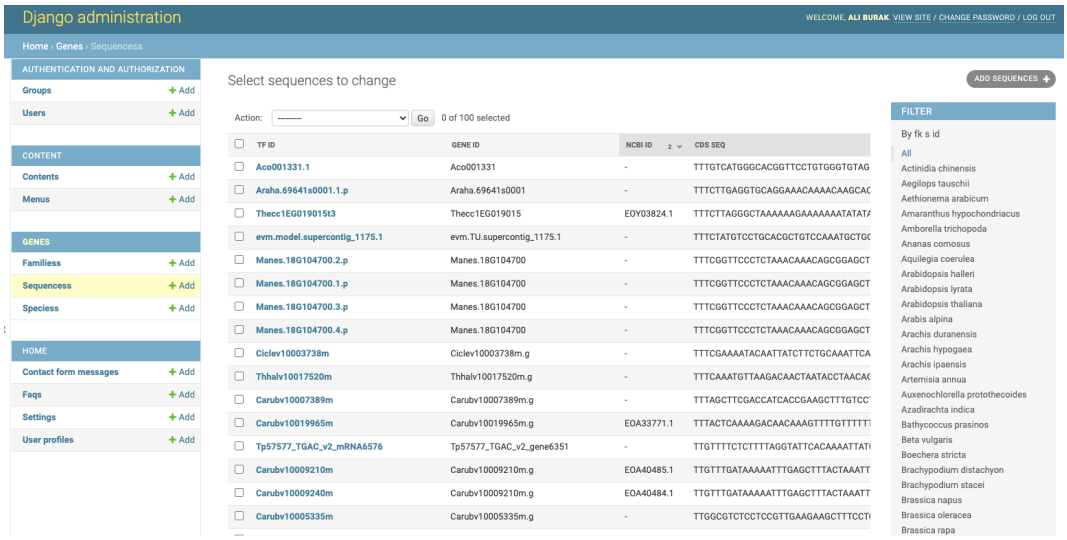
(f) Derin öğrenme sonuç ekranı.

Şekil 5.15. (Devam ediyor).

Şekil 5.16'da ise sitenin tüm yönetimini ve içerik girişini sağlayan yönetici (admin) panelinden ekranlar verilmiştir. Panel sayesinde site ile ilgili tüm düzenlemeler yapılabilmektedir (a) şekli yönetici panelinin genel görüntüsünü ve menülerini, (b) şekli ise dizilerin tüm bilgileri ile beraber ekleme, düzenleme ve silme işlemlerinin yapılabildiği ekranın görüntüsünü içermektedir.



(a) Genel görünüm.



(b) Dizileri düzenleme ekranı.

Şekil 5.16. İnternet sitesinin yönetim (admin) paneli ekranları. a) yönetim panelinin genel görüntüsü, b) yönetim panelinin dizileri düzenleme ekranı.

BÖLÜM 6

BULGULAR VE TARTIŞMA

Tasarlanan modellerde verilerin %80'i eğitim, %10'u doğrulama ve %10'u test için kullanılmıştır. Sadece %20 test yapılması yerine veri seti bu şekilde ayrılarak hem doğrulama hem de test yapılmış ve model iki defa kontrole tabi tutulmuştur. Ayrıca 132330 dizi olduğu için bu bölüm yapılan testler neticesinde örnek miktarı yeterli olduğundan bu bölme miktarları uygun olarak tespit edilmiştir. Eğitim için 105864 dizi, doğrulama için 13233 dizi ve test için 13233 dizi kullanılmıştır. Öğrenme katsayıları, LSTM ve GRU ağlarında 0.01, CNN ağlarında 0.001 ve hibrit modellerde 0.01 varsayılan değer olarak yapılan denemeler sonucunda belirlenmiştir. En uygun epoch sayılarının belirlenmesi için bir erken durdurma aracı eklenmiş olup, her model için farklı epoch sayıları belirlenmiştir. Ayrıca ADAM tüm modellerde optimizasyon fonksiyonu olarak kullanılmıştır. Her model için yapılan deneyler sonucunda farklı katman sayıları ve batch boyutları belirlenmiştir. Bölüm 5.2 içerisinde tanımlanan tüm modellerin en başarıları seçilmiş ve bu modeller arasında doğruluk, kesinlik, hassasiyet, f-skor ve eğitim zamanına bağlı olarak bir kıyaslama yapılmıştır. Ayrıca bu modeller için 10 katlı çapraz doğrulama işlemi uygulanmış ve yapılan kıyaslama pekiştirilmiştir. Çizelge 6.1, tasarlanan modeller içerisinden en başarılı parametrelere sahip seçilmiş modellerin doğruluk, kesinlik, hassasiyet, f-skor ve eğitim zamanına göre karşılaştırmasını göstermektedir.

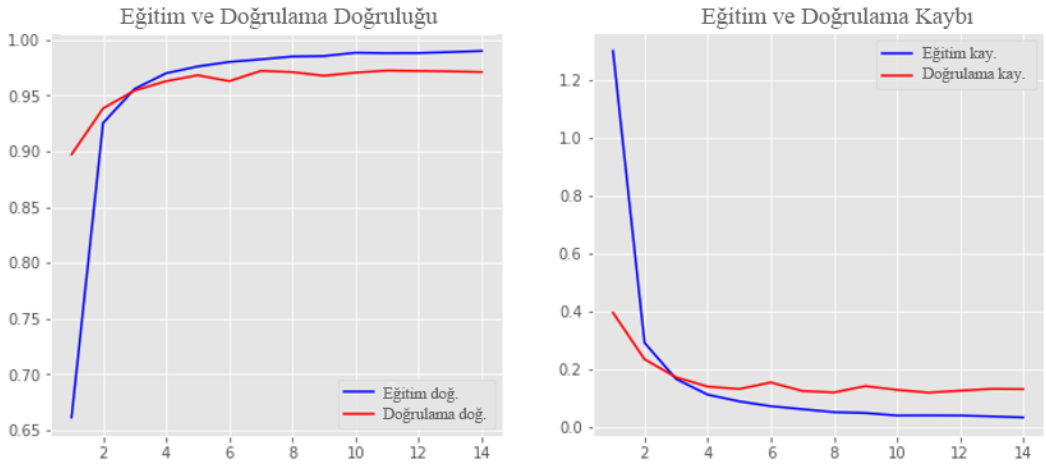
Çizelge 6.1. Tasarlanan modellerin test sonuçları.

Model (Model Kodu)	**Doğruluk (%)**	**Kesinlik (%)**	**Hassasiyet (%)**	**F-Skor (%)**	**Eğitim Süresi (dk)**
Çift Yönlü LSTM (M-1-3)	97.38	96.31	95.24	95.69	20.34
İki Katmanlı Çift Yönlü LSTM (M-2-4)	97.80	**97.10**	96.31	**96.60**	14.48
Çift Yönlü GRU (M-3-6)	97.51	96.75	93.96	94.92	13.88
İki Katmanlı Çift Yönlü GRU (M-4-4)	97.48	95.53	**96.58**	95.78	11.07

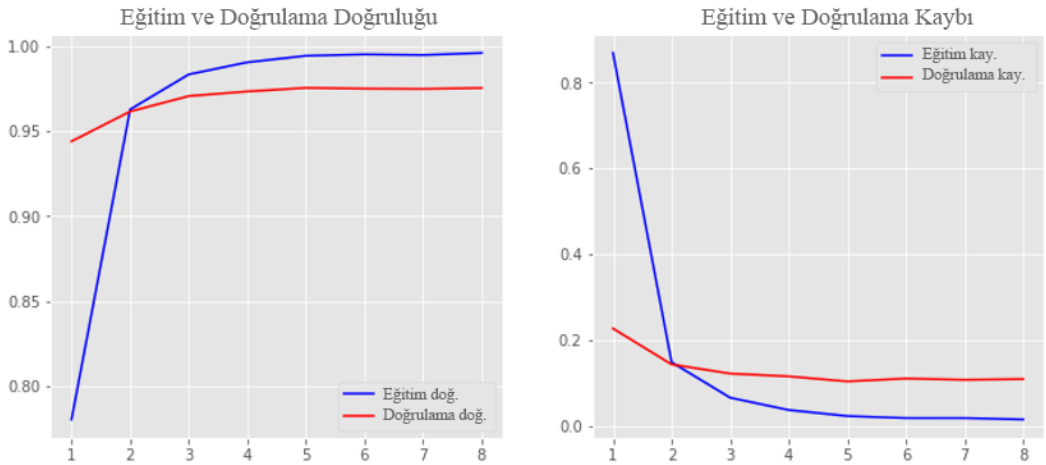
Çizelge 6.1. (Devam ediyor).

CNN (M-5-2)	97.09	93.02	88.71	90.11	29.14
CNN LSTM (M-6-3)	97.25	93.25	93.75	93.26	**10.30**
CNN GRU (M-7-3)	97.70	95.02	95.59	94.96	12.08
CNN Çift Yönlü LSTM (M-8-1)	98.03	95.70	95.98	95.34	11.92
CNN Çift Yönlü GRU (M-9-4)	**98.23**	95.88	95.27	95.36	11.80

Şekil 6.1'de, tasarlanan modeller içerisinden en başarılı parametrelere sahip seçilmiş modellerin eğitim ve doğrulama sonuç grafikleri toplu olarak verilmiştir.



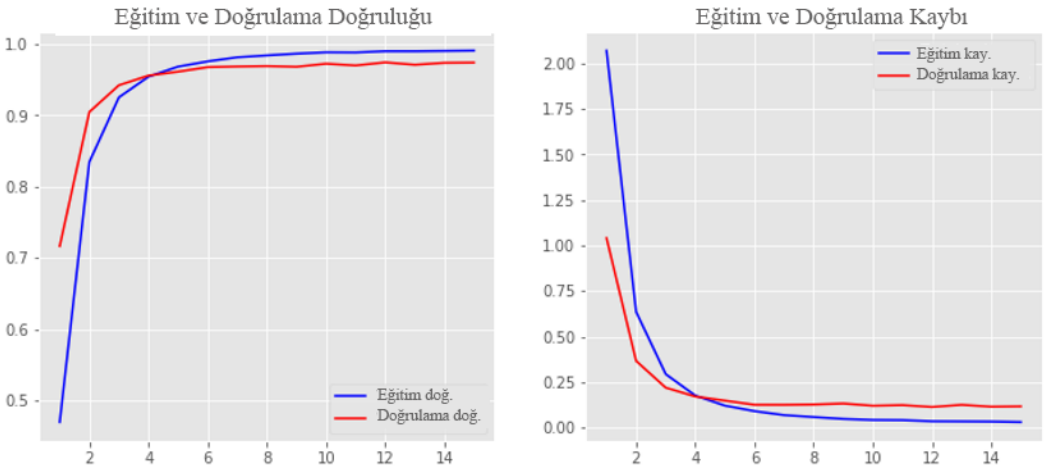
(a) Çift Yönlü LSTM modelinin eğitim ve doğrulama sonuç grafikleri.



