T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

COVID-19 MUTASYONLARININ TESPİTİNDE YAPAY ZEKA TABANLI ALGORİTMALARIN KULLANILMASI

DOKTORA TEZİ

Mehmet BURUKANLI

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Bilim Dalı

TEMMUZ 2024

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

COVID-19 MUTASYONLARININ TESPİTİNDE YAPAY ZEKA TABANLI ALGORİTMALARIN KULLANILMASI

DOKTORA TEZİ

Mehmet BURUKANLI

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Bilim Dalı

Tez Danışmanı: Prof. Dr. Nejat YUMUŞAK

TEMMUZ 2024

Mehmet Burukanlı tarafından hazırlanan "COVID-19 Mutasyonlarının Tespitinde Yapay Zeka Tabanlı Algoritmaların Kullanılması" adlı tez çalışması 10.07.2024 tarihinde aşağıdaki jüri tarafından oy birliği ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Bilim Dalı'nda Doktora tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı :	**Prof. Dr. Nejat YUMUŞAK (**Danışman**)** Sakarya Üniversitesi	.....
Jüri Üyesi :	**Prof. Dr. Devrim AKGÜN** Sakarya Üniversitesi	.....
Jüri Üyesi :	**Prof. Dr. Cüneyt BAYILMIŞ** Sakarya Üniversitesi	.....
Jüri Üyesi :	**Prof. Dr. Metin VARAN** Sakarya Uygulamalı Bilimleri Üniversitesi	.....
Jüri Üyesi :	**Doç. Dr. Halit ÖZTEKİN** Sakarya Uygulamalı Bilimler Üniversitesi	.....

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum "COVID-19 MUTASYONLARININ TESPİTİNDE YAPAY ZEKA TABANLI ALGORİTMALARIN KULLANILMASI" başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

(10/07/2024).

Mehmet Burukanlı

Eşime ve çocuklarıma

TEŞEKKÜR

Tez çalşıması süresince, her türlü desteğini benden esirgemeyen ve her türlü konuda bana destek veren danışman hocam Sayın Prof. Dr. Nejat YUMUŞAK'a en içten dileklerimle teşekkürlerimi sunuyorum.

Bu günlere gelmemde büyük emekleri olan çok değerli aileme şükranlarımı borç bilirim. Ayrıca, tez çalışması boyunca her daim yanımda olan sevgili eşime teşekkür ediyorum.

Mehmet Burukanlı

İÇİNDEKİLER

Sayfa

KISALTMALAR

ACE-2 : Anjiyotensin Dönüştürücü Enzim 2

ARDS : Akut Solunum Sıkıntısı Sendromu

AUC : Alıcı İşletim Karakteristik Eğrisi Altındaki Alan

AVI : Grip Antijenik Varyantları (Antigenic Variants of Influenza)

CD : Konektör Alanı

CH : Merkezi Helis

CNN : Evrişimsel Sinir Ağı (Convolutional Neural Network)

COVİD-19 : Koronavirüs Hastalığı 2019 (COVID-19)

CT : Bilgisayar Tomografisi (Computer Tomography)

CT : Sitoplazmik Kuyruk

CTD1 : C-Terminal Alanı 1

CTD2 : C-Terminal Alanı 2

CXR : Göğüs Röntgeni (Chest X-ray)

CXRVN : Göğüs Röntgeni COVID Ağı adı verilen CXR görüntüleri (CXR images called Chest X-Ray COVID Network)

DCNN : Derin Evrişimli Sinir Ağı (Deep Convolutional Neural Network)

DNA : Deoksiribonükleik Asit

DSÖ : Dünya Sağlık Örgütü (World Health Organization)

DT : Karar Ağacı (Decison tree)

ET : Ekstra Ağaç (Extra Tree)

FFN : İleri Beslemeli Ağ (Feed Forward Network)

FN : Yanlış Negatif (False Negative)

FP : Fusion Peptidi

FP : Yanlış Pozitif (False Positive)

FPPR : Füzyon-Peptit Proksimal Bölgesi

GB : Gradyan Arttırma (Gradient Boosting)

GELU : Gauss Hatası Doğrusal Birimi (Gaussian Error Linear Unit)

GRU : Kapılı Tekrarlayan Birim (Gated Recurrent Unit)

GSEN : Gri Ölçekli Uzaysal Kullanım Ağı (Gray-Scale Spatial Exploitation Net)

HA : Hemaglutinin (Hemagglutinin)

HR1 : Heptad Tekrarı 1

HR2 : Heptad Tekrarı 2

KNN : K-En Yakın Komşu (K-Nearest Neighbor)

LR : Lojistik Regresyon (Logistic Regression)

LSTM : Uzun Kısa Süreli Bellek (Long Short-Term Memory)

MCC : Matthews Korelasyon Katsayısı (Matthews Correlation Coefficient)

MERS : Orta Doğu Solunum Sendromu Koronavirusü

MHA : Çoklu Kafalı Dikkat (Multi Head Attention)

MLP : Çok Katmanlı Algılayıcı (Multi Layer Perceptron)

MLP-Mixer : Çok Katmanlı Algılayıcı Karıştırıcısı (Multi Layer Perceptron Mixer)

MSA : Çoklu Dizi Hizalama (Multiple Sequence Alignment)

NCBI : Ulusal Biyoteknoloji Bilgi Merkezi

NLP : Doğal Dil İşleme (Natural Language Processing)

NN : Sinir Ağı

NSCT : Alt Örneklenmemiş Konturlet Dönüşümü (Non Subsampled Contourlet Transform)

NTD : N-Terminal Alanı

RBD : Reseptör Bağlama Alanı

RELU : Düzeltilmiş Doğrusal Birim (Rectified Linear Unit)

RF : Rastgele Orman (Random forest)

RNA : Ribonükleik Asit

RNN : Tekrarlayan Sinir Ağı (Recurrent Neural Network)

SARS-CoV-1 : Şiddetli Akut Solunum Sendromu Koronavirüs 1

SARS-CoV-2 : Şiddetli Akut Solunum Sendromu Koronavirüs 2

SP : Sinyal Peptidi

SPM : Sıralı Desen Madenciliği (Sequential Pattern Mining)

SVM : Destek Vektör Makinesi (Support Vector Machine)

TM : Transmembran Alanı

TMPRSS2S2 : Transmembran Serin Proteaz-2

TN : Gerçek Negatif (True Negative)

TP : Gerçek Pozitif (True Positive)

XGBoost : Aşırı Gradyan Artırma (eXtreme Gradient Boosting)

YSA : Yapay Sinir Ağları (Artificial neural networks)

SİMGELER

Q	: Sorgu (Query)
K	: Anahtar (Key)
V	: Değer (Value)
W	: Ağırlık Matrisi (Weight Matrix)
h	: Paralel Çalışan Ölçekli Nokta Ürün Dikkat Katmanlaının Sayısı (Transformer Mimarisi İçin)
$\boldsymbol{x}$	: Giriş vektörü
$\boldsymbol{x_a}$	: x Giriş Dizisinin a.ncı Değeri
$\boldsymbol{x_b}$	: x Verilerindeki Diğer Dizileri
$\boldsymbol{G}$	: x Dizisinin Boyutu
$\boldsymbol{LF}$	: Kayıp Fonksiyonu (Loss Function)
$\boldsymbol{y_t}$	: İkili Sınıflandırmada t.nci Zamandaki Gerçek Değerleri
$\dot{\boldsymbol{y}}_{\boldsymbol{t}}$	: İkili Sınıflandırmada t.nci Zamandaki Tahmin Edilen Değerleri
$\boldsymbol{F}$	: Seçilen Kalıntı Bölgelerinin (Residue Sites) Kümesi
$\boldsymbol{D^{(i)}}$	: COVİD-19 Mutasyon Tahmini İçin i.nci Eğitim Örneklerindeki Seçilen Pozisyonlarının Sayısı
$\boldsymbol{\Sigma}$	: Toplam Sembolü
b	: Bias Değeri
log	: Logaritma Fonksiyonu
$\boldsymbol{h_i}$	: Hiper Ağ (Hypernetwork) Fonksiyonu (HyperMixer Mimarisi İçin)
N	: Giriş Dizisindeki Jetonların (Tokens) Sayısı
$\boldsymbol{d}$	: Tokenların Boyutu
$\boldsymbol{d'}$	: Gizli Katman Sayısı (Hidden Size)
$\boldsymbol{\sigma}$	: Aktivasyon Fonksiyonu
$\mathbb{R}$	: Gerçel (Real) Sayılar
M	: Giriş Dizisinin Değişken (Variable) Boyutu
$\boldsymbol{p}$	: Ek Bilgiyi (Positional) Kodlayan Bir Vektör
O	: O Notasyonu

TABLO LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

COVID-19 MUTASYONLARININ TESPİTİNDE YAPAY ZEKA TABANLI ALGORİTMALARIN KULLANILMASI

ÖZET

Koronavirüs hastalığı 2019 (COVİD-19) virüsü, son zamanlarda ortaya çıkan ve bulaşılıcılığı oldukça yüksek olan ölümcül bir koronavirüs türüdür. COVİD-19 virüsünün hızlı yayılması, insanlar arasında büyük korku ve paniğe neden olmuştur. Ülkeler, COVİD-19 virüsü ile mücadele etmek için tam kapanma, sokağa çıkma yasağı gibi bazı önlemler almak zorunda kalmışlardır. Fakat bu alınan önlemlere ragmen COVİD-19 virüsü yayılmaya devam etmiştir. COVİD-19 virüsü ile mücadele etmenin başka biri yöntemi ise aşı ve ilaçların geliştirilmesidir. COVİD-19 virüsüyle mücadelede aşı ve ilaçların geliştirilmesi büyük önem taşımaktadır. Geliştirilen bu aşı ve ilaçların etkinliği, COVİD-19 virüsünün mutasyona uğraması sonucu ya önemli oranda azalmış yada tamamen yok olmuştur. Bu nedenle, COVİD-19 mutasyonlarıyla mücadele etmek oldukça önemlidir. COVİD-19 virüsünün yapısında gelecekte meydana gelebilecek mutasyonlar önceden tahmin edilebilirse aşı ve ilaçlar daha kolay geliştirilebilir. Böylece enfekte olan alanlar karantinaya alınabilecek ve sonuçta COVİD-19 virüsüyle mücadele daha kolay olabilecektir. Yapay zeka tabanlı yaklaşımlar COVİD-19 virüsü tespitinde de umut verici sonuçlar sunmaktadır. Literatür incelendiğinde COVİD-19 virüsü ile ilgili gerçekleştirilen çalışmaların geneli COVİD-19 virüsünün diğer yönleri ile ilgili çalışmalardır. Bu nedenle literatürde COVİD-19 virüsünün mutasyon tahmin edilmesi açısından ciddi boşluk bulunmaktadır. Bu tez çalışmasında biz bu boşluğu bir nebze olsun doldurmayı amaçladık. Bu tez çalışmasında, COVİD-19 virüsü yapısında gelecekte meydana gelebilecek mutasyonları tahmin etmek için yapay zeka tabanlı üç model (TfrAdmCov, StackGridCov ve HyperAttCov) önerilmiştir.

İlk önerilen TfrAdmCov modeli, adam optimizasyon algoritmasına sahip tamamen transformer kodlayıcı tabanlıdır. Önerilen TfrAdmCov model ile giriş dizisindeki değişkenler arasındaki bağımlılıklar kolay bir şekilde yakanalabilmektedir. Önerilen TfrAdmCov modeli, transformer tabanlı olması sebebiyle, aynı anda paralel hesaplama yapabilmektedir. Ayrıca, önerilen TfrAdmCov modelinin performansını arttırmak için eğitim, test ve Kfold veri setlerini oluşturma aşamasında agglomerative kümeleme algoritması tercih edilmiştir. Ek olarak, makine öğrenmesi algoritmalarının en iyi hyperparametre değerlerinin ayarlamak için GridSearchCV algoritmasında faydalanılmıştır. Deneysel sonuçlar detaylı olarak incelendiğinde, önerilen TfrAdmCov modelinin hem klasik yapay zeka tabanlı modellerden hem de birkaç son teknoloji modellerden daha iyi performans elde ettiğini göstermiştir. Önerilen TfrAdmCov modeli, COVİD-19 test veri seti üzerinde %99.93 doğruluk değerine, %100.00 kesinlik değerine, %97.38 hassasiyet değerine, %98.67 F1-skor değerine ve %98.65 MCC değerine ulaşmıştır. Benzer şekilde 10 rastgele deneminin ortalaması alındığında da, önerilen TfrAdmCov modeli, COVİD-19 test veri seti üzerinde %99.924 ile doğruluk, %97.18 ile hassasiyet, %98.57 ile F1-skor ve %98.54 ile MCC değeri açısından diğer modellerden daha iyi sonuçlar elde etmiştir. Önerilen

TfrAdmCov modeli ile derin öğrenme modellerinin istatistiksel açıdan kıyaslamak için farklı rastgele tohumlarla 10 rastgele denemenin ortalaması alınarak elde elde edilen sonuçlar analiz edilmiştir. Ortalama, standart sapma, medyan, min ve maks gibi istatistiksel ölçümler kullanılarak her model için doğruluk, kesinlik, hatırlama, F1-skor ve MCC performans ölçüm metriği açısından detaylı değerlendirme gerçekleştirilmiştir. Ayrıca, önerilen TfrAdmCov modelinin performansını değerlendirmek için influenza A/H3N2 HA veri seti üzerinde mutasyon tahmini gerçekleştirilmiştir. Önerilen TfrAdmCov modeli, H3N2 HA test veri seti üzerinde %96.33 doğruluk, %81.55 kesinlik, %52.33 hassasiyet, %63.75 F1-skor ve %63.61 MCC değerlerinde diğer modellere göre daha iyi sonuçlar elde etmiştir. İnfluenza H3N2 HA test veri seti üzerindeki sonuçlar, önerilen TfrAdmCov modelinin oldukça sağlam olduğunu göstermiştir.