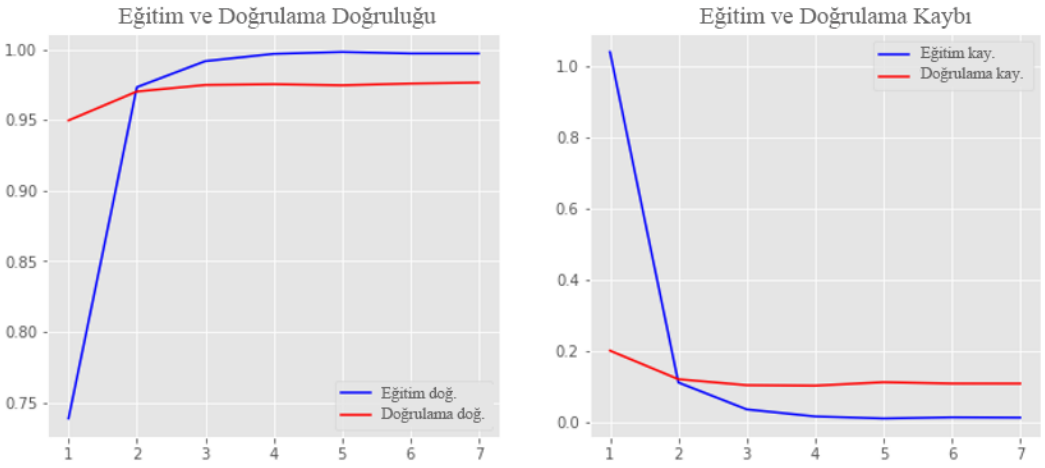
(b) Çift Katmanlı Çift Yönlü LSTM modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 6.1. Seçilmiş modellerin eğitim ve doğrulama sonuç grafikleri. a) Çift Yönlü LSTM, b) Çift Katmanlı Çift Yönlü LSTM, c) Çift Yönlü GRU, d) Çift

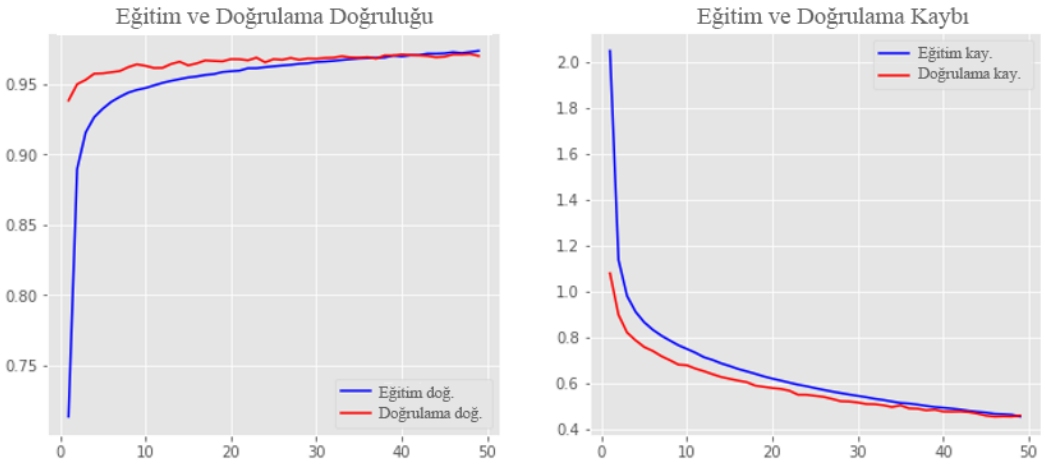
Katmanlı Çift Yönlü GRU, e) CNN, f) CNN LSTM, g) CNN GRU, h) CNN Çift Yönlü LSTM, i) CNN Çift Yönlü GRU.



(c) Çift Yönlü GRU modelinin eğitim ve doğrulama sonuç grafikleri.

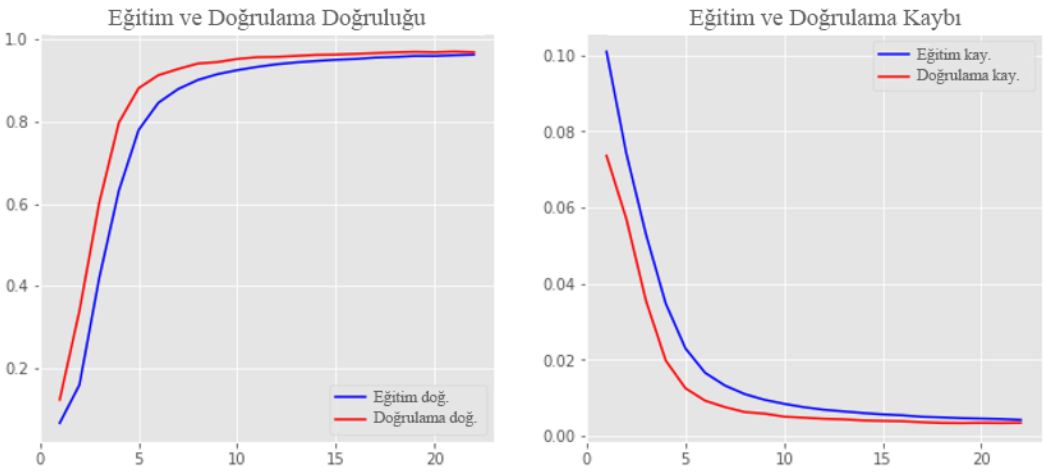


(d) Çift Katmanlı Çift Yönlü GRU modelinin eğitim ve doğrulama sonuç grafikleri.

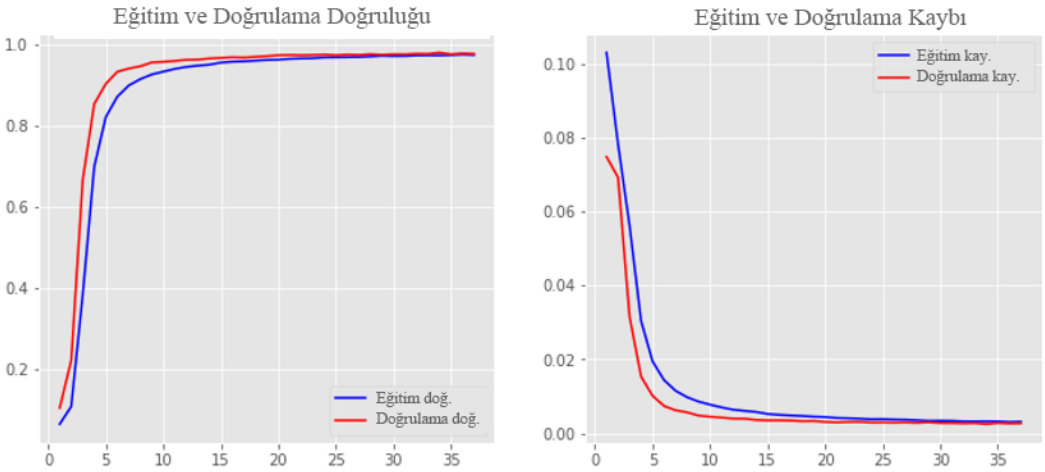


(e) CNN modelinin eğitim ve doğrulama sonuç grafikleri.

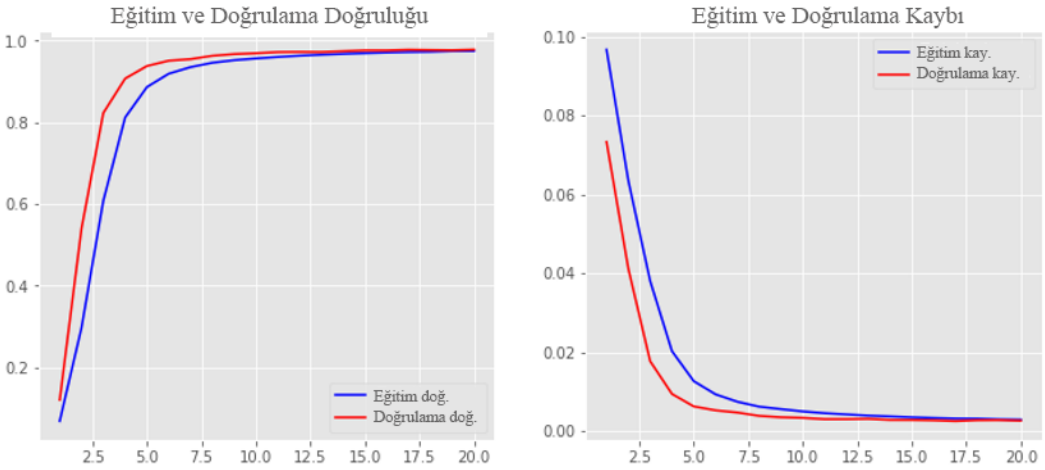
Şekil 6.1. (Devam ediyor).



(f) CNN LSTM modelinin eğitim ve doğrulama sonuç grafikleri.

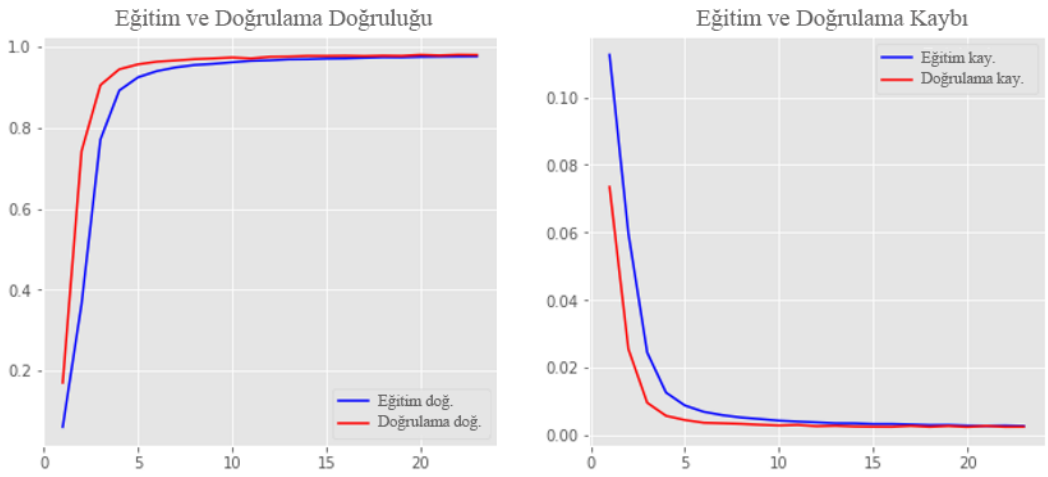


(g) CNN GRU modelinin eğitim ve doğrulama sonuç grafikleri.



(h) CNN Çift Yönlü LSTM modelinin eğitim ve doğrulama sonuç grafikleri.

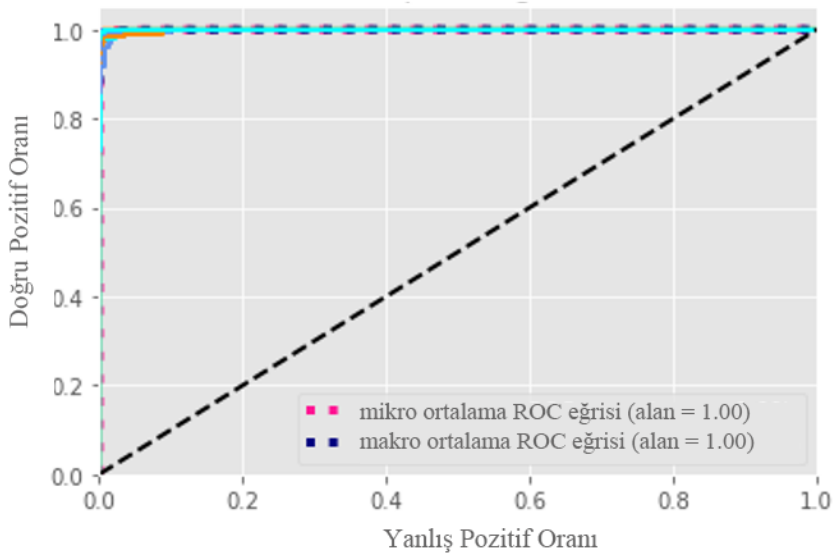
Şekil 6.1. (Devam ediyor).