İkinci olarak, COVİD-19 virüsünün mutasyon tahmini için sağlam bir StackGridCov modeli önerdik. Önerilen StackGridCov modeli, tamamen topluluk öğrenme tablanlıdır. Önerilen StackGridCov modelinin ve diğer modellerin performansını artırmak için GridSearchCV hiperparametre ayarlama algoritması kullanılmıştır. Önerilen StackGridCov modelinin ve diğer modellerin performansını değerlendirmek için, holdout tekniğinin yanı sıra stratified 10 katlı çapraz doğrulama tekniğinden faydalanılmıştır. Ek olarak önerilen StackGridCov modelinin performansını değerlendirmek için daha önce ortaya çıkan influenza A/H1N1 HA virüsü veri seti üzerinde mutasyon tahmini gerçekleştirilmiştir. GridSearchCV yöntemine sahip önerilen StackGridCov modeli, COVİD-19 test veri setinde 0.6623 doğruluk değeri, 0.6723 F1-skor değeri, 0.3273 MCC değeri ve 0.7018 AUC değeri ile diğer algoritmalardan daha iyi performans gösterimiştir. Ayrıca, önerilen StackGridCov modeli, influenza A/H1N1 HA test veri setinde 0.9460 doğruluk değeri, 0.7969 hassasiyet değeri, 0.8093 F1-skor değeri ve 0.7780 MCC değeri açısından diğer modellerden daha iyi performans göstermiştir. Sonuç olarak, GridSearchCV hiperparametre tekniğinin kullanılmasının genel olarak önerilen StackGridCov modeli ile diğer modellerim performansını arttırdığı gözlemlenmiştir.

Üçüncü olarak, COVİD-19 virüs mutasyon tahmini için HyperMixer ve dikkat mekanizmalarına dayalı olan HyperAttCov modeli önerilmiştir. Önerilen HyperAttCov modelinin performansının en yüksek seviyeye çıkartmak için dikkat mekanızmalarından faydalanılmıştır. Önerilen HyperAttCov modeli, birçok derin öğrenme tabanlı ve makine öğrenmesi modellerinden daha iyi performans elde etmiştir. Deneysel sonuçlar detaylı olarak incelendiğinde, önerilen HyperAttCov modelinin, COVİD-19 test veri seti üzerinde %70.0 doğruluk değerine, %92.0 kesinlik değerine ve %46.5 MCC değerine ulaştığını gözlemlenmiştir. Benzer şekilde, önerilen HyperAttCov modeli, 10 adet rastgele denemenin ortalaması alındığında COVİD-19 test veri seti üzerinde %70.2 doğruluk değerine, %90.4 hassasiyet değerine ve %46.2 MCC değerine ulaşmıştır. Ayrıca, önerilen HyperAttCov modeli literatürdeki çalışmayla karşılaştırıldığında, test veri seti kümesi üzerinde oldukça başarılı sonuçlar elde etmiştir. Sonuç olarak, önerilen TfrAdmCov, StackGridCov ve HyperAttCov modelleri, COVİD-19 veri setinde meydana gelecek mutasyonları başarılı bir şekilde tahmin edebilmektedir. Elde edilen sonuçlar aşı ve ilaç geliştirilmesi açısından umut vericidır.

USE OF ARTIFICIAL INTELLIGENCE-BASED ALGORITHMS IN DETECTING COVID-19 MUTATIONS

SUMMARY

Coronavirus disease 2019 (COVID-19) virus is a deadly type of coronavirus that has emerged recently and is highly contagious. The rapid spread of the COVID-19 virus has caused great fear and panic among people. Countries have had to take some measures such as complete closure and curfew to combat the COVID-19 virus. However, despite these measures, the COVID-19 virus continued to spread. Another method to combat the COVID-19 virus is the development of vaccines and drugs. The development of vaccines and drugs is of great importance in combating the COVID-19 virus. The effectiveness of these developed vaccines and drugs has either significantly decreased or disappeared completely as a result of the mutation of the COVID-19 virus. Therefore, it is very important to combat COVID-19 mutations. If future mutations in the structure of the COVID-19 virus can be predicted, vaccines and drugs can be developed more easily. Therefore, infected areas can be quarantined and ultimately the fight against the COVID-19 virus will be easier. Artificial intelligence-based approaches also offer promising results in detecting or predicting the COVID-19 virus. When the literature has been examined, most of the studies on the COVID-19 virus are studies on other aspects of the COVID-19 virus. For this reason, there is a serious gap in the literature in terms of mutation prediction of the COVID-19 virus. In this thesis study, we aim to fill this gap to some extent. In thesis study, three artificial intelligence-based models (TfrAdmCov, StackGridCov and HyperAttCov) have been proposed to predict future mutations in the COVID-19 Spike (S) protein structure.

Firstly, the proposed TfrAdmCov model is completely transformer encoder based with Adam optimization algorithm. With the proposed TfrAdmCov model, dependencies between the variables in the input sequence can be easily captured. The proposed TfrAdmCov model can perform parallel calculations simultaneously because it is transformer encoder-based architecture. In addition, in order to increase the performance of the proposed TfrAdmCov model, agglomerative clustering algorithm has been preferred during creation of the training, testing and Kfold datasets. Additionally, the GridSearchCV algorithm has been used to set the best hyperparameter values of machine learning algorithms. The experimental results in detail shows that the proposed TfrAdmCov model achieves better performance than both classical artificial intelligence -based models and several state-of-the-art models. The proposed TfrAdmCov model achieved 99.93% accuracy value, 100.00% precision value, 97.38% recall value, 98.67% F1-score value and 98.65% MCC value on the COVID-19 testing dataset. In the COVID-19 testing dataset, the TfrAdmCov model with the Adam optimization algorithm correctly predicted 335 samples out of 344 samples in the "mutation" class, while it incorrectly predicted only 9 samples out of 344 samples in the "mutation" class. In addition, the proposed TfrAdmCov model with Adam optimization algorithm correctly predicted all samples out of 12386 samples in the “no mutation” class. Similarly, when the average of 10 random experiments have

been taken, the proposed TfrAdmCov model achieved better results than other models in terms of accuracy with 99.924%, recall with 97.18%, F1-score with 98.57% and MCC value with 98.54% on the COVID-19 testing dataset. In addition, in order to statistically compare the proposed TfrAdmCov model with the deep learning models, the results obtained have been analyzed by taking the average of 10 random trials with different random seeds. Detailed evaluation has been carried out for each model in terms of accuracy, precision, recall, F1-score and MCC performance measurement metric using statistical measurements such as mean, standard deviation, median, mininum and maximum. The proposed TfrAdmCov model obtained an average of 0.999238, standard deviation of 0.000036, median of 0.999214, minimum of 0.999214 and maximum of 0.999293 among the 10 accuracy values obtained on the COVID-19 testing dataset. We also performed mutation prediction on the influenza A/H3N2 HA dataset to evaluate the performance of the proposed TfrAdmCov model. The proposed TfrAdmCov model achieved better results than other models 96.33% accuracy, 81.55% precision, 52.33% recall, 63.75% F1-score and 63.61% MCC values on the H3N2 HA testing dataset. On the H3N2 HA testing dataset, the proposed TfrAdmCov model correctly predicted 853 samples out of 1630 samples in the "mutation" class, while it incorrectly predicted 777 samples out of 1630 samples in the "mutation" class. In addition, the proposed TfrAdmCov model correctly predicted 24577 out of 24770 samples in the "no mutation" class, while it incorrectly predicted 193 out of 24770 samples in the "no mutation" class. Results on the influenza H3N2 HA testing dataset showed that the proposed TfrAdmCov model is quite robust.

Secondly, we propose a robust StackGridCov model for mutation prediction of the COVID-19 virus. The proposed StackGridCov model is based on ensemble learning. The proposed StackGridCov model is a very successful model that maximizes the performance as much as possible by using many machine learning algorithms. The main reason for this can be expressed as the proposed StackGridCov model reduces the possibility of overfitting by combining the strengths of several base models. These base models may make errors in different parts of the input sequences. By combining the outputs of these base classifiers, the meta-classifier can compensate for these errors and ultimately make a more accurate prediction. The proposed StackGridCov model is flexible as different machine learning algorithms can be used in both the level-0 layer and the level-1 layer. The proposed StackGridCov model is more robust than other ensemble learning and other artificial intelligence techniques as it is less affected by overfitting. This is because the base learners are trained on the same training dataset and the meta learner is trained on the new large dataset by combining the predictions of these base classifiers on the training dataset, ultimately reducing the possibility of overfitting. In this thesis study, while the base learners at level-0 have been selected as SVM, RF, XGBoost, ANN, DT, GB, ET, AdaBoost learner has been chosen as the meta classifier at level-1. This selection of both base classifiers and meta classifier significantly improved the performance of the proposed StackGridCov model. In addition, we use the GridSearchCV hyperparameter tuning algorithm to improve the performance of the proposed StackGridCov model and other models. To evaluate the performance of the proposed StackGridCov model and other models, the stratified 10-fold cross-validation technique as well as the holdout technique has been used. Additionally, to evaluate the performance of the proposed StackGridCov model, mutation prediction has been performed on the previously emerging influenza A/H1N1 HA virus dataset. The proposed StackGridCov model with GridSearchCV method outperformed other algorithms in terms of accuracy value of 0.6623, F1-score value of 0.6723, MCC value of 0.3273 and AUC value of 0.7018 on the COVID-19

testing dataset. Moreover, the proposed StackGridCov algorithm with GridSearchCV technique outperformed the StackGridCov model without GridSearchCV technique on the COVID-19 testing dataset. The proposed StackGridCov model with GridSearchCV method increased the accuracy value (from 0.6016 to 0.6623), precision value (from 0.5833 to 0.6415), recall value (from 0.6566 to 0.7062), F1-score value (from 0.6178 to 0.6723).), the MCC value (from 0.2063 to 0.3273) and the AUC value (from 0.6133 to 0.7018). The proposed StackGridCov model with the GridSearchCV method correctly predicted 399 samples out of 565 samples in the "mutation" class on the COVID-19 testing dataset, while it incorrectly predicted only 166 samples out of 565 samples in the "mutation" class. In addition, the proposed StackGridCov model with the GridSearchCV method correctly predicted 223 samples out of 587 samples in the “no mutation” class on the COVID-19 testing dataset, while it incorrectly predicted 364 samples out of 587 samples in the “no mutation” class. Similarly, the proposed StackGridCov outperformed other models in terms of accuracy value of 0.6610, a precision value of 0.6614, an F1-score value of 0.6607 and an MCC value of 0.3226 on the KFold dataset. Moreover, the proposed StackGridCov model outperformed other models in terms of accuracy value of 0.9460, recall value of 0.7969, F1-score value of 0.8093 and MCC value of 0.7780 on the Influenza A/H1N1 HA testing dataset. As a result, it has been observed that using the GridSearchCV hyperparameter technique has been generally increased the performance of the proposed StackGridCov model and other models.

Thirdly, the HyperAttCov model, which is based on LSTM, HyperMixer and attention mechanisms, is proposed for COVID-19 virus mutation prediction. Attention mechanisms have been used to maximize the performance of the proposed HyperAttCov model. The proposed HyperAttCov model is able to capture the most relevant input features and long-term temporal dependencies in the input sequence. Additionally, in this thesis study, attention mechanisms (input attention mechanism and temporal attention mechanism) have been used to improve the performance of the proposed HyperAttCov model by focusing on important parts of the COVID-19 dataset. While the input attention mechanism is applied to the entire input dataset, the temporal attention mechanism is applied to the data obtained from the HyperMixer architecture. The proposed HyperAttCov model achieved better performance than many deep learning-based and machine learning models. When the experimental results have been examined in detail, it has been observed that the proposed HyperAttCov model reached 70.0% accuracy value, 92.0% precision value and 46.5% MCC value in the COVID-19 testing dataset. Similarly, the proposed HyperAttCov model achieved 70.2% accuracy value, 90.4% precision value and 46.2% MCC value on the COVID-19 testing dataset when averaged over 10 random trials. In addition, the proposed HyperAttCov model achieved very successful results on the COVID-19 testing dataset compared to the study in the literature. As a result, the proposed TfrAdmCov, StackGridCov and HyperAttCov models can successfully predict mutations that will occur on both the COVID-19 S protein and the influenza datasets. In addition, in this thesis study, it has been observed that the use of agglomerative clustering algorithm and GridSearchCV hyperparameter technique played an effective role in mutation prediction of the COVID-19 virus. The results obtained this thesis study are promising for vaccines and drugs development.

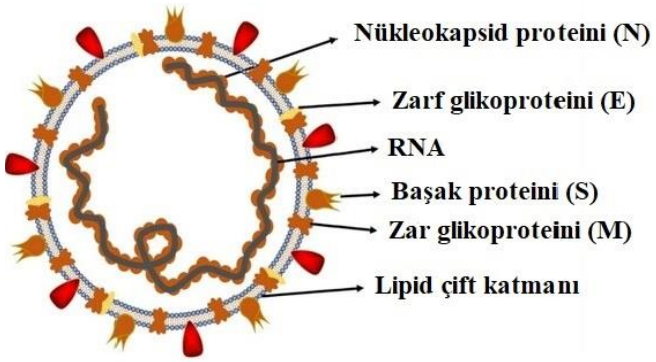
1. GİRİŞ

Koronavirüsler tek sarmallı pozitif polariteli Ribonükleik Asit (RNA) genom dizisi sahiptirler. Koronavirüsler ilk kez 1960'larda keşfedilmiştir (Haimed ve ark., 2021). İlk bulunan koronavirüsler HCoV-229E ve HCoV-OC43 koronavirüsleridir. Daha sonra 2003 yılında şiddetli akut solunum sendromu koronavirüs 1 (SARS-CoV-1) koronavirüsü, 2004 yılında HCoV-NL63 koronavirüsü, 2005 yılında HCoV-HKU1 koronavirüsü, 2012 yılında MERS (orta doğu solunum sendromu koronavirus) koronavirüsü ve son olarak da şiddetli akut solunum sendromu koronavirüs 2 koronavirüsün (SARS-CoV-2) neden olduğu COVİD-19, Aralık 2019 yılının sonunda tespit edildi (Haimed ve ark., 2021). Dünya Sağlık Örgütü (DSÖ) tarafından yeniden adlandırılan COVİD-19 virüsü Sohrabi ve ark. (2020), ilk olarak Aralık 2019 sonunda Çin'in Hubei eyaletinin başkenti Wuhan şehrinde ortaya çıktı (Wu ve ark., 2020). COVİD-19 virüsü başta Çin olmak üzere birçok hızla ülkeye yayıldı. Ülkeler, COVİD-19 virüsüyle mücadele için kısmen veya tamamen kapanmak zorunda kaldı. Bu durum halk arasında büyük korku ve paniğe neden oldu (Hai-Dong ve ark., 2022). DSÖ, 30 Ocak 2020'de COVİD-19 salgını için Dünya Acil Durumu ilan etmiş, ardından 11 Mart 2020'de ise pandemiye dönüştüğünü tüm dünyaya duyurmuştu (Sharma ve ark., 2021)(Tang ve ark., 2024). 9 Haziran 2024 itibarıyla dünya çapında teyit edilen vaka sayısı 775.615.736, teyit edilen ölüm sayısı ise 7.051.323'tür (DSÖ, 2023). COVİD-19 virüsü, enfekte kişilerin yaklaşık %80'inde hafif semptomlara neden olurken, bazı kişilerde akut solunum sıkıntısı sendromu (ARDS) neden olmuştur (Sharma ve ark., 2021). ARDS, çoklu organ yetmezliğine ve diğer ciddi hastalıklara neden olabilmektedir (Suri ve ark., 2020). COVİD-19 virüsünün tanısına yönelik birçok test kiti geliştirilmiştir (Zainol Rashid ve ark., 2020). En yaygın olarak kullanılan ve başarısı kanıtlanmış gerçek zamanlı ters transkriptaz polimeraz zincir reaksiyonu (rRT-PCR), COVİD-19 virüsünün tespitinde sıklıkla kullanılmaktadır (Serena Low ve ark., 2021). rRT-PCR test sonuçları genellikle birkaç saat ile 2 gün arasında elde edilir (Sharma ve ark., 2021). COVİD-19 virüsünün etkinliğini/yayılımını azaltmak amacıyla fiziksel veya sosyal mesafe, tam kapanma, kapalı alanların havalandırılması, öksürme ve hapşırma durumunda ağız ve burnun kapatılması, karantına gibi önlemler

alınmıştır. Ayrıca çeşitli aşılar geliştirilmiş ve COVİD-19 virüsünün etkinliği belli ölçüde azaltılmıştır. Ancak COVİD-19 virüsünün sık sık mutasyona uğraması, bu aşı ve ilaçların etkinliğini ya büyük ölçüde azaltmış ya da yok etme seviyesine getirtmiştir. Bu nedenle, COVİD-19 virüsüyle mücadele oldukça zor bir hal almıştır. Bu zorlukların üstesinden gelebilmek için COVİD-19 virüsü üzerinde meydana gelebilecek mutasyonların önceden tahmin edilmesi hayati önem taşımaktadır. Eğer COVİD-19 virüsünün yapısında özellikle S proteininde mutasyonlar tahmin edilebilirse, COVİD-19 virüsü mutasyona uğrasa bile aşı ve ilaçlar hızlı bir şekilde güncellenebilir. Özellikle son zamanlarda dizi bazlı mutasyon görevlerinde yapay zeka tabanlı modeller oldukça etkin bir şekilde kullanılmaktadır.

1.1. COVİD-19 (SARS-CoV-2)

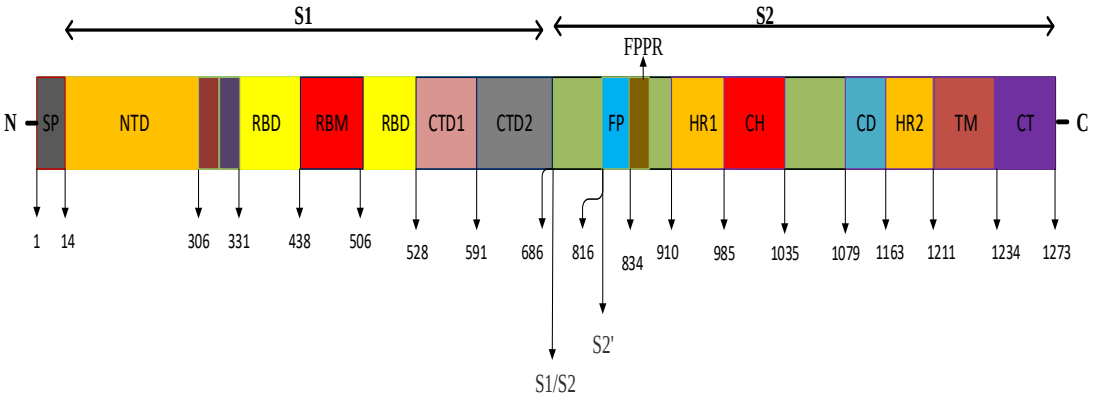
COVİD-19 virüsü, koronaviridae familyasına ait, zarflı, tek sarmallı RNA genomlarına sahip, pozitif anlamda betakoronavirüsün bir cinsidir (de Wit ve Cook, 2020). Koronavirüslerin alfa, beta, gama ve delta olmak üzere dört türü mevcuttur (Jaimes ve ark., 2020). İnsan koronavirüsleri alfa ve beta cinslerindedir (Shereen ve ark., 2020)(Cui ve ark.). COVİD-19 virüsünün genomu, tüm koronavirüsler arasında yarasa-RaTG13 koronavirüsü ile %96'nın üzerinde en yüksek genom benzerliğine sahiptir. Ayrıca SARS-CoV-1 ile %79'un üzerinde ve MERS koronavirüsü ile %50'nin üzerinde genomik benzerlik göstermektedir (Sharma ve ark., 2021). Çin'de ilk olarak ortaya çıkan COVİD-19 virüsünün genomu 29903 kilobaz uzunluk aralığına sahiptir (Nawaz ve ark., 2021). COVİD-19 virüsünün yapısı, yapısal proteinler (Başak (S), Zarf (E), Zar (M) ve Nükleokapsid (N)), yapısal olmayan proteinler (NSP1-NSP16) ve yardımcı proteinlerden (ORF3a, ORF3b, ORF6, ORF7a, ORF7b, ORF8, ORF9b, ORF9c and ORF10) oluşmaktadır (Wu ve ark., 2022). COVİD-19 virüsünün genel yapısı, Şekil **1.1**'de gösterilmiştir (Shereen ve ark., 2020).



Şekil 1.1. COVİD-19 virüsünün genel yapısı (Shereen ve ark., 2020)(Burukanlı ve Yumuşak, 2024a).

1.1.1. COVİD-19 başak (spike) S proteini

COVİD-19 virüsünün yüzeyindeki S proteini, bir transmembran glikoprotein olarak ifade edilmektedir (Zhang ve ark., 2021). S proteini, toplamda 1273 amino asitten oluşmaktadır (Zhang ve ark., 2021). Şekil **1.2**'de görüldüğü gibi, COVİD-19 S proteini, Sinyal Peptidi (SP), S1, S1/S2, S2 alt birimlerinden oluşmaktadır (Huang ve ark., 2020). S1 alt birimi, N-Terminal Alanı (NTD), Reseptör Bağlama Alanı (RBD) ve C-Terminal Alanı 1 (CTD1) ve C-Terminal Alanı 2 (CTD2) alanlarından oluşurken, S2 alt birimi ise Fusion Peptid (FP), Füzyon-Peptit Proksimal Bölgesi (FPPR), Heptad Tekrarı 1 (HR1), Merkezi Helis (CH), Konektör Alanı (CD), Heptad Tekrarı 2 (HR2), Transmembran Alanı (TM) ve Sitoplazmik Kuyruk (CT) alanlarından oluşmaktadır (Barnes ve ark., 2020). COVİD-19 virüsünün yapısı üzerindeki RBD alanı aracılığıyla, konakçı hücre yüzeyindeki Anjiyotensin Dönüştürücü Enzim 2 (ACE-2) proteinine bağlanır. Daha sonra S2 alt birimi kullanılarak konakçı hücre ile füzyon gerçekleşir ve ardından COVİD-19 virüsü konakçı hücreye girer. ACE2 reseptörüne bağlandıktan sonra S proteini bazı değişikliklere uğrar ve S proteininin S1/S2 bölgesinde furin proteazlar tarafından bölünür ve S1, S2 alt birimlerinin üretilmesini sağlar. COVİD-19 virüsünün konakçı hücreye girişini kolaylaştırmak amacıyla hücre yüzeyindeki transmembran serin proteaz-2 (TMPRSS2S2), S2 alt birimindeki S2' alanını bölerek S proteininin hazırlanmasında rol oynar. COVİD-19 S RBD, bir reseptör bağlama motifi (RBM) ve bir çekirdek yapı içerir (Jackson ve ark., 2022). Şekil **1.2**'de COVİD-19 S proteininin detaylı yapısı gösterilmiştir (Jackson ve ark., 2022).



Şekil 1.2. COVİD-19 S proteininin detaylı yapısı (Jackson ve ark., 2022)(Burukanlı ve Yumuşak, 2024a).

1.1.2. Mutasyon

Mutasyon, kısaca bir canlının genomundaki Deoksiribonükleik Asit (DNA) veya RNA dizisinde meydana gelen kalıcı değişiklikler olarak ifade edilebilir. RNA virüsleri DNA virüslerinden daha fazla mutasyona uğrar (Shaikh ve ark., 2021). Özellikle virüs, RNA genomunu konakçı hücreye kopyalarken sıklıkla mutasyona uğrar (Qin ve ark., 2021). COVİD-19 virüsü, birçok kez mutasyona uğramıştır (Hossain ve ark., 2021). Geliştirilen test kitleri, baskın COVİD-19 varyantlarını tam olarak yakalayamamaktadır. Mevcut aşıların mutasyonlara karşı etkinliği de önemli ölçüde azalmıştır. COVİD-19 virüsünün genom dizilimini, davranışını, kökenini ve ne kadar hızlı mutasyona uğradığını anlamak, aşı ve ilaçların geliştirilmesi açısından büyük önem taşımaktadır (Haimed ve ark., 2021). COVİD-19 virüsü, zaman içerisinde farklı bölgelerde mutasyona uğrayarak yeni varyantları ortaya çıkartmıştır. Bu yeni varyantların büyük çoğunluğu herhangi bir olumsuz etkiye yol açmasa da delta/omikron gibi bazı baskın varyantların bulaşıcılığı/ölümcül olması nedeniyle salgının seyrini değiştirmiştir (Shiehzadegan ve ark., 2021). COVİD-19 virüsünün genom dizisinin detaylı analizi ve mutasyon analizi, aşı veya ilaç geliştirilmesine katkı sağlayacaktır (Ahmed ve Jeon, 2022). Şu ana kadar, COVİD-19 virüsünün neden olduğu bazı baskın varyantlar şu şekilde ifade edilebilir; İngiltere'de tespit edilen B.1.1.7 (Alfa) varyantı, Güney Afrika'da tespit edilen B.1.351 (Beta) varyantı ve B.1.1.529 (Omikron) varyantı, Brezilya'da tespit edilen P.1 (Gama) varyantı ve. Hindistan'da tespit edilen B.1.617.2 (Delta) varyantı olarak ifade edilebilir (Qin ve ark., 2021)(Lopez-Rincon ve ark., 2021)(Gage ve ark., 2021)(Sokhansanj ve Rosen, 2022). Mevcut verilere göre, B.1.1.529 (Omikron) varyantı dünya çapında en yaygın

olan varyanttır (Madhi ve ark., 2022). Tespit edilen varyantlara göre, COVİD-19 test kitlerinde de güncellemeler yapılmaktadır. Bu sayede mutasyona uğramış virüslerle enfekte olan hastalarda güncellenen test kitlerinde olumsuz sonuçların ortaya çıkmasının önüne geçilebilecektir. COVİD-19 mutasyonlarının ortaya çıkmasıyla birlikte mevcut aşı ve ilaçların etkileri önemli ölçüde azalmıştır. COVİD-19 koronavirüsünün genom diziliminin analizi ve gelişmiş makine öğrenimine dayalı modellerin kullanılması, doktorların COVİD-19 virüsünün genetik yapısını anlamalarına yardımcı olabilecektir. Ayrıca COVİD-19 virüsünün genom dizilişinin anlaşılması aşı veya geliştirilecek ilaçların geliştirilmesine katkı sağlayacaktır (Ahmed ve Jeon, 2021).

1.2. Tezin Amacı

Tezin amacı aşağıdaki gibi birkaç madde ile sıralanabilir.

- Yapay zeka tabanlı algoritmalar kullanılarak COVİD-19 virüsünün S proteini üzerindeki mutasyonlarını tahmin etmek,
- COVİD-19 mutasyonlarının tahmini için yapay zeka tabanlı modeller önermek,
- COVİD-19 mutasyonları önceden tahmin edilerek, hızlı ve etkili bir şekilde geliştirilecek olan aşı ve ilaçların geliştirilmesine fikir vermek,
- COVİD-19 mutasyonları tahmini için yapay zeka tabanlı algoritmaların kullanılması ve sonuçlarının elde edilip benzer literatür çalışmalarıyla kıyaslanmak,
- COVİD-19 virüsü ile ilgili çok az çalışma mevcuttur. Bu nedenle literatüre bir nebze olsun katkı sağlamak olarak ifade edilebilir.