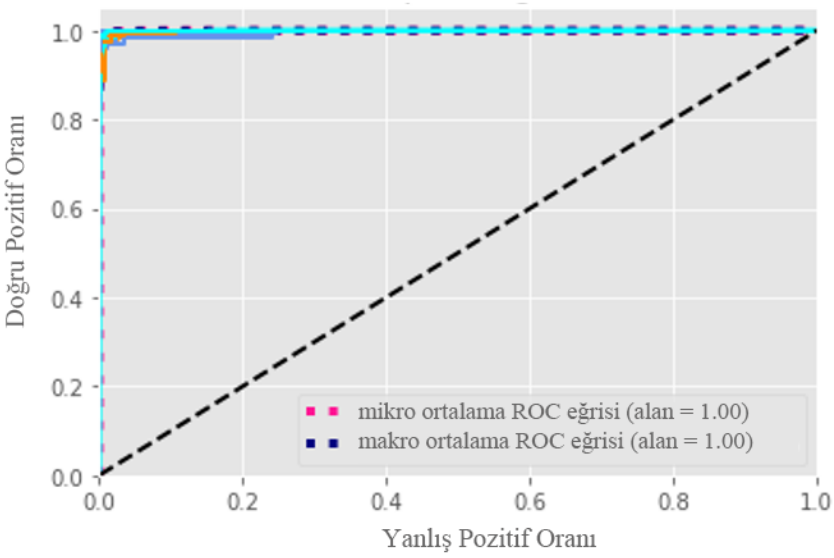
(i) CNN Çift Yönlü GRU modelinin eğitim ve doğrulama sonuç grafikleri.

Şekil 6.1. (Devam ediyor).

Çizelge 6.1 ve Şekil 6.1 incelendiğinde uygulanan modellerde eğitim süresine göre RNN tabanlı modellerin daha hızlı olduğu, ardından hibrit modellerin geldiği ve en yavaş modelin CNN tabanlı model olduğu görülmüştür. Doğruluk değerlerine göre yapılan kıyaslamada ise CNN ve Çift Yönlü GRU modellerinin kombinasyonu en başarılı model olarak görülmüştür. Bu modeli CNN ve Çift Yönlü LSTM, Çift Katmanlı Çift Yönlü LSTM, CNN GRU, Çift Yönlü GRU, Çift Katmanlı Çift Yönlü GRU, Çift Yönlü LSTM, CNN LSTM ve son olarak da CNN modelleri takip etmektedir. Bu sonuçlar çerçevesinde hazırlanan bitki TF protein veri seti için başarı sırasına göre ilk üç modelden ikisinin hibrit modeller olduğu görülmektedir. Doğruluk değerlerine göre yapılan kıyasta en başarılı model CNN Çift Yönlü GRU modeli, f-skor değerlerine göre yapılan kıyasta ise en başarılı model Çift Katlı Çift Yönlü LSTM modeli olarak görülmektedir. Modellerin veri setinin sınıfları üzerindeki sınıflandırma başarılarını görmek üzere tasarlanan modellerden seçilmiş ve en başarılı olmuş olanların ROC eğrisi grafikleri Şekil 6.2'de sunulmaktadır.

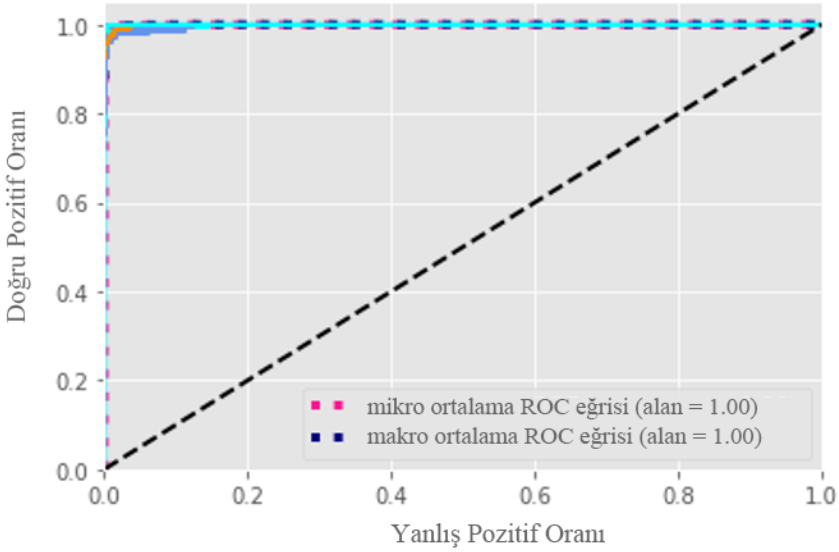


(a) Çift Yönlü LSTM modelinin ROC grafiği.

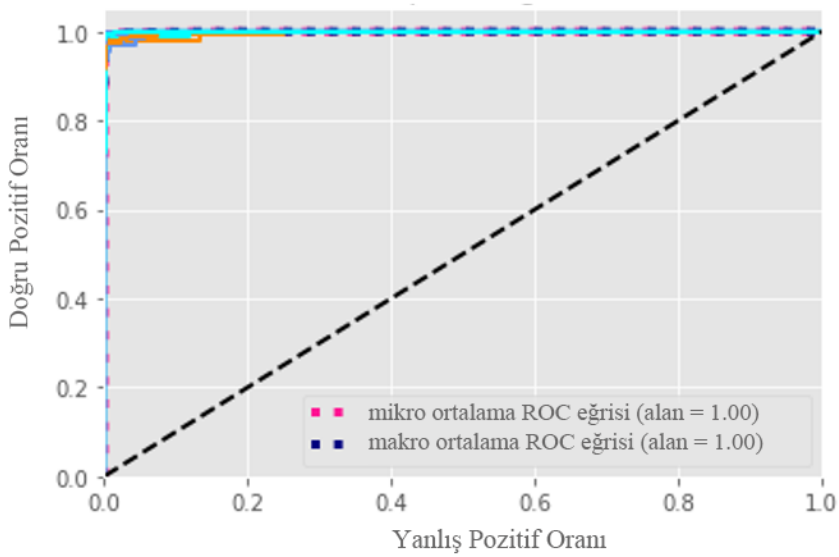


(b) Çift Katmanlı Çift Yönlü LSTM modelinin ROC grafiği.

Şekil 6.2. Seçilmiş modellerin ROC eğrisi grafikleri. a) Çift Yönlü LSTM, b) Çift Katmanlı Çift Yönlü LSTM, c) Çift Yönlü GRU, d) Çift Katmanlı Çift Yönlü GRU, e) CNN, f) CNN LSTM, g) CNN GRU, h) CNN Çift Yönlü LSTM, i) CNN Çift Yönlü GRU, j) CNN Çift Yönlü GRU (yakınlaştırılmış).

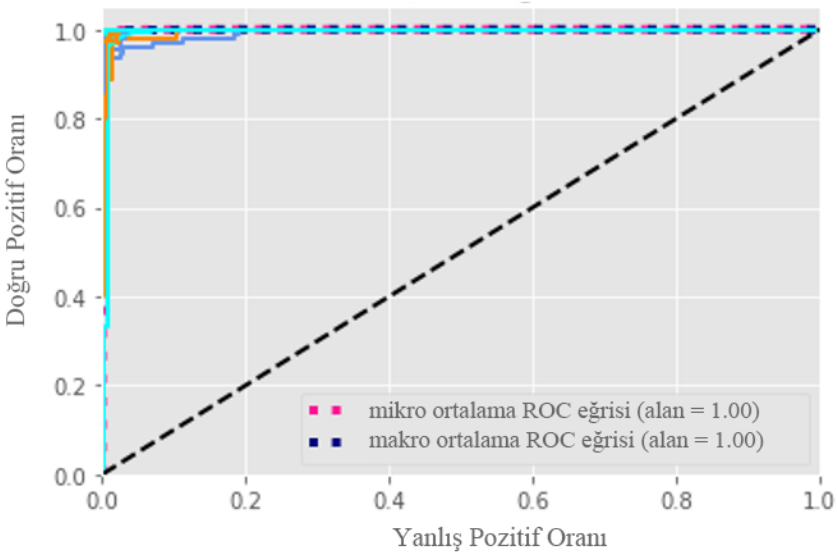


(c) Çift Yönlü GRU modelinin ROC grafiği.

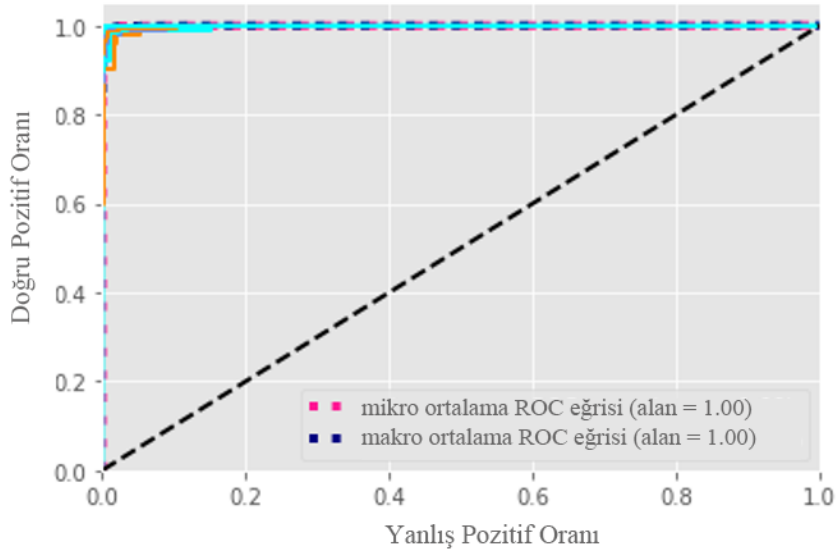


(d) Çift Katmanlı Çift Yönlü GRU modelinin ROC grafiği.

Şekil 6.2. (Devam ediyor).

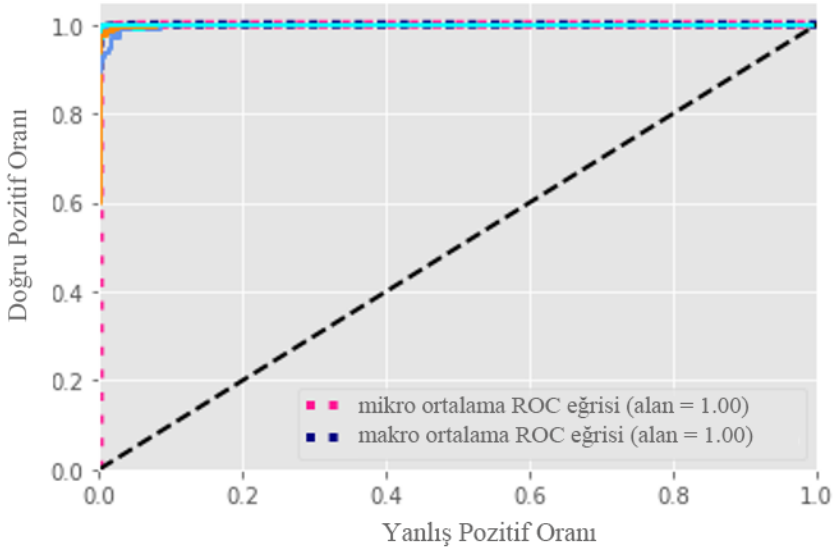


(e) CNN modelinin ROC grafiği.