1.3. Tezin Organizasyonu

Tezin geri kalan bölümlerin okunabilirliğini kolaylaştırmak için tez organizasyonu şu şekilde ifade edilebilir: Bölüm 2'de COVİD-19 virüsü ile diğer vürüsler ile ilgili detaylı literatür taramasından bahsedilir. Bölüm 3'te bu tez çalışmasında kullanılan veri seti, klasik yapay zeka modelleri ile önerilen modellerden bahsedilir. Bölüm 4'te önerilen modeller ile diğer modellerin kullanılan COVİD-19 ve influenza veri setleri

üzerinde elde edilen bulgular detaylı olarak tartışılır. Bölüm 5'te ise sonuç ve önerilerden bahsedilir.

2. LİTERATÜR TARAMASI

Literatür ayrıntılı olarak incelendiğinde çalışmaların çoğunluğunun COVİD-19 mutasyon tahmini dışındaki çalışmalar olduğu görülmektedir. Ayrıca influenza virüsü üzerinde de sıklıkla mutasyon tahmini yapılmaktadır. Literatürde konuyla ilgili bazı çalışmalara aşağıda yer verilmiştir. Tarek ve ark. (2023), çalışmalarında COVİD-19 ölüm tahmini için evrişimsel sinir ağı (CNN), kapılı tekrarlayan birim (GRU) tabanlı hibrit modelini CNN-GRU modelini önerdiler. CNN-GRU modelini kullanarak Hindistan veri seti üzerinde COVİD-19 ölümlerini tahmin ettiler. Önerdikleri CNN-GRU modelini mevcut modellerle karşılaştırdıklarında önerdikleri CNN-GRU modelinin daha başarılı olduğunu gözlemlemişlerdir. ElAraby ve ark. (2022), çalışmalarında COVİD-19 Göğüs Röntgeni (CXR) görüntülerini sınıflandırmak için stokastik gradyan iniş optimizasyon tekniğine sahip Gri Ölçekli Uzaysal Kullanım Ağı (GSEN) modelini önerdiler. Önerdikleri GSEN modeli, diğer modellerden daha iyi sonuçlar elde etti. Elzeki ve ark. (2021), çalışmalarında göğüs röntgeni (CXR) görüntülerinden COVİD-19 virüsünü tespit etmek için Göğüs Röntgeni COVID Ağı adı verilen CXR görüntüleri (CXRVN) adlı modeli önerdiler. Önerdikleri CXRVN modelini, üç veri seti üzerinde test ettiler. Test sonucunda CXRVN modelinin, COVİD-19 virüsünü tespit etmede oldukça başarılı olduğu gözlemlediler. Elzeki ve ark. (2021), çalışmalarında dengesiz COVİD-19 veri seti için CXR görüntüleri elde etmek amacıyla derin öğrenme tabanlı modeli (Alt örneklenmemiş Konturlet Dönüşümü (NSCT) + CNN_VGG19) kullanarak, yeni bir algısal iki katmanlı görüntü füzyonu önerdiler. Önerdikleri modeli, diğer modellerle detaylı olarak karşılaştırdılar. Önerdikleri model, diğer modellere göre daha iyi performans elde etti. Chakraborty ve ark. (2022), çalışmalarında diyabetik hastalar için bulanık çıkarım sistemi ve makine öğrenimi modelleri yoluyla bir COVİD-19 risk tahmin yaklaşımı önerdiler. Önerdikleri modelin performansını değerlendirmek için stratified K-katlı çapraz doğrulama tekniğini kullandılar. Deneysel sonuçlar, önerdikleri modelin COVİD-19 risk tahmininde diğer mevcut modellere göre daha başarılı olduğunu gösterdi. Hassan ve ark. (2024), çalışmalarında bilgisayar tomografisi (CT) tarama görüntülerinden COVİD-19 virüsünün sınıflandırılması için bir derin evrişimli sinir ağı (DCNN)

modeli önerdiler. Sonuç olarak, önerdikleri model COVİD-19 sınıflandırma görevinde birçok son teknoloji modelden daha iyi performans gösterdi. Shrestha ve ark. (2022), çalışmalarında beyin tümörünü tespit etmek için derin öğrenme Tabanlı Evrişim Sinir Ağı (DCNN) modelini önerdiler. Önerdikleri DCNN modeli, beyin tümörünün tespitinde dikkate değer sonuçlar elde etti. Hassan ve ark. (2022), çalışmalarında akıllı şehirler için derin öğrenme tabanlı bir otomatik COVİD-19 tespit modeli önerdiler. Önerdikleri model, özellikle kalabalık yerlerde COVİD-19 virüsünün otomatik testitinde oldukça iyi sonuçlar elde etti. Cai ve ark. (2024), çalışmalarında gelecek sezonun baskın influenza A virüs suşunun hemaglutinin (HA) protein dizisini tahmin etmek için kodlayıcı-kod çözücü tabanlı FluPMT modeli önerdiler. Dizi kalıntıları arasındaki bağımlılıkları araştırmak için dikkat mekanizmalarını kullandılar ve influenza A virüslerinin evrimini modellemek için zaman serilerini kullandılar. Sonuç olarak, FluPMT modelinin hem H1N1 veri seti hem de H3N2 veri seti üzerindeki performansının diğer modellere göre daha iyi olduğunu göstermişlerdir. Li ve ark. (2023), çalışmalarında uzun kodlamayan RNA'ların (lncRNA'lar) hücre altı lokalizasyonunu tahmin etmek için GraphLncLoc adında bir grafik derin öğrenme ağ tabanlı modeli önerdiler. GraphLncLoc modeli, gizli özellikleri öğrenmek için grafik evrişimli ağları kullanır ve ardından elde edilen yüksek seviyeli özellikler, nihai tahmini gerçekleştirmek için tamamen bağlı bir katmanla beslenir. Çalışmada sonunda GraphLncLoc modelinin diğer modellere göre daha iyi performans elde ettiğini gösterdiler. Yin ve ark. (2022), çalışmalarında influenza antijenik varyantlarını tahmin etmek için IAV-CNN olarak adlandırılan, 2 boyutlu evrişimli sinir ağı (CNN) tabanlı bir model önerdiler. IAV-CNN modeli ile diğer modelleri üç influenza veri kümesi (H1N1, H3N2 ve H5N1) üzerinde eğitip ve test ettiler. Sonuç olarak, IAV-CNN modelinin, üç grip veri kümesinde en gelişmiş modellerden daha iyi performans gösterdi. Abbas ve ark. (2022), çalışmalarında antijenik influenza HA dizi çiftleri üzerinde çeşitli derin öğrenme modellerini uyguladılar. Önerdikleri derin öğrenme modelleri, influenza A virüsü üzerinde dikkate değer sonuçlar elde etti. Salama ve ark. (2016), çalışmalarında RNA'yı oluşturan proteinlerin amino asit dizilerini, sinir ağlarını (NN) ve kaba set tekniklerini kullanarak RNA virüsü mutasyonlarını tahmin ettiler. Çalışmalarında Çin ve Güney Kore'den elde edilen newcastle RNA virüsü dizilerinden oluşan bir veri seti kullandılar. Sonuçları incelendiklerinde, kaba küme tekniğinin sinir ağlarına göre daha iyi sonuçlar verdiğini gözlemlediler. Kaba set tekniğinin doğruluk oranının %75'in üzerinde olduğunu göstermişlerdir. Mohamed ve

ark. (2021), çalışmalarında seq2seq LSTM derin öğrenmeyi kullanarak bir sonraki DNA dizisini tahmin ettiler. Çalışmalarında New Castle hastalık virüsü veri seti ile H1N1 influenza virüsü veri setini kullandılar. Önerdikleri modelinin başarı oranı (doğruluğu) New Castle hastalık virüsü veri seti %96.9, H1N1 influenza virüsü veri setindeki başarı oranı (doğruluk) ise %98.9'dur. Yin ve ark. (2020), yaptıkları çalışmada influenza A/H1N1, H3N2, H5N1 virüslerinin hemaglutinin (HA) protein dizilerini kullanarak gelecek grip sezonunda mutasyonların meydana gelip gelmeyeceğini tahmin etmişlerdir. İnfluenza A virüslerinin mutasyon tahmini için etkili ve sağlam bir zaman serisi mutasyon tahmin modeli olan Tempel modelini önerdiler. Çalışmalarında üç influenza veri seti (H1N1, H3N2, H5N1) üzerindeki deneysel sonuçlar incelendiğinde, önerilen Tempel modelinin literatürde yaygın olarak kullanılan diğer yaklaşımlardan daha iyi performans elde etti. Tempel modeli, 0.991 doğruluk değerine ulaştı. Yin ve ark. (2023), çalışmalarında virülans tahmini için genel bir çerçeve olan ViPal modelini önerdiler. ViPal modeli, diğer modellerden daha iyi performans gösterdi. Peng ve ark. (2023), çalışmalarında influenza A virüsü suşları arasındaki antijenik mesafeyi tahmin etmek için yeni bir niceliksel tahmin yöntemi önerdiler. Önerdikleri yöntem, influenza A virüsü suşları arasındaki antijenik mesafeleri tahmin etmede diğer yöntemlerden daha iyi performans gösterdi. Yin ve ark. (2023), çalışmalarında influenza A virüslerinin antijenitesini tahmin etmek için (CL-CAP) olarak adlandırılan, karşılaştırmalı öğrenmeye sahip bir evrişimli sinir ağı modeli önerdiler. Önerdikleri CL-CAP modelini birçok güncel yaklaşımla karşılaştırdılar. Önerdikleri CL-CAP modeli, diğer yaklaşımlardan daha iyi performans elde etti. Saha ve ark. (2020), çalışmalarında Hindistan'da izole edilen 566 COVİD-19 genom dizisini mutasyon analizi için kullanmışlardır. Dizilerin hizalanması, Ulusal Biyoteknoloji Bilgi Merkezi'nden (NCBI) alınan referans (NC_45512.2) dizisi kullanılarak çoklu dizi hizalama yöntemi (MSA) CLUSTALW Anonim (2023a) kullanılarak gerçekleştirdiler. Diziler hizalandıktan sonra mutasyon bölgesini bulmak ve her bir COVİD-19 genomunu analiz etmek için bir fikir birliği (consensus) dizisi oluşturdular. Çalışma sonucunda, Hindistan'da izole edilen 566 genom dizisinde 933 ikame/nokta mutasyon, 2449 silme mutasyonu ve 2 ekleme mutasyonu olmak üzere toplam 3384 mutasyon noktası tespit ettiler. Wang ve ark. (2020), çalışmalarında mutasyon analizi için 31421 COVİD-19 genom dizisini kullanmışlardır. Ayrıca, COVİD-19 genlerinin mutasyon oranını ve h-indeksini hesapladılar. Yazarlar, COVİD-19 virüsünün yapısını oluşturan genler arasında en

fazla N geninin mutasyona uğradığını ifade ettiler. Ayrıca N geninin, COVİD-19 genomundaki en savunmasız gen olduğunu da belirttiler. Haimed ve ark. (2021), çalışmalarında yapay zeka ve büyük veriyi kullanarak, COVİD-19 virüsünün kalıplarını ve evrimsel davranışını ortaya çıkarmak için bir tersine mühendislik yaklaşımını önerdiler. COVİD-19 virüsünün bir sonraki evrimleşen örneğini tahmin etmek için Uzun Kısa Süreli Bellek (LSTM) yöntemini kullandılar. Ayrıca, COVİD-19 virüsünün 29 amino asit uzunluğunda küçük bir proteini olan ORF7a amino asit dizisini kullandılar. Çalışmanın sonunda ORF7a proteininin olası evrimleşmiş örneğini %40-%50 başarı oranıyla tahmin ettiler. Bu başarı oranını arttırmak için tutarlı kalıplar kullanarak başarı oranını %72'ye çıkardılar. Nawaz ve ark. (2021), çalışmalarında yapay zeka tekniklerini kullanarak COVİD-19 genom dizilerinden detaylı bilgi elde ettiler. Nükleotid bazlarının sık görülen kalıplarını ve bunların birbirleriyle olan ilişkilerini ortaya çıkaran gizli kalıpların olup olmadığını görmek için bilgisayar ortamında sıralı desen madenciliği (SPM) ile çeşitli deneyler yaptılar. Ayrıca, genom dizilerinde nükleotid bazlarının değiştiği yerleri bulmak ve mutasyon oranını hesaplamak amacıyla genom dizilerinde mutasyon analizi için bir algoritma önerdiler. Hossain ve ark. (2021), yaptıkları çalışmada LSTM derin öğrenme modelini COVİD-19 genom dizisine uygulayarak gelecekte oluşabilecek 2000. varyantın mutasyon oranını tahmin etmişlerdir. Toplamda 259044 COVİD-19 tam genom dizisi kullandılar. Kullanılan bu örneklerden toplam 3334545 mutasyon tespit ettiler. Zhou ve ark. (2023a), çalışmalarında COVİD-19 mutasyon tahmini için transformer tabanlı mutasyon tahmin çerçevesi olarak adlandırılan TEMPO modelini önerdiler. Zamansal bilgilerle birleştirilmiş viral diziler oluşturmak için filogenetik ağaç bazlı bir örnekleme yöntemi tasarladılar. Ayrıca, önerdikleri TEMPO modeli, daha önce ortaya çıkmamış 22 mutasyonu da başarıyla tahmin etti. Önerdikleri TEMPO modeli, COVİD-19 veri kümesi üzerinde 0.655 doğruluk değerine ulaştı. Burukanlı ve ark. (2022), yaptıkları çalışmada COVİD-19 virüsün ilk ortaya çıkan varyantı (NC_045512.2) ile Türkiye'de ortaya çıkan MW306668.1 ile MT955161.1 varyantları için mutasyon analizi gerçekleştirdiler. Gerçekleştirdikleri mutasyon analizi sonucunda, MT955161.1 varyantının MW306668.1 varyantına göre daha fazla mutasyona uğradığının tespit ettiler.

Sonuç olarak, literatür detaylı olarak incelendiğinde çalışmaların çoğunluğunun COVİD-19 virüsünün diğer yönleri üzerine olduğu görülmektedir. Ancak yapay zeka

tabanlı modeller kullanılarak COVİD-19 virüsünün mutasyon tahmini konusunda çok az çalışma bulunmaktadır. Bu tez çalışmasında literatürdeki bu boşluğa odaklanılmıştır. COVİD-19 virüsü yapısında gelecekte meydana gelebilecek mutasyonları tahmin etmek için yapay zeka tabanlı TfrAdmCov modeli Burukanlı ve Yumuşak (2024a), StackGridCov modeli Burukanlı ve Yumuşak (2024b), ve HyperAttCov modeli Burukanlı ve Yumuşak (2024c), olmak üzere üç model önerilmiştir. Önerilen TfrAdmCov modeli, Adam optimizasyon tekniğene sahip transformer kodlayıcı tabanlı bir modeldir. Önerilen TfrAdmCov modeli, transformer encoder tabanlı olması nedeniyle, aynı anda paralel hesaplama gerçekleştirebilmektedir. Deneysel sonuçlar, önerilen TfrAdmCov modelinin COVİD-19 mutasyon tahmini için hem geleneksel yapay zeka tabanlı modellerden hem de literatürdeki birkaç son teknoloji modellerden daha iyi performans elde ettiğini görülmüştür. İkinci olarak, önerilen StackGridCov modeli, stacking topluluk öğrenme tablanlı bir modeldir. Bu tez çalışmasında önerilen StackGridCov modeli ile diğer modellerin performansını üst seviyelere çıkartmak için GridSearchCV hiperparametre ayarlama tekniğinden faydalanılmıştır. Üçüncü olarak, önerilen HyperAttCov modeli, LSTM kodlayıcı, HyperMixer ve dikkat mekanızmalarına dayalı bir sağlam modeldir. Önerilen HyperAttCov modeli, COVİD-19 mutasyon tahmini için birçok derin öğrenme tabanlı ve makine öğrenmesi modellerinden daha iyi performans elde etmiştir. Önerilen modeller hakında detaylı bilgiler, Bölüm 3 ve Bölüm 4'te verilmiştir.

.

3. MATERYAL VE YÖNTEM

3.1. Klasik Modeller

Bu bölümde COVİD-19 mutasyon tahmini için kullanılan klasik modellerden bahsedilmiştir.

3.1.1. Destek vektör makinesi

SVM modeli, sınıflandırma ve regresyon problemlerinin çözümünde sıklıkla kullanılan denetimli bir öğrenme yaklaşımıdır. SVM modelinin genelleme yeteneği oldukça yüksektir. SVM modelinin amacı, iki sınıf arasındaki ayrım marjını maksimuma çıkaracak doğrusal bir optimal hiperdüzlem bulmaktır. SVM modelinin en önemli avantajlarından biri yüksek oranda başarılı sonuçlar elde etmesi, dezavantajı ise çok geç sonuç vermesidir (Cortes ve Vapnik, 1995).

3.1.2. Random forest

RF modeli, karar ağaçlarına dayalı bir topluluk öğrenme yaklaşımı olarak ifade edilebilir. Genellikle sınıflandırma ve tahmin problemlerinde kullanılır (Breiman, 2001).

3.1.3. Yapay sinir ağları

YSA modeli, insan beyninin çalışma prensibinden ilham alınarak tasarlanmıştır. YSA modeli, sınıflandırma ve tahmin problemlerinde yaygın olarak kullanılan bir makine öğrenmesi yaklaşımıdır. YSA modeli, temel olarak üç katmandan oluşur: giriş katmanı, gizli katman ve çıkış katmanı. Giriş katmanı, giriş modelinin bilgilerini tutarken, çıkış katmanı ise sınıflandırma için gizli katmanın tuttuğu giriş bilgilerini kullanır. Sinir ağındaki hata oranını en aza indirmek için gizli katman(lar) atanarak ağırlıklar güncellenir (Taspinar ve ark., 2022)(Post ve ark., 2021)(Agatonovic-Kustrin ve Beresford, 2000)

3.1.4. Karar ağacı

DT modeli, sınıflandırma ve regresyon problemlerinde sıklıkla kullanılan bir öğrenme yaklaşımıdır. DT modeli, yapısı entropi ve bilgi kazancı kavramına dayanmaktadır. DT modelinin temel avantajları hızlı ve uygun maliyetli olması, kurallarının anlaşılır

olması ve yüksek boyutlu verilerle iyi çalışmasıdır. Ancak uygun ağaç yapısı oluşturmanın zorluğu ve eğitim süresinin uzun olması DT modelinin dezavantajları arasında yer almaktadır (Post ve ark., 2021)(Kotsiantis, 2013).

3.1.5. Gradient boosting

GB modeli, sınıflandırma ve regresyon problemleri için kullanılan popüler bir topluluk öğrenme yöntemidir. GB modeli, birkaç zayıf algoritmayı güçlü geçiş algoritmalarıyla birleştirerek daha iyi sonuçlar sağlar (Friedman, 2001)(Friedman, 2002).

3.1.6. Extra tree

ET modeli, DT'lere dayanan bir topluluk öğrenme yaklaşımıdır. Sınıflandırma problemlerinde DT'lerin tahmini çoğunluk oyu kullanılarak yapılırken, regresyon problemlerinde DT'lerin tahmininin ortalaması alınarak gerçekleştirilir. Başka bir deyişle, güçlü bir ET modelini oluşturmak için tüm zayıf DT modelleri birleştirilir (sınıflandırma problemlerinde oylama yapılarak veya regresyon problemlerinde ortalama alınarak) (Toche Tchio ve ark., 2024).

3.1.7. K-en yakın komşu

KNN modeli, sınıflandırma ve regresyon problemlerinin çözümünde sıklıkla kullanılan bir öğrenme yaklaşımıdır. KNN modeli, aynı zamanda önceki verilerle K yakın ilişkisine bakarak sınıflandırma yapan etkili bir makine öğrenmesi tabanlı öğrenme modeli olarak da ifade edilebilir (Guo ve ark., 2003). Burada K değerinin tek haneli rakamlardan seçilmesine dikkat edilir.

3.1.8. XGBoost

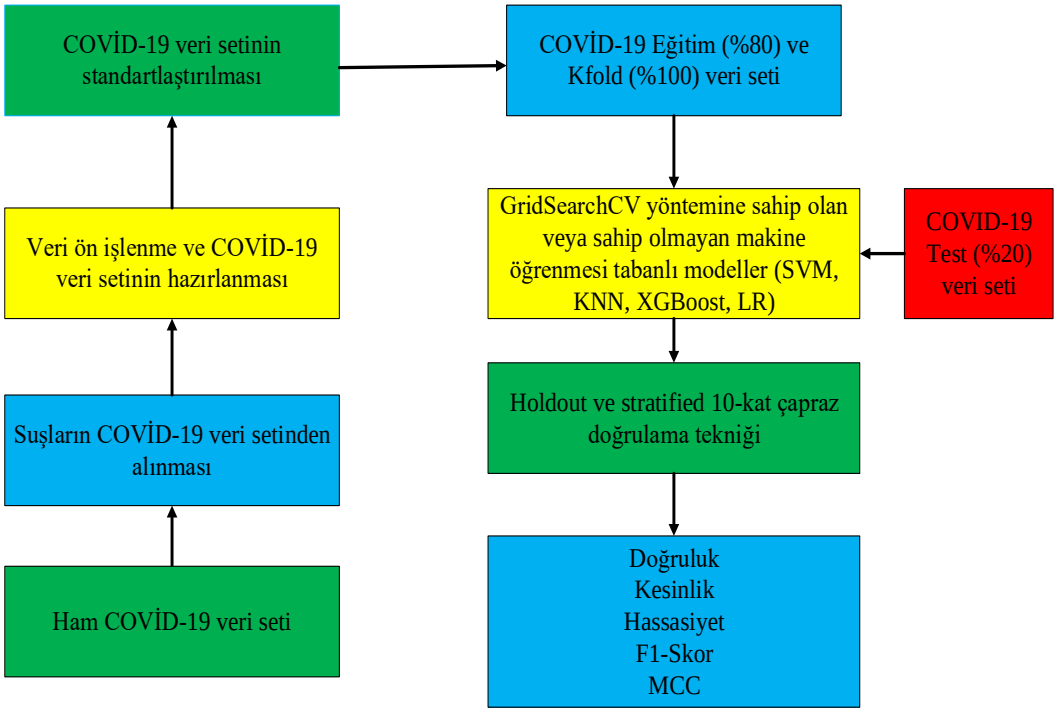
XGBoost modeli, Gradient Boosting modelinin optimize edilmiş ve performansı geliştirilmiş versiyonu olarak ifade edilen makine öğrenimi tabanlı bir modeldir. Sonuçların hızlı bir şekilde elde edilmesi, aşırı öğrenmenin (ezberlemenin) önlenmesi ve yüksek performans sağlanması bu modelin önemli avantajları arasında yer almaktadır (Memon ve ark., 2019).

3.1.9. Logistic regression

LR modeli, farklı alanlarda (istatistik, veri madenciliği vb.) yaygın olarak kullanılan standart olasılık istatistiksel sınıflandırma modeli olarak ifade edilmektedir. Bu modelin herhangi bir örnek üzerindeki çıktısı olasılık cinsindendir. Özellikle ikili sınıflandırmada kullanımı yaygındır (Feng ve ark., 2014).

3.1.10. COVİD-19 virüsünün mutasyon tahmini için makine öğrenimi modellerinin iş akışları

COVİD-19 virüsünün mutasyon tahmini için makine öğrenimi modellerinin iş akışları, Şekil **3.1**'de gösterilmiştir. Şekil **3.1**'de görüldüğü üzere, öncelikle Anonim (2023b) referans web adresinden ham COVİD-19 S protein suşları indirilmiştir. Daha sonra indirilen bu suşlar CLUSTAW Anonim (2023a) çoklu dizi hizalama (MSA) yöntemiyle hizalanarak veri seti elde edilmiştir. Veri setinden alınan her suş için, COVİD-19 S protein dizisini oluşturan 1273 alan (amino asitler) , 5 boyutlu küçük dizi bölünmüştür. Bu 5 küçük dizi, üst üste örtüşen 3 gramlık (3 overlapping 3 grams) dizilere bölünmüştür. Üst üste örtüşen 3 gramlık 3 küçük dizinin her biri, ProtVec'e dayalı 100 boyutlu gömülü dizilerle temsil edilmiştir. Daha sonra üst üste örtüşen 3 gramlık 3 küçük dizilerin toplamı alınarak 100 boyutlu tek bir vektörle temsil edilmiştir. Elde edilen 100 boyutlu tek vektör verisi StandardScaler Thara ve ark. (2019) yöntemi uygulanarak standardize edilmiştir. Daha sonra bu standartlaştırılmış verilere GridSearchCV'li veya GridSearchCV'siz makine öğrenimi tabanlı modeller uygulanmıştır. Daha sonra, holdout ve stratified 10 kat çapraz doğrulama teknikleri kullanılarak doğruluk, kesinlik, hassasiyet, F1-skor ve MCC performans ölçüm değerleri elde edilmiştir.



Şekil 3.1. COVİD-19 virüsünün mutasyon tahmini için makine öğrenimi modellerinin iş akışları (Burukanlı ve Yumuşak, 2024a).

3.1.11. RNN modeli

İleri beslemeli ağların (FFN) aksine, tekrarlayan sinir ağı (RNN) modeli, bir sonraki katmandan bir önceki katmana bilgi sağlayabilmektedir. RNN modeli, kısa vadeli bağımlılıkları modelleyebilse de, (vanishing/exploding) gradyan probleminden dolayı uzun vadeli bağımlılıkları modelleyemezler. RNN modelinin dizi tabanlı görevlerde kullanımı oldukça yaygındır (Zaremba ve ark., 2014).

3.1.12. LSTM modeli

LSTM modeli, kısa vadeli bağımlılıkları yakalayabilmesinin yanı sıra uzun vadeli bağımlılıkları da yakalayan RNN'lerin bir çeşidi olarak tanımlanabilir. Ancak RNN'den farkı kendi hafızasına sahip olmasıdır. Yani LSTM modeli, RNN'lerden daha güçlüdür. Ek olarak, LSTM modelinin kullanıma sunulmasıyla birlikte RNN'lerdeki (vanishing/exploding) gradyan sorunu da ortadan kaldırılmıştır (Hochreiter ve Schmidhuber, 1997).

3.1.13. GRU modeli

GRU modeli, LSTM modelinin basitleştirilmiş bir versiyonudur. Özellikle uzun vadeli bağımlılıkları verimli bir şekilde öğrenmek için geliştirildi. Ayrıca LSTM'den daha az kapısı bulunmaktadır (Chung ve ark., 2014).