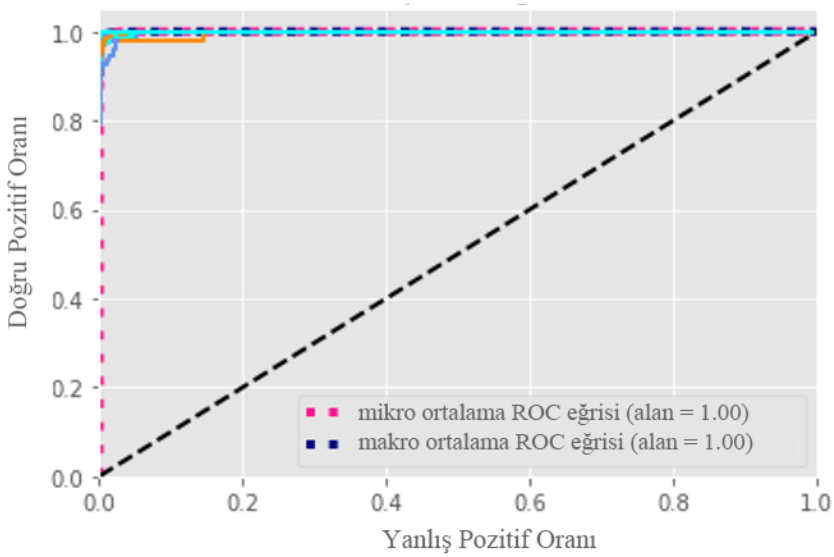


(f) CNN LSTM modelinin ROC grafiği.

Şekil 6.2. (Devam ediyor).

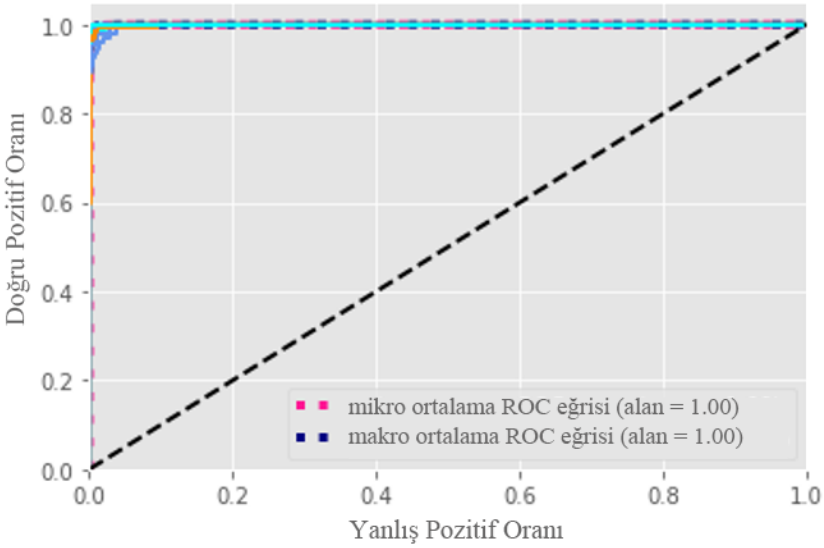


(g) CNN GRU modelinin ROC grafiği.

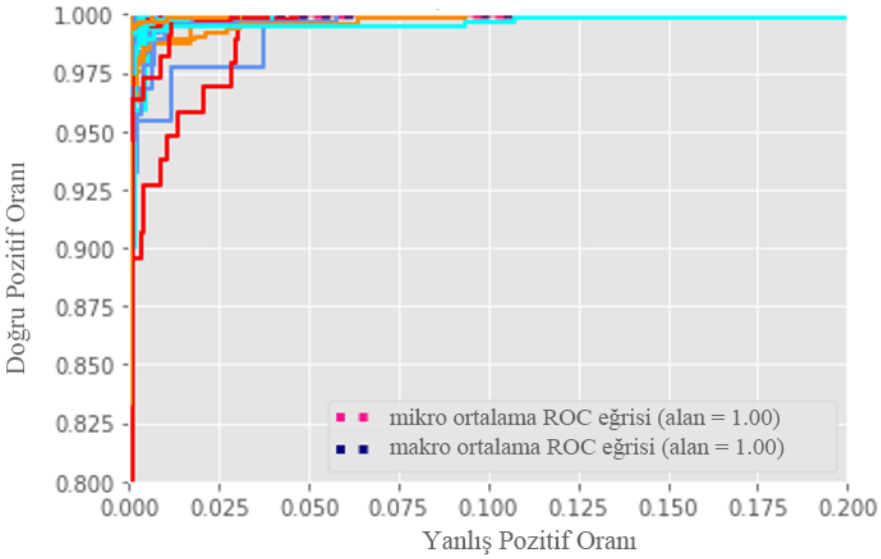


(h) CNN Çift Yönlü LSTM modelinin ROC grafiği.

Şekil 6.2. (Devam ediyor).



(i) CNN Çift Yönlü GRU modelinin ROC grafiği.



(j) CNN Çift Yönlü GRU modelinin ROC grafiği (yakınlaştırılmış).

Şekil 6.2. (Devam ediyor).

Şekil 6.2'deki grafiklerdeki her eğri bir sınıfı temsil etmektedir. Önerilen Çift Katlı Çift Yönlü LSTM modelinin (b) şeklinde verilen ROC grafiği ve CNN Çift Yönlü GRU modelinin (i) şeklinde verilen ROC eğrisi grafiği ve (j) şeklinde verilen grafiğin yakınlaştırılmış versiyonu incelendiğinde, 58 sınıf için ayrı ayrı olan 58 eğrinin de 1.0 değerine çok yakın olduğu, modelin tüm sınıfları oldukça yüksek bir başarı ile sınıflandırdığı görülmektedir. (j) şeklinde verilen, (i) şeklindeki grafiğin

yakınlaştırılmış versiyonunda her bir sınıfın eğrileri daha rahat bir şekilde görülmektedir.

Tasarlanan modeller içerisinden en başarılı parametrelere sahip seçilmiş modellerin scikit-learn kütüphanesindeki train_test_split aracı ile %80 eğitim, %10 doğrulama ve %10 test olarak rastgele ayrılmış veri setlerindeki başarılar görülmüştür. Yine aynı modellerin 10 kat çapraz doğrulama testleri yapılarak başarılarının ikinci bir doğrulaması yapılmıştır ve modellerin başrıları perçinlenmiştir. 10 kat çapraz doğrulama sonuçları Çizelge 6.2'de verilmiştir.

Çizelge 6.2. Tasarlanan modellerin 10-kat çapraz doğrulama sonuçları.

Model	**10 kat çapraz doğrulama sonuçları**
Çift Yönlü LSTM (M-1-3)	97.46
İki Katmanlı Çift Yönlü LSTM (M-2-4)	97.91
Çift Yönlü GRU (M-3-6)	97.57
İki Katmanlı Çift Yönlü GRU (M-4-4)	97.79
CNN (M-5-2)	97.25
CNN LSTM (M-6-3)	97.63
CNN GRU (M-7-3)	96.49
CNN Çift Yönlü LSTM (M-8-1)	97.90
CNN Çift Yönlü GRU (M-9-4)	**98.07**

Çizelge 6.1, Şekil 6.1, Şekil 6.2 ve Çizelge 6.2 incelendiğinde ortak bir sonuç olarak k-mer Word2Vec ön işleme adımına sahip CNN Çift Yönlü GRU hibrit derin öğrenme modelinin bu çalışma kapsamında hazırlanmış olan bitki TF protein veri seti için yine bu çalışma kapsamında hazırlanmış olan doğruluk oranı açısından en başarılı model olduğu görülmüştür. Ayrıca Çift Katlı Çift Yönlü LSTM modeli de f-skor değerine göre en başarılı model olarak görülmüştür. Bu iki model, bitki TF proteinlerin için bu çalışma kapsamında önerilmiş olan modellerdir. Bu önerilen Çift Katmanlı Çift Yönlü LSTM modelinin ve önerilen CNN Çift Yönlü GRU hibrit modelinin tasarımları sırasıyla Çizelge 6.3'te ve Çizelge 6.4'te verilmiştir.

Çizelge 6.3. Önerilen Çift Katmanlı Çift Yönlü LSTM derin öğrenme modelinin yapısı.

Katman	**Çıkış Boyutu**	**Param #**
Embedding	(None, 450, 300)	2784600
Bidirectional	(None, 450, 512)	1140736
Bidirectional	(None, 512)	1574912
Dense	(None, 58)	29754

Çizelge 6.4. Önerilen CNN Çift Yönlü GRU hibrit derin öğrenme modelinin yapısı.

Katman	**Çıkış Boyutu**	**Param #**
Embedding	(None, 250, 300)	2784600
Convolutional 1D (128)	(None, 250, 128)	115328
Max Pooling 1D (3)	(None, 83, 128)	0
Dropout (0.3)	(None, 83, 128)	0
Convolutional 1D (256)	(None, 83, 256)	98560
Max Pooling 1D (3)	(None, 27, 256)	0
Dropout (0.35)	(None, 27, 256)	0
Convolutional 1D (256)	(None, 27, 256)	196864
Max Pooling 1D (3)	(None, 9, 256)	0
Dropout (0.4)	(None, 9, 256)	0
Bidirectional GRU (384)	(None, 9, 768)	1479168
Dropout (0.25)	(None, 9, 768)	0
Flatten	(None, 6912)	0
Dense (128)	(None, 128)	884864
Dropout (0.45)	(None, 128)	0
Dense (58) (Classification)	(None, 58)	7482

Bitki TF proteinleri veri seti bu çalışma kapsamında üretilmiş yeni bir veri seti olduğundan literatürdeki en yüksek başarı oranlarından birine sahip olan ve farklı bir veri seti için önerilen bir çift yönlü LSTM modeli ve CNN tabanlı ResNet modeli [20] bu çalışmadaki bitki TF protein veri seti ile bu modellerin başarısı test edilmiştir. Bu modellerin başarsı, önerilen hibrit derin öğrenme modeline göre daha düşük bulunmuştur. Nitekim Bölüm 2'deki literatür araştırması incelendiğinde de benzer veri setleri kullanan modeller arasında hem en başarılı hem de k-mer Word2Vec ön işleme adımı ve CNN Çift Yönlü GRU hibrit yapısı ile en yeni nesil model olduğu açıkça görülmektedir. Karşılaştırma sonuçları Çizelge 6.5'te gösterilmektedir.

Çizelge 6.5. Önerilen bu çalışmadaki veri seti ile diğer yöntemlerin karşılaştırması.

Çalışma	**Model**	**Doğruluk (%)**	**Kesinlik (%)**	**Hassasiyet (%)**	**F-Skor (%)**
2022, Bileshi vd. [20]	Çift Yönlü LSTM	85.04	86.21	82.03	84.45
2022, Bileshi vd. [20]	ProtCNN	85.18	94.01	80.33	85.09
Önerilen tekli model (M-2-4)	Çift Katmanlı Çift Yönlü LSTM (k-mer ve Word2Vec)	97.80	**97.10**	**96.31**	**96.60**
Önerilen hibrit model (M-9-4)	CNN Çift Yönlü GRU (k-mer ve Word2Vec)	**98.23**	95.88	95.27	95.36

Çizelge 6.5'e ve Bölüm 2'de verilen çalışmalarda kullanılan veri setleri ve yöntemler farklı olduğu için doğrudan bir karşılaştırma yapmak mümkün değildir. TF proteinlerinin ve diğer proteinlerin sınıflandırılmasında istatistiksel modellerin ve farklı makine öğrenmesi modellerinin kullanıldığı görülmektedir. Bu modellerin bir kısmında dizilere ek olarak çeşitli tanımlayıcı numaraların, aile bilgilerinin veya dizi bilgilerinin verilmesi gerekmektedir. Yakın dönem çalışmalarında derin öğrenme çalışmaları yer alsa dahi bu çalışmalar çoğunlukla klasik kod sözlüğü veya tek sıcak kodlama ön işleme adımlarını içeren, yalnızca CNN veya yalnızca RNN veya RNN temelli LSTM mimarilerini kullanan tekli modellerdir. Birtakım çalışmalarda Word2Vec temelli ön işleme çalışmaları [17] olsa da bu çalışmada gerçekleştirilmiş olan Word2Vec ön işlemeli, CNN ve GRU mimarilerinin birleşiminden gücünü alan hibrit model yapısında bir çalışma literatürde yer almamaktadır. Bu çalışma hibrit yapısı ve bu hbrit yapı içeisinde yer alan kısımları ile literatüre katılmış olan önemli bir yenilik olarak ön plana çıkmaktadır.

Yine de benzer veri setleri ile çalışılmış olan bazı modeller ile bu çalışma kapsamında tasarlanmış olan Çift Katlı Çift Yönlü LSTM modeli ve hibrit model kıyaslanmış ve önerilen hibrit model hem en yüksek başarıya sahip olmuş, hem de Word2Vec + CNN + Çift Yönlü GRU üçlü hibrit yapısıyla literatüre açıkça bir yenilik katmıştır. Bu çalışmada önerilen tekli ve hibrit model ile benzer veri setleriyle çalışan literatürdeki diğer modellerin kıyaslaması Çizelge 6.6'da verilmiştir.

Çizelge 6.6. Önerilen hibrit model ile benzer veri setlerine sahip çalışmaların kıyaslanması.

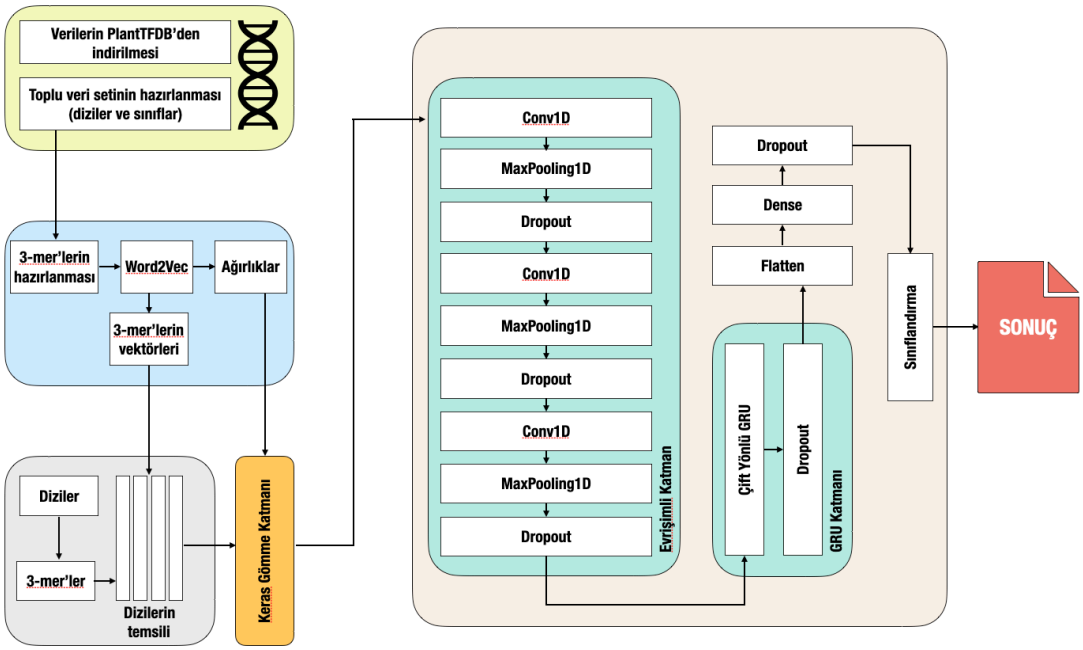
Yıl	Yazar	Yöntem	Veri Seti	Doğruluk (%)
2017	Li vd. [19]	Çift Yönlü LSTM	SCOPe	85.8
2019	Le vd. [18]	PSSM + CNN + GRU	Veziküler taşıma proteinleri	85.8
2019	Rao vd. [21]	3 Katmanlı Çift Yönlü LSTM, ResNet	Pfam	85
2019	Belzen vd. [22]	3 katmanlı ResNet	CAFA3	93.7
2020	Strodthoff vd. [7]	İleri+geri besleme, ön-eğit. CNN	EC40, EC50 enzim	98
2022	Bileschi vd. [20]	Çift Yönlü LSTM, ResNet	Pfam Seed	95.8
2022	Önerilen tekli model (M-2-4)	Word2Vec + 2 Katmanlı Çift Yönlü LSTM	Bitki TF Proteinleri (özgün)	97.80
2022	Önerilen hibrit model (M-9-4)	Word2Vec + CNN + Çift Yönlü GRU	Bitki TF Proteinleri (özgün)	**98.23**

Çizelge 6.6'da da görüldüğü gibi literatürdeki benzer veri setleri ile yapılmış çalışmalar ile bu tez kapsamında önerilmiş hibrit model kıyaslandığında model hem en yüksek başarıya sahiptir hem de üçlü hibrit yapısı ile literatüre önemli bir yenilik sağlamıştır. Çalışmalara detaylı olarak bakıldığında Li ve arkadaşlarının çalışmasında dizi boyutları 400 karakter olarak belirlenmiştir ve eğitim 150 epochta tamamlanmıştır. Çift yönlü LSTM katmanındaki ünite sayısı ise 40 olarak belirlenmiştir. LSTM ünite sayısı düşük kalmış, ayrıca model hem dizileri uzun kullandığı için daha fazla kaynak kullanımı gerekmiş hem de epoch değerini yüksek belirlediği için doğruluk oranı bu çalışmanın gerisinde kalmış ve eğitimini daha uzun dönemde tamamlamıştır. Bu sonuçlar da %85.8 doğruluğa sahip olan Li ve arkadaşlarının çalışmasını bu tez kapsamında önerilen hibrit modelin gerisinde bırakmıştır [19]. Le ve arkadaşlarının çalışmasında PSSM ve dizilerlerle beraber CNN ve GRU katmanları kullanılmıştır. Bu çalışmada dizilerin bulunduğu fasta dosyalarından PSSM'lerin üretimi yapılmıştır. Dizi benzerliklerinin kontrolü için ise BLAST kullanılmıştır. Bu ek bilgilerin gerekliliği ve ek analizlerin gerekliliği, ayrıca modelin %85.8 doğruluk oranı bu tez kapsamında önerilen hibrit model, Le ve arkadaşlarının çalışmasının önüne geçirmektedir [18]. Rao ve arkadaşlarının çalışmasında çift yönlü LSTM ve CNN tabanlı ResNet modelleri kullanılmıştır. Bu çalışmanın ön işleme adımında HMM tabanlı girdilerin kullanılması, ek işlem ihtiyacını beraberinde getirmiştir. Ayrıca ön işleme kısmında tek-sıcak kodlama kullanılması ve en çalışmanın en başarılı modeli olan 3 katlı çift yönlü LSTM

modelinin %85 olan doğruluk oranı bu modeli de bu çalışma kapsamında üretilen hibrit modelin gerisinde bırakmıştır [21]. Belzen ve arkadaşlarının çalışmasında 3 katmanlı bir ResNet modeli kullanılmıştır. Bu modelde ön işleme adımında tek-sıcak kodlama kullanılmıştır. Tek-sıcak kodlama sebebiyle amino asitler arası bağların yakalanamaması, ResNet mimarisinin yoğun katmanları sebebiyle gereken görece uzun eğitim süreleri ve %93.7 doğruluk oranı sebebiyle bu çalışmadaki hibrit model Belzen ve arkadaşlarının çalışmalarından daha başarılıdır [22]. Strodthoff ve arkadaşlarının çalışması ileri ve geri besleme ile önceden eğitilmiş bir CNN modeli içermektedir. Buradaki 7 katmanlı CNN modelinin yoğunluğu, PSSM özelliklerinin ek hesap maliyetleri ve %98 doğruluk oranı sebebiyle bu model de önerilen hibrit modelin gerisinde kalmıştır [7]. Bileschi ve arkadaşlarının çalışmasında ise kod sözlüğü ve tek-sıcak kodlama ön işleme yöntemleri kullanan bir çift yönlü LSTM modeli ve ayrıca bir de ResNet modeli üretilmiştir. Burada en yüksek doğruluk oranı olarak %95.8 elde edilmesi ve kod sözlüğü ve tek-sıcak kodlama ön işleme yöntemleri ile amino asitler arası ilişkilerin kaçırılması sebebi ile bu model de önerilen hibrit modelin gerisinde kalmıştır [20].

Önceki çalışmalara bakıldığında önerilen modelin Word2Vec ön işleme yöntemi ile amino asitler arası bağlantıyı ve yakınlığı yakalayabilmesi, hibrit yapı sayesinde daha az katmana sahip bir CNN mimarisi ile özellik çıkarımı yapılması ve bir çift yönlü GRU katmanı ile daha uzun vadeli bağımlılıkların daha az maliyetle tespit edilerek kısa bir eğitim süreci ile beraber literatürdeki benzer çalışmalara göre daha yüksek bir doğruluk oranı ile sınıflandırmanın yapılması literatüre katılan bir yenilik olarak açıkça göze çarpmaktadır. Ayrıca önerilen model, üçlü hibrit yapısı ile litertürde ben alandaki çalışmalar arasında bir ilke imza atmaktadır.

Şekil 6.3'te, önerilen CNN Çift Yönlü GRU hibrit derin öğrenme modeli ile hazırlanmış bu çalışmanın akış şeması verilmiştir. Bu şema, çalışmanın verilerin ham hali ile toplanmasından sınıflandırma sonucuna kadar ki tüm adımlarını şematik olarak içermekte ve çalışma hakkında bir özetleyici görevi üstlenmektedir.



Şekil 6.3. Çalışmanın akış şeması.

BÖLÜM 7

SONUÇLAR VE ÖNERİLER

Bu çalışmada, Bitki Transkripsiyon Faktörü Veritabanından çeşitli görevlerde rol oynayan bitki transkripsiyon faktörü proteinlerini sınıflandırmak için üç aşamalı bir hibrit derin öğrenme modeli geliştirilmiştir.