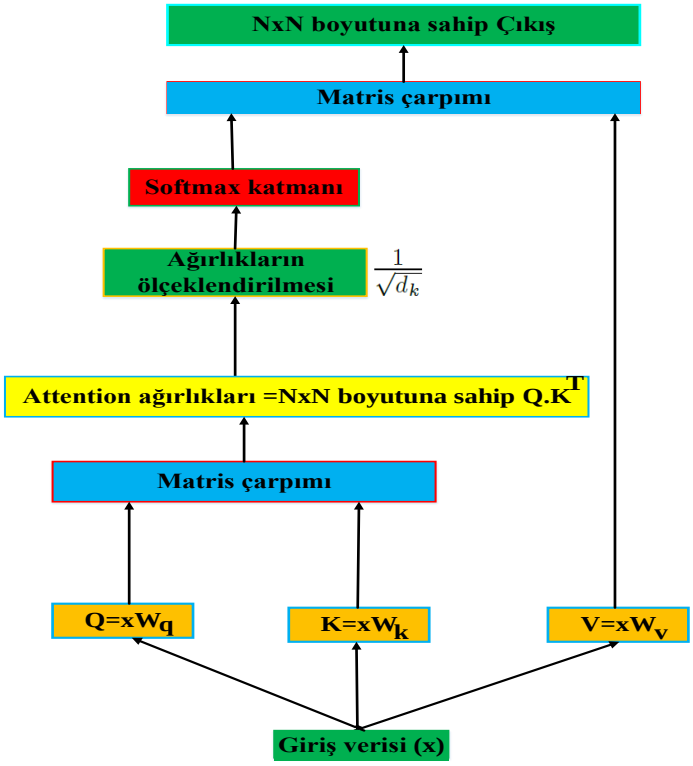
3.2. Önerilen TfrAdmCov Modeli

Önerilen TfrAdmCov modeli, tamamen transformer kodlayıcı tabanlı bir modeldir. Transformer kodlayıcı model, tamamen dikkat mekanizması tabanlı bir mimaridir ve doğal dil işleme (NLP) görevlerinde sıklıkla kullanılmaktadır (Kalyan ve ark., 2021). Transformer kodlayıcı katmanındaki dikkat mekanizması, eğitim sırasında modelin performansını en üst düzeye çıkarmak için özellik kümesindeki yalnızca en önemli özelliklere odaklanır. Bu sayede gereksiz hesaplama kaynakları azaltılır ve modelin daha iyi genelleme performansı elde etmesine olanak tanır. Transformer kodlayıcı modeli, giriş dizisindeki dikkat mekanizması sayesinde uzun vadeli bağımlılıkları kolaylıkla yakalayabilir ve büyük ölçekli paralel hesaplama gerçekleştirebilir (Zhou ve ark., 2023a). Standart transformer mimarisi, transformer kodlayıcı-kod çözücü (encoder-decoder) katmanlarından oluşmaktadır. Bu tez çalışmasında sadece transformer kodlayıcı katmanını kullanılmıştır. Her transformer kodlayıcı katmanının iki alt katmanı vardır: çoklu kafalı dikkat (MHA) ve ileri beslemeli ağ (FFN). Ayrıca, transformer kodlayıcı katmanı, iki alt katmanın her birinin etrafında bir artık

bağlantıya (residual connection) sahiptir ve ardından katman normalizasyonu (layer nozmalization) gerçekleştirilir (Vaswani ve ark., 2017)(Pacal, 2024a). Ölçekli noktalı ürün dikkati veya öz dikkat (scaled-dot product attention), Şekil **3.2**'de gösterilmiştir. Ölçekli nokta ürün dikkat mekanizması, eğitim sırasında model parametrelerini ayarlamak için W_q, W_k, W_v ağırlık matrislerini kullanır. Q, K ve V vektörleri, W ağırlık matrisleri ile gömülü x girişleri arasındaki matris çarpımı yoluyla elde edilir: i indeksi, d uzunluğuna sahip giriş dizisindeki jeton (token) konumunu belirtir. $Q = x_i W_q, K = x_i W_k, V = x_i W_v$. Bir dikkat işlevi, bir sorguyu $(Q = \{Q_1, \ldots, Q_N\})$ ve bir dizi anahtar/değer çiftini $(\{K, V\} = \{K_1, V_1, \ldots, K_M, V_M\})$ bir çıktıya eşler. Çıktı, değerlerin ağırlıklı toplamı olarak hesaplanır (Vaswani ve ark., 2017)(Galassi ve ark., 2021). Dikkat (Attention) fonksiyonu denklem (3.1)'de verilmiştir.

$$Attention\ (Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3.1}$$

Burada d_k, anahtar (key) boyutudur, d_v ise değer (value) boyutudur. Dikkat ağırlıklarını ölçeklendirmek için $\frac{1}{\sqrt{d_k}}$ kullanılır.



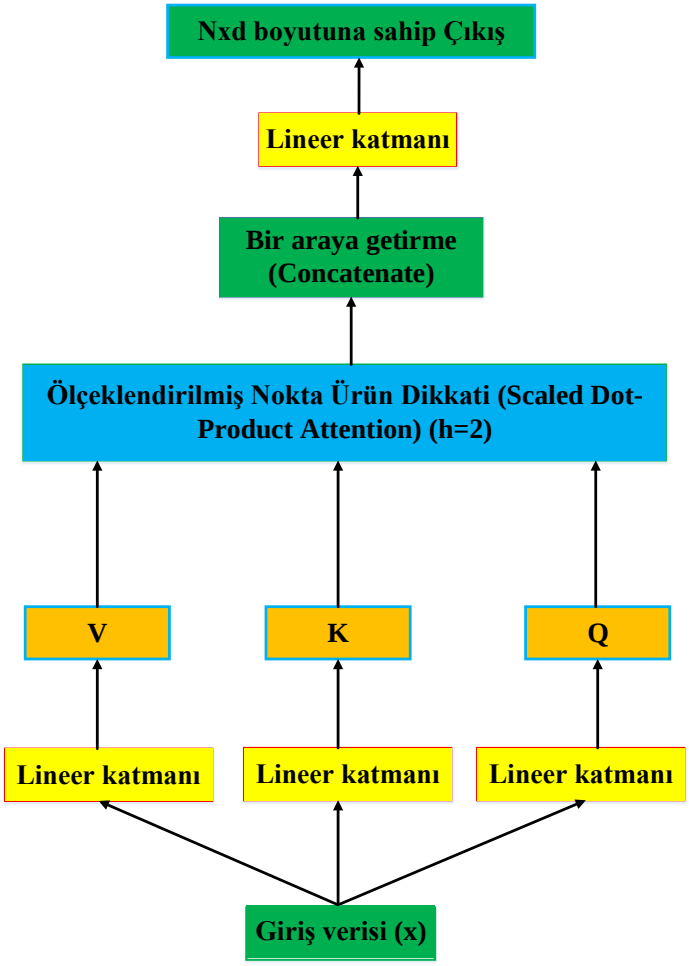
Şekil 3.2. Ölçekli noktalı ürün dikkati (scaled-dot product attention) (Burukanlı ve Yumuşak, 2024a).

Şekil **3.2**'de görüldüğü gibi Q, K, V matrisleri x giriş dizisi kullanılarak elde edilmektedir. Daha sonra $Q.K^T$'nin çarpılması sonucu dikkat ağırlıkları elde edilir. Elde edilen veriler dikkat ağırlıklarının $\frac{1}{\sqrt{d_k}}$. ile çarpılmasıyla ölçeklendirilir. Ölçeklendirilmiş veriler, softmax fonksiyonuna girdi olarak verilir. Softmax fonksiyonu ile veriler normalize edilir ve normalize edilen bu veriler V matrisi ile çarpılarak nihai çıktı elde edilir. Burada N, giriş dizisindeki jetonların (token) sayısıdır ve d, bu jetonların boyutudur. MHA mekanizması, modelin farklı konumlardaki farklı temsili alt uzaylardan gelen bilgilere ortaklaşa katılmasını sağlayan bir mekanizma olarak ifade edilebilir. Başka bir deyişle MHA, giriş dizisindeki her simge görevinin, aynı anda veya paralel olarak çalışan bir veya daha fazla öz dikkat kullanılarak farklı öz dikkat kafalarıyla paylaşılmasına olanak tanır. Bu, çıktının önceki girdiye bağlı olduğu RNN tabanlı modellerin aksine, birden fazla işlemin aynı anda gerçekleştirilmesine olanak tanır. MHA mekanizması, bir K anahtarı, bir V değeri ve bir Q sorgusu üzerinde çalışan bir veya daha fazla ölçekli nokta çarpım dikkatine (öz dikkat-self attention) dayanır (Voita ve ark., 2020). MHA mekanizması, denklem (3.2) ve (3.3)'teki formüller kullanılarak elde edilmektedir.

$$\mathrm{MHA(Q,K,V)} = \text{ Bir araya getirme}(Baş_1, \dots, Baş_h)W^O \tag{3.2}$$

$$Baş_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{3.3}$$

$W_i^Q \in \mathbb{R}^{dxd_q}, W_i^K \in \mathbb{R}^{dxd_k}, W_i^V \in \mathbb{R}^{dxd_v}$ ve $W^O \in \mathbb{R}^{hd_vxd}$ (Vaswani et al., 2017). Burada W_i^Q, W_i^K, W_i^V projeksiyon (projection) matrisleridir. W^O, son (final) doğrusal projeksiyon matrisidir (Voita ve ark., 2020). MHA mekanizmasının görseli, Şekil **3.3**'te gösterilmiştir. Bu tez çalışmasında, MHA mekanizmasında, $h = 2$ seçilmiştir. h, paralel çalışan ölçekli nokta ürün dikkat katmanlaının sayısını ifade eder. Ölçeklendirilmiş noktalı her ürün dikkat katmanı için $dk = dv = d/h = 50$. $d = 100$ olarak ayarlanmıştır.



Not: Nxd boyutuna sahip Q, K ve V matrisleri

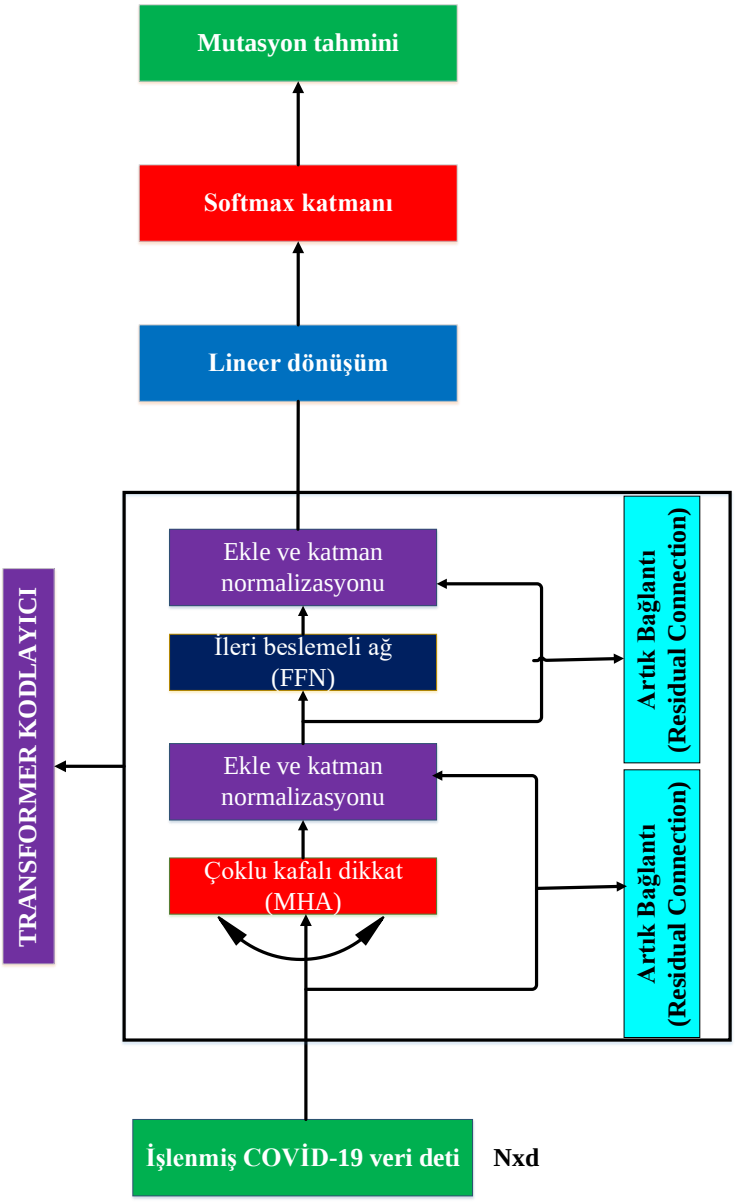
Şekil 3.3. MHA mekanizması (Burukanlı ve Yumuşak, 2024a).

Şekil **3.3**'te de görüldüğü gibi Q, K, V matrislerini elde etmek için x girdi dizisi doğrusal katmandan geçirilir. Daha sonra ölçeklendirilmiş nokta ürün dikkat katmanlarına girdi olarak Q, K, V matrisleri verilir. Ölçeklendirilmiş nokta ürün dikkat katmanlarından elde edilen çıktılar birleştirilir ve çıktılar doğrusal katmandan geçirilerek nihai çıktı elde edilir. Transformer kodlayıcı katmanı, iki doğrusal dönüşümden oluşan tamamen bağlı bir FFN katmanından ve bu iki doğrusal dönüşüm arasında bir düzeltilmiş doğrusal birim (RELU) aktivasyon fonksiyonundan oluşur. FNN katmanının formülü denklem (3.4)'te verilmiştir.