Girdi olarak, sadece bir protein dizisi alan model, biyolojik deneylerde ve bazı istatistiksel tabanlı veya makine öğrenmesi tabanlı modellerde birden fazla parametreye ihtiyaç duyulmasının aksine, proteinlerin sınıflandırılmasını verimli ve yüksek bir başarı oranı ile gerçekleştirmektedir. Model, bitki transkripsiyon faktör proteinleri alanındaki en başarılı hibrit modeldir ve benzer veri setleriyle çalışan modeller arasında en yüksek başarıya sahiptir. Önerilen modellerden Çift Katlı Çift Yönlü LSTM modeli, hazırlanan bitki transkripsiyon faktör proteinleri veri seti ile %97.80 test başarısına, %96.60 f-skor değerine ve %97.91 10-katlı çapraz doğrulama sonucuna ulaşmıştır. Diğer önerilen model olan CNN Çift Yönlü GRU hibrit modeli ise hazırlanan bitki transkripsiyon faktör proteinleri veri seti ile %98.23 test başarısına, %95.36 f-skor değerine ve %98.07 10-katlı çapraz doğrulama sonucuna ulaşmıştır. Önerilen Çift Katlı Çift Yönlü LSTM modeli, %96.60 en yüksek f-skor değeri ile literatürde en yüksek değere sahiptir. Önerilen hibrit model, Word2Vec temelli ön işleme kısmı, öznitelik çıkarımı için CNN katmanları ve uzun-kısa vadeli bağımlılıkların tespiti ile bitki TF protein ailelerinin tespiti için çift yönlü GRU katmanı ile üçlü hibrit bir yapı ile oluşturulmuştur ve bu üçlü hibrit yapı ile literatürde bir ilktir ve TF proteinlerinin sınıflandırılması alanındaki çalışmalarda da literatürde en yüksek doğruluk oranına sahiptir.

Bu çalışmada önerilen hibrit modelin geliştirilmesi sırasında kullanmak üzere hazırlanan bitki TF protein veri seti, devam çalışmalarında ve diğer protein dizisi sınıflandırma çalışmalarında kullanılmak üzere literatüre kazandırılmıştır. Ek olarak,

hazırlanan protein kelime-vektör gösterimi ve yakınlık değerleri, farklı protein sınıflandırma çalışmalarında kullanılmak üzere bir önceden eğitilmiş (pre-train) model olarak hazırlanmış ve literatüre kazandırılmıştır.

Özetle önerilen Çift Katlı Çift Yönlü LSTM modelinin tasarımında iki ayrı kısım kullanılmıştır. Bunlardan ilki kaliteli bir eğitim almak için dizileri 3-mer'e bölme, kelime vektör temsilleri oluşturma ve benzerliği gösteren ağırlıkları kaydetme kısmı; ikincisi ise iki katmana sahip bir LSTM katmanı ile nispeten uzun protein dizilerinin uzun-kısa dönem bağımlılıklarını belirleme ve sınıflandırma kısmıdır. Bir diğer önerilen model olan CNN Çift Yönlü GRU hibrit modelinin tasarımında ise üç ayrı kısım kullanılmıştır. Bunlardan birinci kısım kaliteli bir eğitim almak için dizileri 3-mer'e bölme, kelime vektör temsilleri oluşturma ve benzerliği gösteren ağırlıkları kaydetme kısmı; ikinci kısım, CNN ağlarının özellik çıkarma başarısını kullanarak özellik çıkarma kısmı ve üçüncü kısım, LSTM yapısında çalışan ancak LSTM'e göre daha hızlı olan ve sınıflandırma başarısını temel RNN modellerine göre tıpkı LSTM gibi önemli ölçüde artıran çift yönlü GRU katmanı ile nispeten uzun protein dizilerinin uzun-kısa dönem bağımlılıklarını belirleme ve sınıflandırma kısmıdır. Bu modellerle araştırmacılar başka bir bilgiye ihtiyaç duymadan model ile beraber protein dizilerini sorgulayabilecek, dizinin hangi protein ailesine ait olduğunu belirleyecek, diğer ailelere olan benzerliğini öğrenebilecek ve bunları saha çalışmalarında kullanabileceklerdir.

Bu sayede bir dizi biyolojik deneyle kaybedilecek zaman farklı çalışmalarda harcanabilecektir. Ayrıca bu hız, kolaylık ve zaman, birim zamanda daha fazla analiz ve araştırma yapılmasına olanak sağlayacaktır. Aynı şekilde böyle bir üçlü hibrit yapının bitki transkripsiyon faktörlerinin sınıflandırılmasında kullanılması literatüre kazandırılmış bir yenilik olarak öne çıkmaktadır. Ayrıca model sayesinde biyolojik deneylerde yapılabilecek insan veya solüsyon, makine veya madde kaynaklı hataların en aza indirilmesi hedeflenmiştir. Ek olarak bu çalışma sayesinde tamamlanan bir protein transkripsiyon faktörü veri seti, ilgili proteinlerin tespiti ve dolayısıyla ilgili alandaki biyolojik çalışmalarda ve biyoinformatik ve yapay zekâ çalışmalarında kullanılabilir.

Derin öğrenme modellerine ek olarak literatürdeki eksikliklerin analizi ile beraber yeni nesil bir bHLH bitki TF protein veritabanı hazırlanmıştır. Bu veritabanı içerisinde bitkilerde yer alan bHLH TF proteinleri ile alakalı genel ve teknik bilgiler, tanımlayıcı numaralar (NCBI ID, TF ID, Gene ID gibi), DNA dizileri ve protein dizileri yer almaktadır. Ayrıca veritabanını internet sitesinde uluslararası kullanıma sahip olan analiz ve hizalama araçlarından HMM ve BLAST araçları eklenmiş ve yeni nesil bir derin öğrenme protein sınıflandırma aracı eklenerek literatüre yeni nesil bir veritabanı ve analiz aracı katılmıştır. Bu açıdan da çalışma literatüre eklenen bir yenilik olarak ön plana çıkmaktadır.

Gelecekteki çalışmalar, üç aşamalı hibrit modeli daha kapsamlı bir veri seti veya bir sınıflandırma probleminde daha hızlı ve verimli çalışacak şekilde hazırlayabilir ve farklı genetik kaynaklarda çalışmaya uygun hale getirebilir. Önerilen yöntem, protein dizilerinin benzer yapıları sebebiyle farklı veri setlerinde de yüksek başarı gösterebilir. Farklı alemlerdeki (bitki, hayvan gibi) transkripsiyon faktörleri benzer yapılara sahip olduğundan, önerilen model diğer veri setlerinde de başarılı olacaktır. Farklı veri kümelerinde daha yüksek başarı elde etmek üzere ince ayar yapmak için yöntemin ön işleme bölümünde maksimum uzunluk ve diğer kısımlarda birkaç değişiklik faydalı olacaktır. Ayrıca model üzerinde yapılan optimizasyon ve ince ayar çalışmaları başarının geliştirilmesinde önemli bir rol oynayacaktır. Modelin adımları üzerindeki değişiklikler ve daha geniş çaplı veri setleri ile eğitim yapılması modeli daha üst seviyelere çıkartacaktır. Ek olarak, bu çalışma, gelecekteki bir transkripsiyon faktörü ile DNA bölgelerinin bağlantı tahmini ve bölge tespiti için temel sağlayacaktır. Ayrıca bitkilerdeki bHLH TF proteinleri için hazırlanan bu yeni nesil veritabanı diğer alemlerdeki bHLH TF proteinlerinin analizi ile beraber bir üst seviyeye taşınabilir ve yeni analiz araçları eklenebilir.

KAYNAKLAR

1. Petrey, D. and Honig, B., "Is protein classification necessary? Toward alternative approaches to function annotation", ***Current Opinion in Structural Biology***, 19 (3): 363–368 (2009).
2. Baldi, P. and Brunak, S., "Bioinformatics, Second Edition: The Machine Learning Approach", ***MIT Press***, Cambridge, 121–123 (2001).
3. Eddy, S. R., "Hidden Markov models", ***Current Opinion in Structural Biology***, 6 (3): 361–365 (1996).
4. Gromiha, M. M., "Protein Sequence Analysis", ***Protein Bioinformatics***, 29–62 (2010).
5. Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J., "Basic local alignment search tool", ***Journal of Molecular Biology***, 215 (3): 403–410 (1990).
6. Price, M. N., Wetmore, K. M., Waters, R. J., Callaghan, M., Ray, J., Liu, H., Kuehl, J. v, Melnyk, R. A., Lamson, J. S., Suh, Y., Carlson, H. K., Esquivel, Z., Sadeeshkumar, H., Chakraborty, R., Zane, G. M., Rubin, B. E., Wall, J. D., Visel, A., Bristow, J., Blow, M. J., Arkin, A. P., and Deutschbauer, A. M., "Mutant phenotypes for thousands of bacterial genes of unknown function", ***Nature***, 557 (7706): 503—509 (2018).
7. Strodthoff, N., Wagner, P., Wenzel, M., and Samek, W., "UDSMProt: universal deep sequence models for protein classification", ***Bioinformatics***, 36 (8): 2401–2409 (2020).
8. Naveenkumar, K. S., Mohammed Harun, B. R., Vinayakumar, R., and Soman, K. P., "Protein Family Classification using Deep Learning", ***BioRxiv***, 414128 (2018).
9. Du, X., Cai, Y., Wang, S., and Zhang, L., "Overview of deep learning", (2016).
10. Karin, M., "Too many transcription factors: positive and negative interactions", ***The New Biologist***, 2 (2): 126–131 (1990).
11. Latchman, D. S., "Transcription factors: An overview", ***The International Journal of Biochemistry & Cell Biology***, 29 (12): 1305–1312 (1997).
12. Huerta, M., Haseltine, F., Liu, Y., Downing, G., and Seto, B., "NIH working definition of bioinformatics and computational biology", (2000).
13. Shen, H. bin and Chou, K. C., "EzyPred: A top–down approach for predicting enzyme functional classes and subclasses", ***Biochemical and Biophysical Research Communications***, 364 (1): 53–59 (2007).
14. Cozzetto, D., Minneci, F., Currant, H., and Jones, D. T., "FFPred 3: feature-based function prediction for all Gene Ontology domains", ***Scientific Reports***, 6 (1): 31865 (2016).
15. Dalkiran, A., Rifaioglu, A. S., Martin, M. J., Cetin-Atalay, R., Atalay, V., and Doğan, T., "ECPred: a tool for the prediction of the enzymatic functions of protein sequences based on the EC nomenclature", ***BMC Bioinformatics***, 19 (1): 334 (2018).