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \tag{3.4}$$

Burada x, (W_1, W_2), (b_1, b_2) sırasıyla giriş gömülü vektörü (input embedding vector), ağırlıkları (weights) ve biasları (biases) temsil eder (Vaswani ve ark., 2017). Ayrıca transformer kodlayıcıda artık veya atlama bağlantısı (residual or skip connection), giriş

dizisinin korunmasına yardımcı olarak transformer modelinin daha karmaşık fonksiyonları öğrenmesine olanak tanır. Ek olarak, artık bağlantı, transformer kodlayıcıdaki vanishing gradient sorununun önlenmesine yardımcı olur ve transformer modelinin performansını artırır. Bu tez çalışmasında makine çevirisi görevi yapmadığımız için sadece transformer kodlayıcı katmanını kullanılmıştır. Bu amaçla önerilen TfrAdmCov modeli yalnızca giriş dizisinin özelliklerini öğrenir ve öğrenilen bu özelliklere dayanarak COVİD-19 mutasyon tahminini gerçekleştirir. COVİD-19 virüsünün mutasyon tahmini için önerilen TfrAdmCov modelinin iş akışı Şekil **3.4**'te gösterilmiştir.



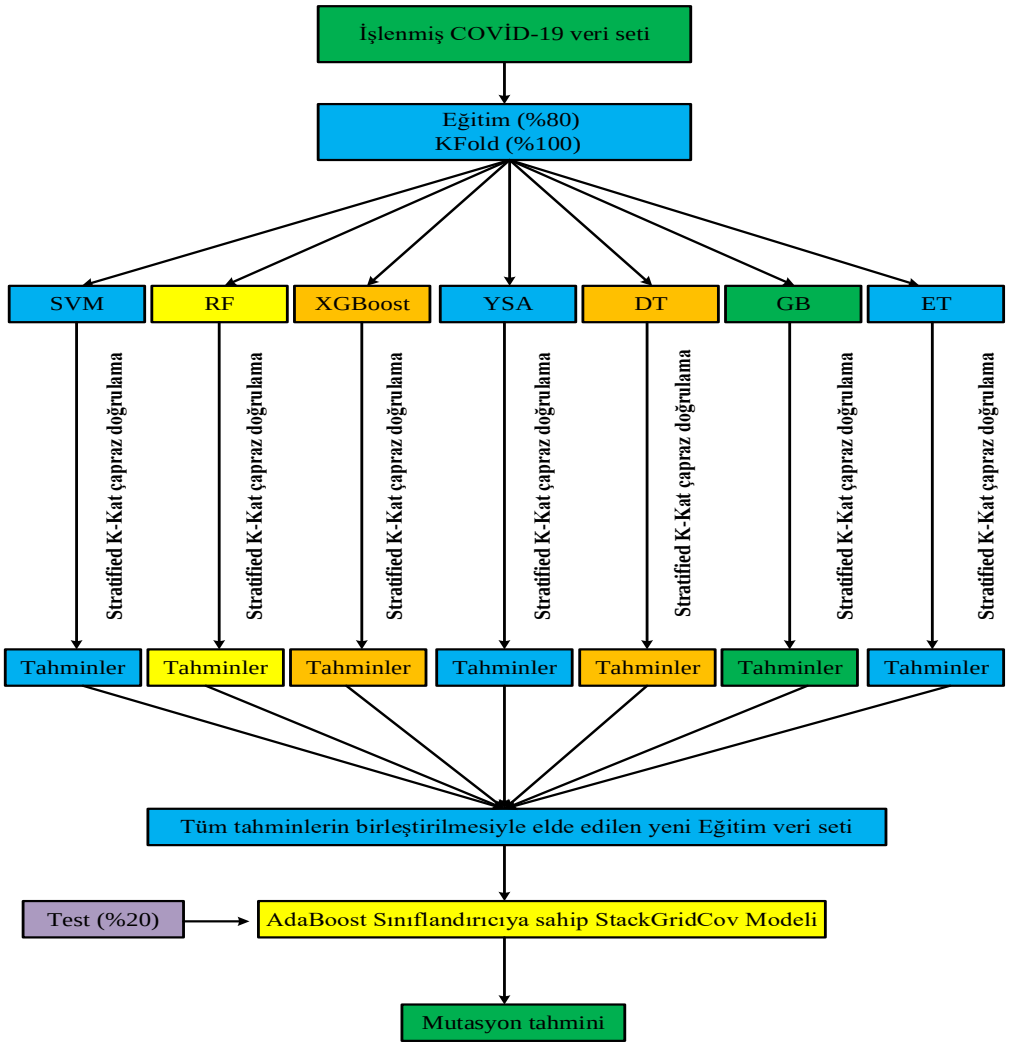
Şekil 3.4. COVİD-19 virüsünün mutasyon tahmini için önerilen TfrAdmCov modelinin iş akışları (Burukanlı ve Yumuşak, 2024a).

Şekil **3.4**'teki N, giriş dizisindeki jetonların sayısıdır ve d, bu jetonların boyutluluğudur. Giriş dizisinin boyutu Nxd'dir. Şekil **3.4**'te görüldüğü gibi işlenmiş COVİD-19 veri seti MHA'ya girdi olarak verilmektedir. MHA'dan elde edilen veriler katman normalizasyonuna (layer nomalization) girdi olarak verilir. Katman normalizasyonundan elde edilen veriler FFN katmanına girdi olarak verilmektedir. FFN katmanından elde edilen veriler katman normalizasyonuna girdi olarak verilmektedir. Katman normalizasyonundan elde edilen veriler doğrusal dönüşüm katmanına (linear transformation) girdi olarak verilmektedir. Doğrusal dönüşüm katmanından elde edilen veriler softmax katmanına girdi olarak verilmektedir. Daha sonra elde edilen yeni veri seti softmax katmanından geçirilerek son olarak COVİD-19 virüsünün mutasyon tahmini gerçekleştirilir.

3.3. Önerilen StackGridCov Modeli

Topluluk Öğrenme, sınıflandırma ve regresyon problemlerinde sıklıkla kullanılan bir öğrenme yaklaşımı olarak ifade edilebilir (Dong ve ark., 2020)(Dietterich, 2002)(Sewell, 2011)(Divina ve ark., 2018). Başka bir deyişle, topluluk öğrenme kısaca, birçok temel öğrenme algoritmasını birleştirerek daha sağlam ve oldukça iyi sonuçlar elde eden bir öğrenme yaklaşımı oluşturma yöntemi olarak tanımlanır (Dong ve ark., 2020)(Dietterich, 2002)(Sewell, 2011)(Divina ve ark., 2018). Genel olarak, topluluk öğrenme tek bir temel algoritmadan daha iyi sonuçlar verir. Genelleme yetenekleri çok güçlü olan topluluk öğrenme yaklaşımları en iyi sınıflandırma yöntemleri arasında yer almaktadır. En popüler topluluk öğrenme yaklaşımları AdaBoost, Bagging, Voting, Stacking (Stacked Generalization) öğrenme yöntemleridir (Dietterich, 2002)(Divina ve ark., 2018). Bu tez çalışmasında, önerilen StackGridCov modeli, seviye-0 (temel öğrenici seçilir) ve seviye-1 (meta öğrenici seçilir) üzerine kurulu bir Stacking topluluk öğrenimi tabanlı modeldir (Divina ve ark., 2018)(Post ve ark., 2021). Önerilen StackGridCov modeli, birçok makine öğrenmesi algoritmasını kullanarak performansı mümkün olduğu kadar en üst düzeye çıkaran oldukça başarılı modeldir. Bunun temel nedeni, önerilen StackGridCov modelinin birkaç temel modelin güçlü yönlerini birleştirerek aşırı uyum olasılığını azaltması olarak ifade edilebilir. Bu temel modeller girdi dizilerinin farklı kısımlarında hatalar yapabilir. Meta-model, bu temel sınıflandırıcıların çıktılarını birleştirerek bu hataları telafi edebilir ve sonuçta daha doğru bir tahmin yapabilir. Önerilen StackGridCov modeli, hem seviye-0 katmanında hem de seviye-1 katmanında farklı makine öğrenme

algoritmaları kullanılabildiğinden esnektir. Önerilen StackGridCov modeli, aşırı uyumdan daha az etkilenmesi nedeniyle diğer topluluk öğrenimi ve diğer yapay zeka tekniklerinden daha sağlamdır. Bunun nedeni, temel öğrenicilerin aynı eğitim veri seti üzerinde eğitilmesi ve meta modelin, bu temel sınıflandırıcıların eğitim veri seti üzerindeki tahminlerini birleştirerek yeni büyük veri seti üzerinde eğitilmesi ve sonuçta aşırı uyum olasılığının azaltılmasıdır. Bu tez çalışmasında seviye-0'da temel öğreniciler SVM, RF, XGBoost, YSA, DT, GB, ET olarak seçilirken, seviye-1'de meta öğrenici olarak AdaBoost sınıflandırıcı seçilmiştir (Taspinar ve ark., 2022)(Post ve ark., 2021)(Adaboost ve ark., 2009). Hem temel sınıflandırıcıların hem de meta sınıflandırıcının bu şekilde seçilmesi, önerilen StackGridCov modelinin performansını önemli ölçüde artırmıştır. COVİD-19 virüsünün mutasyon tahmini için önerilen StackGridCov modelinin iş akışı Şekil **3.5**'te gösterilmiştir.



Şekil 3.5. COVİD-19 virüsünün mutasyon tahmini için önerilen StackGridCov modelinin iş akışı (Burukanlı ve Yumuşak, 2024b).

Şekil **3.5**'te görüldüğü gibi, işlenmiş COVİD-19 veri seti, eğitim veri seti (%80) ve test veri seti (%20) olarak bölünmüştür. Eğitim veri seti, stratified K-katlı çapraz doğrulama için eğitim veri seti katlamaları (folds) ve eğitim veri seti doğrulaması olarak ikiye ayrılmıştır. Daha sonra, her makine öğrenimi algoritması, eğitim veri seti katlamaları üzerinde eğitilir ve eğitim veri seti doğrulaması üzerinde tahmin yapılır. Daha sonra tüm makine öğrenme algoritmalarının tahmin sonuçları birleştirilerek yeni tahmin veri seti elde edililir. AdaBoost sınıflandırıcısı sahip önerilen StackGridCov modeli, bu yeni tahmin veri seti üzerinde eğitilir. Daha sonra önerilen StackGridCov modeli test veri seti (%20) üzerinde test edilir ve nihayetinde COVİD-19 virüsünün mutasyon tahmini gerçekletirilir. Stratified K-katlı çapraz doğrulama tekniğine sahip önerilen StackGridCov algoritmasının sözde kodu (pseudo-code), Şekil **3.6**'da gösterilmiştir.

Giriş: İşlenmiş veri seti $= \{(x_1\ y_1), (x_2, y_2), \ldots, x_n, y_n)$. İşlenmiş veri seti, Eğitim veri seti ve Test veri seti olarak bölünmüştür. Eğitim veri seti, stratified K-katlı çapraz doğrulama için Eğitim Katlamaları $(folds)$ veri seti ve Eğitim doğrulaması $(validation)$ veri seti olarak ikiye ayrılmıştır.

Temel öğrenici modeller: BLM_1 , BLM_2 , …, BLM_S ; (Seviye-0)

Meta öğrenici modeller: MLM; (Seviye-1)

Çıkış: Test veri seti üzerinde mutasyon tahmini

for i = 1, …, S **do**

    $X_i = \emptyset$;

    for k = 1, …, K **do**

        $Eğitilmiş\_BLM_{ik} = BLM_i\,(Eğitim\ katlamaları\ veri\ seti)$# (Seviye-0)

        $X_{ik} = Eğitilmiş\_BLM_{ik}\,(Eğitim\ doğrulaması\ veri\ seti)$

    end for;

    $X_i = X_i \cup \{(X_{i1}, X_{i2}, \ldots, X_{iK})\}$

end for;

$Yeni\ tahminler\ veri\ seti = \emptyset$; # Meta sınıflandırıcı için yeni bir tahmin veri seti oluşturma

$Yeni\ tahminler\ veri\ seti = Yeni\ tahminler\ veri\ seti \cup \{(X_1, X_2 \ldots, X_S)\}$

$Eğitilmiş\_MLM = MLM(Yeni\ tahminler\ veri\ seti)$# (Seviye-1)

Return $Eğitilmiş\_MLM(Test\ veri\ seti))$

Şekil 3.6. Stratified K-katlı çapraz doğrulama tekniğine sahip önerilen StackGridCov algoritmasının sözde kodu (pseudo-code) (Burukanlı ve Yumuşak, 2024b).