16. Gong, Q., Ning, W., and Tian, W., "GoFDR: A sequence alignment based method for predicting protein functions", ***Methods***, 93: 3–14 (2016).
17. Asgari, E. and Mofrad, M. R. K., "Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics", ***PLoS ONE***, 10 (11): (2015).
18. Le, N. Q. K., Yapp, E. K. Y., Nagasundaram, N., Chua, M. C. H., and Yeh, H. Y., "Computational identification of vesicular transport proteins from sequences using deep gated recurrent units architecture", ***Computational and Structural Biotechnology Journal***, 17: 1245–1254 (2019).
19. Li, S., Chen, J., and Liu, B., "Protein remote homology detection based on bidirectional long short-term memory", ***BMC Bioinformatics***, 18 (1): 443 (2017).
20. Bileschi, M. L., Belanger, D., Bryant, D. H., Sanderson, T., Carter, B., Sculley, D., Bateman, A., DePristo, M. A., and Colwell, L. J., "Using deep learning to annotate the protein universe", ***Nature Biotechnology***, 40 (6): 932–937 (2022).
21. Rao, R., Bhattacharya, N., Thomas, N., Duan, Y., Chen, X., Canny, J., Abbeel, P., and Song, Y. S., "Evaluating Protein Transfer Learning with TAPE", ***Advances in Neural Information Processing Systems***, 32: 9689–9701 (2019).
22. Upmeier zu Belzen, J., Bürgel, T., Holderbach, S., Bubeck, F., Adam, L., Gandor, C., Klein, M., Mathony, J., Pfuderer, P., Platz, L., Przybilla, M., Schwendemann, M., Heid, D., Hoffmann, M. D., Jendrusch, M., Schmelas, C., Waldhauer, M., Lehmann, I., Niopek, D., and Eils, R., "Leveraging implicit knowledge in neural networks for functional dissection and engineering of proteins", ***Nature Machine Intelligence***, 1 (5): 225–235 (2019).
23. Torrisi, M., Pollastri, G., and Le, Q., "Deep learning methods in protein structure prediction", ***Computational and Structural Biotechnology Journal***, 18: 1301–1310 (2020).
24. Lim, S., Lu, Y., Cho, C. Y., Sung, I., Kim, J., Kim, Y., Park, S., and Kim, S., "A review on compound-protein interaction prediction methods: Data, format, representation and model", ***Computational and Structural Biotechnology Journal***, 19: 1541–1556 (2021).
25. Gustafsson, C., Minshull, J., Govindarajan, S., Ness, J., Villalobos, A., and Welch, M., "Engineering genes for predictable protein expression", ***Protein Expression and Purification***, 83 (1): 37–46 (2012).
26. Murre, C., McCaw, P. S., and Baltimore, D., "A new DNA binding and dimerization motif in immunoglobulin enhancer binding, daughterless, MyoD, and myc proteins", ***Cell***, 56 (5): 777–783 (1989).
27. Jones, S., "An overview of the basic helix-loop-helix proteins", ***Genome Biology***, 5 (6): 226 (2004).
28. Buck, M. J. and Atchley, W. R., "Phylogenetic Analysis of Plant Basic Helix-Loop-Helix Proteins", ***Journal of Molecular Evolution***, 56 (6): 742–750 (2003).
29. Serna, L., "BHLH proteins know when to make a stoma", ***Trends In Plant Science***, 12 (11): 483–485 (2007).
30. Pires, N. and Dolan, L., "Origin and Diversification of Basic-Helix-Loop-Helix Proteins in Plants", ***Molecular Biology and Evolution***, 27 (4): 862–874 (2010).
31. Pires, N. and Dolan, L., "Early evolution of bHLH proteins in plants", ***Plant Signaling & Behavior***, 5 (7): 911–912 (2010).

32. Zhao, F., Li, G., Hu, P., Zhao, X., Li, L., Wei, W., Feng, J., and Zhou, H., "Identification of basic/helix-loop-helix transcription factors reveals candidate genes involved in anthocyanin biosynthesis from the strawberry white-flesh mutant", ***Scientific Reports***, 8 (1): 2721 (2018).
33. Hudson, K. A. and Hudson, M. E., "A Classification of Basic Helix-Loop-Helix Transcription Factors of Soybean", ***International Journal of Genomics***, 2015: 1–10 (2015).
34. Yang, J., Gao, M., Huang, L., Wang, Y., van Nocker, S., Wan, R., Guo, C., Wang, X., and Gao, H., "Identification and expression analysis of the apple (Malus × domestica) basic helix-loop-helix transcription factor family", ***Scientific Reports***, 7 (1): 28 (2017).
35. Baxevanis, A. D., "The Importance of Biological Databases in Biological Discovery", ***Current Protocols in Bioinformatics***, 13 (1): 1.1.1-1.1.5 (2006).
36. Stein, L. D., "Integrating biological databases", ***Nature Reviews Genetics***, 4 (5): 337–345 (2003).
37. Attwood, T., Gisel, A., Eriksson, N.-E., and Bongcam-Rudloff, E., "Concepts, Historical Milestones and the Central Place of Bioinformatics in Modern Biology: A European Perspective", ***Intech Online Publishers. EScholarID:130613***, (2011).
38. Andreeva, A., Kulesha, E., Gough, J., and Murzin, A. G., "The SCOP database in 2020: expanded classification of representative family and superfamily domains of known protein structures", ***Nucleic Acids Research***, 48 (D1): D376–D382 (2020).
39. Kirsanov, D. D., Zanegina, O. N., Aksianov, E. A., Spirin, S. A., Karyagina, A. S., and Alexeevski, A. v, "NPIDB: Nucleic acid-Protein Interaction DataBase", ***Nucleic Acids Research***, 41 (Database issue): D517–D523 (2013).
40. Chen, J. X., "Guide to Graphics Software Tools", 2. Ed., ***Springer London***, London, (2009).
41. R, R. K., N S, N., S P, A., Sinha, D., Veedin Rajan, V. B., Esthaki, V. K., and D'Silva, P., "HSPIR: a manually annotated heat shock protein information resource", ***Bioinformatics (Oxford, England)***, 28 (21): 2853–2855 (2012).
42. Ryu, T., Jung, J., Lee, S., Nam, H. J., Hong, S. W., Yoo, J. W., Lee, D., and Lee, D., "BZIPDB : A database of regulatory information for human bZIP transcription factors", ***BMC Genomics***, 8 (1): 136 (2007).
43. Shih, M. der, Hoekstra, F. A., and Hsing, Y. I. C., "Late Embryogenesis Abundant Proteins", ***Advances in Botanical Research***, 48: 211–255 (2008).
44. Hunault, G. and Jaspard, E., "LEAPdb: a database for the late embryogenesis abundant proteins", ***BMC Genomics***, 11 (1): 221 (2010).
45. Norambuena, T. and Melo, F., "The Protein-DNA Interface database", ***BMC Bioinformatics***, 11 (1): 262 (2010).
46. Lewis, B. A., Walia, R. R., Terribilini, M., Ferguson, J., Zheng, C., Honavar, V., and Dobbs, D., "PRIDB: a Protein-RNA interface database", ***Nucleic Acids Research***, 39 (Database issue): D277–D282 (2011).
47. Shu, J. J., "A new integrated symmetrical table for genetic codes", ***Biosystems***, 151: 21–26 (2017).
48. WATSON, J. D. and CRICK, F. H. C., "Molecular Structure of Nucleic Acids: A Structure for Deoxyribose Nucleic Acid", ***Nature***, 171 (4356): 737–738 (1953).

49. Pradhan, P., Tirumala, S., Liu, X., Sayer, J. M., Jerina, D. M., and Yeh, H. J. C., "Solution Structure of a Trans-Opened (10S)-dA Adduct of (+)-(7S,8R,9S,10R)-7,8-Dihydroxy-9,10-epoxy-7,8,9,10-tetrahydrobenzo[a]pyrene in a Fully Complementary DNA Duplex: Evidence for a Major Syn Conformation", ***Biochemistry***, 40 (20): 5870–5881 (2001).
50. Acar, N., Gündeğer, E., and Selçuki, C., "Protein Yapı Analizleri", Biyoinformatik Temelleri ve Uygulamaları, 1. Ed., ***Pegem Akademi Yayıncılık***, Kastamonu, 85–128 (2018).
51. Schulte, K. W., Green, E., Wilz, A., Platten, M., and Daumke, O., "Structural Basis for Aryl Hydrocarbon Receptor-Mediated Gene Activation", ***Structure***, 25 (7): 1025-1033.e3 (2017).
52. Ferrier, D. R., "Protein Yapısı ve İşlevi", Lippincott Biyokimya: Görsel Anlatımlı Çalışma Kitapları, ***Nobel Tıp Kitapevleri***, İstanbul, 1–68 (2019).
53. Latchman, D. S., "Transcription factors: an overview Function of transcription factors", ***Int. J. Exp. Path***, 74: 417–422 (1993).
54. Karin, M., "Too many transcription factors: positive and negative interactions", ***The New Biologist***, 2 (2): 126–131 (1990).
55. Riaño-Pachón, D. M., Ruzicic, S., Dreyer, I., and Mueller-Roeber, B., "PlnTFDB: an integrative plant transcription factor database", ***BMC Bioinformatics***, 8 (1): 42 (2007).
56. Zhang, D., Li, G., and Wang, Y., "A genome-wide identification and analysis of basic helix-loop-helix transcription factors in cattle", ***Gene***, 626: 241–250 (2017).
57. Wang, Y., Wang, S., Tian, Y., Wang, Q., Chen, S., Li, H., Ma, C., and Li, H., "Functional Characterization of a Sugar Beet BvbHLH93 Transcription Factor in Salt Stress Tolerance", ***International Journal of Molecular Sciences Article***, (2021).
58. Ledent, V. and Vervoort, M., "The Basic Helix-Loop-Helix Protein Family: Comparative Genomics and Phylogenetic Analysis", ***Genome Research***, 11(5): 754-770 (2001).
59. Goodstein, D. M., Shu, S., Howson, R., Neupane, R., Hayes, R. D., Fazo, J., Mitros, T., Dirks, W., Hellsten, U., Putnam, N., and Rokhsar, D. S., "Phytozome: a comparative platform for green plant genomics", ***Nucleic Acids Research***, 40 (D1): D1178–D1186 (2012).
60. Jin, J., Tian, F., Yang, D.-C., Meng, Y.-Q., Kong, L., Luo, J., and Gao, G., "PlantTFDB 4.0: toward a central hub for transcription factors and regulatory interactions in plants", ***Nucleic Acids Research***, 45 (D1): D1040–D1045 (2017).
61. Internet: Python Software Foundation, "Python", **https://www.python.org/** (2019).
62. Internet: QIAGEN Group, "QIAGEN CLC Genomics Workbench", **https://digitalinsights.qiagen.com/products-overview/discovery-insights-portfolio/analysis-and-visualization/qiagen-clc-genomics-workbench/** (2019).
63. IUPAC-IUB Comm. on Biochem. Nomenclature, "A one-letter notation for amino acid sequences. Tentative rules", ***Biochemistry***, 7 (8): 2703–2705 (1968).

64. Ofer, D., Brandes, N., and Linial, M., "The language of proteins: NLP, machine learning & protein sequences", ***Computational and Structural Biotechnology Journal***, 19: 1750–1758 (2021).
65. Schuster-Böckler, B., Schultz, J., and Rahmann, S., "HMM Logos for visualization of protein families", ***BMC Bioinformatics***, 5 (1): 1–8 (2004).
66. Internet: Pfam, "Family: HLH (PF00010)", **http://pfam.xfam.org/family/pf00010** (2019).
67. Internet: National Library of Medicine, "National Center for Biotechnology Information", **https://ncbi.nlm.nih.gov/** (2019).
68. Yang, K. K., Wu, Z., Bedbrook, C. N., and Arnold, F. H., "Learned protein embeddings for machine learning", ***Bioinformatics***, 34 (15): 2642–2648 (2018).
69. Ay Karakuş, B., Talo, M., Hallaç, İ. R., and Aydin, G., "Evaluating deep learning models for sentiment classification", ***Concurrency and Computation: Practice and Experience***, 30 (21): 1–14 (2018).
70. Vries, J. K., Liu, X., and Bahar, I., "The relationship between N-gram patterns and protein secondary structure", ***Proteins: Structure, Function, And Bioinformatics***, 68 (4): 830–838 (2007).
71. Vries, J. K. and Liu, X., "Subfamily specific conservation profiles for proteins based on n-gram patterns", ***BMC Bioinformatics***, 9 (1): 72 (2008).
72. Mikolov, T., Chen, K., Corrado, G., and Dean, J., "Efficient Estimation of Word Representations in Vector Space", (2013).
73. Řehuřek, R. and Sojka, P., "Software Framework for Topic Modelling with Large Corpora", (2010).
74. LeCun, Y., Bengio, Y., and Hinton, G., "Deep learning", ***Nature***, 521 (7553): 436–444 (2015).
75. Greff, K., Srivastava, R. K., Koutnik, J., Steunebrink, B. R., and Schmidhuber, J., "LSTM: A Search Space Odyssey", ***IEEE Transactions on Neural Networks and Learning Systems***, 28 (10): 2222–2232 (2017).
76. Gers, F. A., Schmidhuber, J., and Cummins, F., "Learning to Forget: Continual Prediction with LSTM", ***Neural Computation***, 12 (10): 2451–2471 (2000).
77. van Houdt, G., Mosquera, C., and Nápoles, G., "A review on the long short-term memory model", ***Artificial Intelligence Review***, 53 (8): 5929–5955 (2020).
78. Hochreiter, S. and Schmidhuber, J., "Long Short-Term Memory", ***Neural Computation***, 9 (8): 1735–1780 (1997).
79. Gao, Y. and Glowacka, D., "Deep Gate Recurrent Neural Network", (2016).
80. Zhang, A., Lipton, Z. C., Li, M., and Smola, A. J., "Dive into Deep Learning", ***ArXiv Preprint ArXiv:2106.11342***, (2021).
81. Şeker, A., Diri, B., and Balık, H. H., "Derin Öğrenme Yöntemleri ve Uygulamaları Hakkında Bir İnceleme", ***Gazi Mühendislik Bilimleri Dergisi***, 3 (3): 47–64 (2017).
82. Hubel, D. H. and Wiesel, T. N., "Receptive fields and functional architecture of monkey striate cortex", ***The Journal of Physiology***, 195 (1): 215–243 (1968).
83. Fukushima, K. and Miyake, S., "Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Visual Pattern Recognition", 267–285 (1982).

84. LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L., "Handwritten Digit Recognition with a Back-Propagation Network", (1989).
85. Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P., "Gradient-based learning applied to document recognition", ***Proceedings of the IEEE***, 86 (11): 2278–2324 (1998).
86. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., and Chen, T., "Recent advances in convolutional neural networks", ***Pattern Recognition***, 77: 354–377 (2018).
87. Liu, Y. H., "Feature Extraction and Image Recognition with Convolutional Neural Networks", ***Journal of Physics: Conference Series***, 1087: 062032 (2018).
88. Başaran, E., "Timpanik Membran Görüntü Analizi ve Yapay Zeka Kullanarak Sanal Otitis Media Tanı Sisteminin Geliştirilmesi", Doktora, ***Karabük Üniversitesi Lisansüstü Eğitim Enstitüsü***, Karabük, (2020).
89. Yin, W., Kann, K., Yu, M., SchützeSch, H., and Munich, L., "Comparative Study of CNN and RNN for Natural Language Processing", ***ArXiv Preprint ArXiv:1702.01923***, (2017).
90. Yamashita, R., Nishio, M., Do, R. K. G., and Togashi, K., "Convolutional neural networks: an overview and application in radiology", ***Insights into Imaging***, 9 (4): 611–629 (2018).
91. Uyar, K., "Konvolüsyonel Sinir Ağlarında Ağ Eğitiminin İyileştirilmesi", Doktora, ***Selçuk Üniversitesi Fen Bilimleri Enstitüsü***, Konya, (2022).
92. Eşref, Y., "Türkçe Dizi Etiketleme için Sinir Ağ Modelleri", Yüksek Lisans, ***Hacettepe Üniversitesi Fen Bilimleri Enstitüsü***, Ankara, (2019).
93. Başaran, E., Cömert, Z., and Çelik, Y., "Convolutional neural network approach for automatic tympanic membrane detection and classification", ***Biomedical Signal Processing and Control***, 56: 101734 (2020).
94. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R., "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", ***Journal of Machine Learning Research***, 15 (56): 1929–1958 (2014).
95. ZahediNasab, R. and Mohseni, H., "Neuroevolutionary based convolutional neural network with adaptive activation functions", ***Neurocomputing***, 381: 306–313 (2020).
96. Ozcan, T. and Basturk, A., "Static Image-Based Emotion Recognition Using Convolutional Neural Network", (2019).
97. Krause, J., Cordeiro, J., Parpinelli, R. S., and Lopes, H. S., "A Survey of Swarm Algorithms Applied to Discrete Optimization Problems", Swarm Intelligence and Bio-Inspired Computation, ***Elsevier***, 169–191 (2013).
98. Yang, J. and Yang, G., "Modified Convolutional Neural Network Based on Dropout and the Stochastic Gradient Descent Optimizer", ***Algorithms***, 11 (3): 28 (2018).
99. Internet: Kızrak, A., "Derin Öğrenme İçin Aktivasyon Fonksiyonlarının Karşılaştırılması", **https://ayyucekizrak.medium.com/derin-ogrenme-icin-aktivasyon-fonksiyonlarinin-karsilastirilmasi-cee17fd1d9cd** (2021).
100. Baccelli, G., Stathis, D., Hemani, A., and Martina, M., "NACU: A Non-Linear Arithmetic Unit for Neural Networks", (2020).

101. Parisi, L., Neagu, D., Ma, R., and Campean, F., "Quantum ReLU activation for Convolutional Neural Networks to improve diagnosis of Parkinson's disease and COVID-19", ***Expert Systems with Applications***, 187: 115892 (2022).
102. Nair, V. and Hinton, G. E., "Rectified linear units improve restricted boltzmann machines", (2010).
103. Basturk, A., Yuksei, M. E., Badem, H., and Caliskan, A., "Deep neural network based diagnosis system for melanoma skin cancer", (2017).
104. Yi, D., Lei, Z., Liao, S., and Li, S. Z., "Deep Metric Learning for Person Re-identification", (2014).
105. Chen, W., Tsutsumi, A., Lin, H., and Otawara, K., "Modeling nonlinear dynamics of circulating fluidized beds using neural networks", ***China Particuology***, 3 (1–2): 84–89 (2005).
106. Liang, J., Xu, Y., Bao, C., Quan, Y., and Ji, H., "Barzilai–Borwein-based adaptive learning rate for deep learning", ***Pattern Recognition Letters***, 128: 197–203 (2019).
107. Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., and Wilson, A. G., "Averaging Weights Leads to Wider Optima and Better Generalization", (2018).
108. Internet: Çarkacı, N., "Derin Öğrenme Uygulamalarında En Sık Kullanılan Hiper-Parametreler", **https://medium.com/deep-learning-turkiye/derin-ogrenme-uygulamalarinda-en-sik-kullanilan-hiper-parametreler-ece8e9125c4** (2020).
109. YAZAN, E. and Talu, M. F., "Comparison of the stochastic gradient descent based optimization techniques", (2017).
110. "Encyclopedia of Machine Learning", ***Springer US***, Boston, MA, (2010).
111. Luque, A., Carrasco, A., Martín, A., and de las Heras, A., "The impact of class imbalance in classification performance metrics based on the binary confusion matrix", ***Pattern Recognition***, 91: 216–231 (2019).
112. Ozenne, B., Subtil, F., and Maucort-Boulch, D., "The precision–recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases", ***Journal of Clinical Epidemiology***, 68 (8): 855–859 (2015).
113. KILIC, S., "ROC Analysis in Clinical Decision Making", ***Journal of Mood Disorders***, 3 (3): 135 (2013).
114. Flamholz, Z. N., Crane-Droesch, A., Ungar, L. H., and Weissman, G. E., "Word embeddings trained on published case reports are lightweight, effective for clinical tasks, and free of protected health information", ***Journal of Biomedical Informatics***, 125: 103971 (2022).
115. Rohani, A., Taki, M., and Abdollahpour, M., "A novel soft computing model (Gaussian process regression with K-fold cross validation) for daily and monthly solar radiation forecasting (Part: I)", ***Renewable Energy***, 115: 411–422 (2018).
116. Xiong, Z., Cui, Y., Liu, Z., Zhao, Y., Hu, M., and Hu, J., "Evaluating explorative prediction power of machine learning algorithms for materials discovery using k-fold forward cross-validation", ***Computational Materials Science***, 171: 109203 (2020).
117. Oncul, A. B., Celik, Y., Unel, N. M., and Baloglu, M. C., "Bhlhdb: A next generation database of basic helix loop helix transcription factors based on deep learning model", ***Journal of Bioinformatics And Computational Biology***, (2022).

118. Internet: SQLite Team, "Most Widely Deployed and Used Database Engine", **https://www.sqlite.org/mostdeployed.html** (2020).
119. Internet: SQLite Team, "About SQLite", **https://sqlite.org/about.html** (2020).
120. Internet: Django Foundation, "Meet Django", **https://www.djangoproject.com/** (2021).
121. Wu, C., Gao, R., Zhang, Y., and de Marinis, Y., "PTPD: predicting therapeutic peptides by deep learning and word2vec", ***BMC Bioinformatics***, 20 (1): 456 (2019).
122. Kumar, J., Goomer, R., and Singh, A. K., "Long Short Term Memory Recurrent Neural Network (LSTM-RNN) Based Workload Forecasting Model For Cloud Datacenters", ***Procedia Computer Science***, 125: 676–682 (2018).
123. Eddy, S. R., "Accelerated Profile HMM Searches", ***PLoS Computational Biology***, 7 (10): e1002195 (2011).
124. NCBI Resource Coordinators, "Database resources of the National Center for Biotechnology Information", ***Nucleic Acids Research***, 44 (D1): D7–D19 (2016).
125. Internet: NCBI Resource Coordinators, "Basic Local Alignment Search Tool", **https://blast.ncbi.nlm.nih.gov/Blast.cgi** (2019).

ÖZGEÇMİŞ

Ali Burak ÖNCÜL ilk ve orta öğrenimini Amasya'da tamamladı. Amasya Anadolu Lisesi'nin Matematik-Fen Bölümü'nden mezun oldu. 2009 yılında Ondokuz Mayıs Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü'nde öğrenime başlayıp 2013 yılında mezun oldu. 2013 yılında Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalında yüksek lisans eğitimine başladı ve 2016 yılının başında tamamladı. 2015 yılının başında Kastamonu Üniversitesi Mühendislik ve Mimarlık Fakültesi Bilgisayar Mühendisliği Bölümünde araştırma görevlisi olarak göreve başladı. 2016 yılında Karabük Üniversitesi Fen Bilimleri Enstütüsü Bilgisayar Mühendisliği Anabilim Dalında doktora eğitimine başladı. Halen Kastamonu Üniversitesi Mühendislik ve Mimarlık Fakültesi Bilgisayar Mühendisliği Bölümünde araştırma görevlisi olarak çalışmaya devam etmektedir.