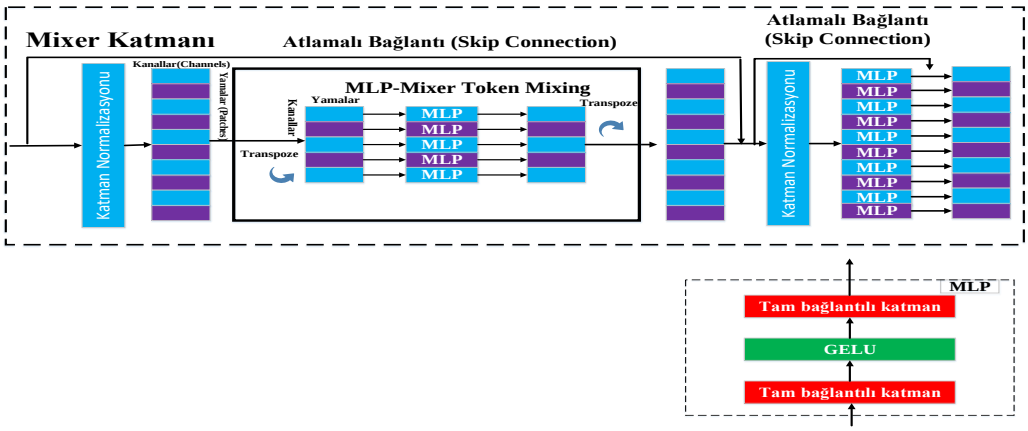
3.3.13. GridSearchCV hyperparametre ayarlama tekniği

GridSearchCV hiperparametre ayarlama tekniği Pirjatullah ve ark. (2021), genellikle yapay zeka tabanlı modellerin hiperparametre optimizasyonu için kullanılmaktadır. GridSearchCV'de herhangi bir yapay zeka tabanlı modelin hiperparametreleri, bu hiperparametre değerlerinin tüm kombinasyonları için ayrı ayrı oluşturulur. Buna göre en başarılı hiperparametre seti elde edilir. Bu hiperparametre tekniğinde tüm kombinasyonlar test edildiğinden en iyi performansı sağlayan hiperparametre seti elde edilir. Bu tez çalışmasında, her öğrenme algoritmasının en iyi parametre değerlerini seçmek için GridSearchCV hiperparametre ayarlama tekniği kullanılmıştır. Her öğrenme algoritmasının özellikleri arasından üç (3) rastgele özellik seçilmiştir. Her özellik için varsayılan değerler de dahil olmak üzere bazı parametre değerleri seçilmiştir. Daha sonra GridSearchCV hiperparametre ayarlama algoritması kullanılarak her özellik için en iyi parametre değerleri elde edilmiştir. Örneğin, üç rastgele hiperparametre (“final_estimator”, “stack_method” ve cv) ve bu hiper parametrelerin değerleri önerilen StackGridCov modeli için ayarlanmıştır. Önerilen StackGridCov modeli için hiperparametrelerin varsayılan değerleri final_estimator= LogisticRegression(), stack_method='auto, cv =None'dır. GridSearchCV algoritmasıyla ayarlanacak hiperparametreler ve bu hiperparametre değerleri şunlardır: final_estimator=[LogisticRegression(), AdaBoostClassifier()], stack_method= ['auto', 'predict_proba', 'decision_function', 'predict'], cv =[None , 10]. Önerilen StackGridCov modelinin hiperparametre ayarlama sürecinin daha iyi anlaşılması için, önerilen StackGridCov modeli öncelikle 1. kombinasyonla kurulur (final_estimator = LogisticRegression(), stack_method = 'auto', cv = None). Daha sonra önerilen StackGridCov modelin doğruluk değeri 5 kat çapraz doğrulama kullanılarak elde edilir. Benzer şekilde önerilen StackGridCov modeli bu defa 2. kombinasyonla kurulur (final_estimator = LogisticRegression(), stack_method = 'auto', cv = 10). Daha sonra önerilen StackGridCov modelinin doğruluk değeri 5 katlı çapraz doğrulama kullanılarak elde edilir. GridSearchCV algoritması kullanılarak hiperparametrelerin tüm kombinasyonları için doğruluk değerleri bu şekilde elde edilir. Sonuçta önerilen StackGridCov modeli için en iyi doğruluk değerini elde eden hiperparametre seti seçilir. Daha sonra önerilen StackGridCov modeli seçilen bu hiperparametre seti ile eğitilir. Önerilen StackGridCov modeli, nihayetinde test veri kümesi üzerinde mutasyon tahminini gerçekleştirir. Her bir öğrenme algoritması için

seçilen hiperparametreler ve bu hiperparametrelerin değerleri SVM modeli için Tablo B. 1'de RF modeli için Tablo B.2'de, XGBoost modeli için Tablo B.3'te, YSA modeli için Tablo B.4'te, DT modeli için Tablo B.5'te, GB modeli için Tablo B.6'da, ET modeli için Tablo B.7'de ve önerilen StackGridCov modeli için Tablo B.8'de gösterilmiştir.

3.4. Önerilen HyperAttCov Modeli

Mai ve ark. (2023), Tolstikhin ve ark. (2021) tarafından sunulan MLP-Mixer mimarisinden esinlenerek NLP görevleri için HyperMixer'ı tasarladılar. Tolstikhin ve ark. (2021), bilgisayarla görme (computer vision) görevleri için 2021 yılında yayınlanan çok katmanlı algılayıcı karıştırıcısını (MLP-Mixer) sundular. Onlar, MLP-Mixer mimarisini konvülasyon (convolution) ve dikkat (attention) yerine tamamen MLP'lere dayalı olarak tasarladılar. MLP-Mixer mimarisinin standart mixer katmanı, Şekil **3.7**'de gösterilmiştir. MLP-Mixer katmanında her token, özelliklerin bir vektörü olarak temsil edilir. MLP-Mixer mimarisinin, her katmanında iki adet MLP kullanır: biri token karıştırma (token mixing) MLP'si, diğeri ise özellik karıştırma (feature mixing) MLP'si. Token karıştırma MLP'si her özelliğe bağımsız olarak uygulanır ve uzamsal (spatial) konumlar arasındaki etkileşimleri modellerken, özellik karıştırma MLP'si ise her token vektörüne bağımsız olarak uygulanarak özellikler (feature) arasındaki etkileşimleri modeller. Pratikte bu durum, özellikleri temsil eden boyutun ve konumları temsil eden boyutun transpozesi alınarak aşılır (Mai ve ark., 2023).



Şekil 3.7. MLP-Mixer mimarisinin standart mikser katmanı (Tolstikhin ve ark., 2021)(Burukanlı ve Yumuşak, 2024c).

Her bir MLP, 2 adet tam bağlantılı katmandan (fully-connected layers) ve GELU altivasyon fonksiyonundan oluşmaktadır.

$i\,(token) \leq M$ özelliğini temsil eden her $x_i^T \in \mathbb{R}^d$ vektörü, denklem (3.5)'te görüldüğü gibi HyperMixer token mixing-MLP'ye giriş olarak verilir. Burada M, giriş dizisinin değişken (variable) boyutudur (Mai ve ark., 2023).

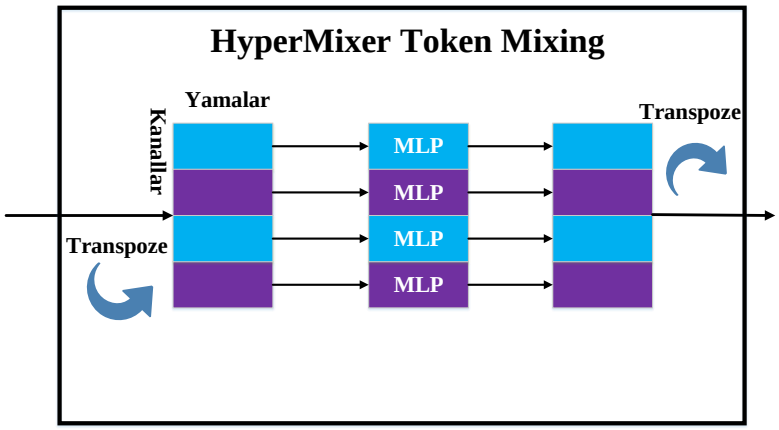
$$\text{HyperMixer token mixing} - MLP(x_i^T) = W_1(\sigma(W_2^T x_i^T)) \tag{3.5}$$

Burada $W_1, W_2 \in \mathbb{R}^{Mxd'}$ ağırlıkları ve σ ise gauss hatası doğrusal birimi (GELU) aktivasyon fonksiyonunu temsil eder (Hendrycks ve Gimpel, 2016). Öğrenmeyi kolaylaştırmak için her MLP'nin çevresine katman normalleştirmesi (layer normalization) Ba ve ark.(2016) ve atlama bağlantıları (skip connections) He ve ark. (2016) dahil edilir. Mai ve ark. (2023), HyperMixer'ın, dinamik olarak ağırlıklarını üretmek için hiper ağlarını (hypernetworks) kullandılar (Ha ve ark., 2016). W_1 ve W_2 ağırlıkları, $h_1, h_2 : \mathbb{R}^{Mxd} \rightarrow \mathbb{R}^{Mxd'}$ parametreli fonksiyonları tarafından üretilir. Burada h_1 ve h_2, jetonlar (tokens) arasındaki doğrusal olmayan etkileşimleri dikkate alan herhangi bir fonksiyon olabilir (Mai ve ark., 2023). Bir hiper ağ (hypernetwork) fonksiyonu, denklem (3.6)'daki tanımlanabilir.

$$h_i(x) = \begin{pmatrix} MLP^{W_i}(x_1 + p_1) \\ \vdots \\ . \\ MLP^{W_i}(x_M + p_M) \end{pmatrix} \in \mathbb{R}^{Mxd'}, \tag{3.6}$$

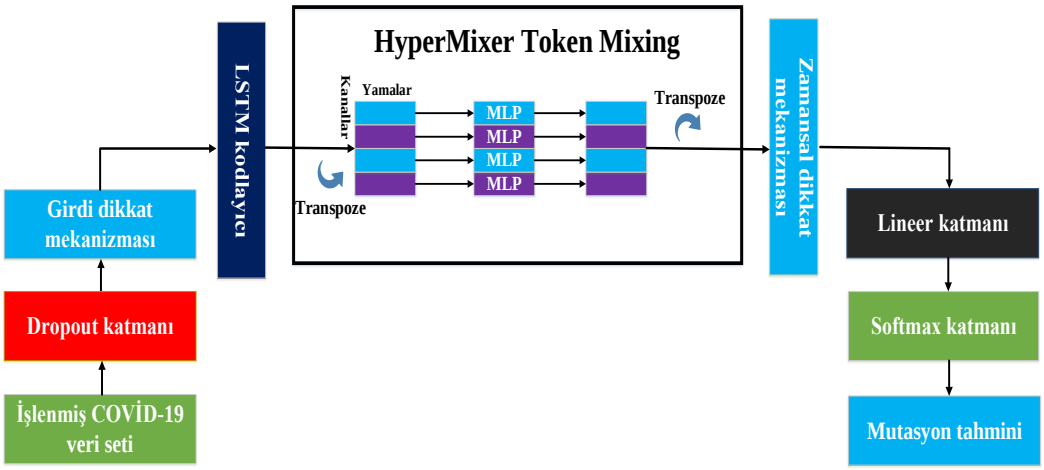
Burada MLP^{W_1}, $MLP^{W_2}: \mathbb{R}^d \rightarrow \mathbb{R}^{d'}$ tam bağlantılı katmana ve GELU aktivasyon fonksiyonuna sahip MLP'lerdir (Mai ve ark., 2023). Bu tez çalışmasında, tüm modeller için (SVM, LR ve RF hariç) token gömme boyutu $d = 100$ ve hidden size $d' = 128$ olarak alınmıştır. $p_i \in \mathbb{R}^d$, ek bilgiyi kodlayan bir vektördür (Ha ve ark., 2016)(Mai ve ark., 2023)(Vaswani ve ark., 2017). Mai ve ark. (2023), HyperMixer'ın alternatif MLP tabanlı modellerden daha iyi performans gösterdiğini göstermişlerdir. MLP tabanlı modeller ve transformerlardan farklı olarak HyperMixer, işlem süresi, eğitim veri seti ve hiper parametre ayarı açısından önemli ölçüde daha düşük maliyetlerle sonuçlara ulaşmayı başarmaktadır (Mai ve ark., 2023). Transformer'ın giriş dizisi uzunluğunda ikinci dereceden/karesel karmaşıklığı ($O(N^2)$) olmasına rağmen, HyperMixer'ın karmaşıklığı giriş dizisi uzunluğunda doğrusaldır ($O(N)$). Bu, HyperMixer'ı daha uzun giriş dizileri üzerinde eğitim için rekabetçi bir alternatif olarak sunmaktadır. Ayrıca token karıştırma kısmında standart MLP-Mixer mimarisi, girişin sabit (fixed) boyutuna ve konuma özgü (position-specific) ağırlıklara sahip bir

MLP kullanırken, HyperMixer mimarisi ise girişin değişken (variable) boyutuna ve konuma göre değişmeyen (position-invariant) bir MLP kullanır. Bu nedenle HyperMixer mimarisi, NLP görevlerinde daha uygundur çünkü konumdan bağımsız, değişken boyutlu bir eşleme kümesi oluşturmayı öğrenir (Mai ve ark., 2023). HyperMixer token mixing, Şekil **3.8**'de gösterilmiştir.



Şekil 3.8. HyperMixer token mixing (Mai ve ark., 2023)(Burukanlı ve Yumuşak, 2024c).

Bu tez çalışmasında NLP görevlerine daha uygun ve maliyeti düşük olan HyperMixer, LSTM ve atttention tabanlı HyperAttCov modelini öneriyoruz. Önerilen HyperAttCov modelinin geliştirilmesinde Yin ve ark. (2020)'dan ilham alınmıştır. Yin ve ark. (2020), derin öğrenme modellerini ve dikkat mekanizmalarını oldukça başarılı bir şekilde entegre etmişlerdir. Yin ve ark. (2020), tarafından sunulan TEMPEL ve DaRnn modelleri, giriş dizisindeki uzun vadeli bağımlılıkları yakalamak için dikkat mekanizmasına sahip RNN tabanlı bir mimarilerdir (Yin ve ark., 2020). Bu tez çalışmasında, COVİD-19 virüsünün mutasyon tahmini için önerilen HyperAttCov modelinin iş akışı Şekil **3.9**'da gösterilmiştir. Önerilen HyperAttCov modeli, giriş dizisindeki en ilgili giriş özelliklerini ve uzun vadeli zamansal bağımlılıkları yakalayabilmektedir. Ayrıca bu tez çalışmasında COVİD-19 veri setinin önemli kısımlarına odaklanarak önerilen HyperAttCov modelinin performansını artırmak için dikkat mekanizmalarından (girdi dikkat mekanizması ve zamansal dikkat mekanizması) yararlanılmıştır. Girdi dikkat mekanizması, tüm giriş veri setine uygulanırken, zamansal dikkat mekanizması ise HyperMixer mimarisinden elde edilen verilere uygulanır. Buradaki amaç, önerilen HyperAttCov modelinin performansını en üst düzeye çıkarmaktır.



Şekil 3.9. COVİD-19 virüsünün mutasyon tahmini için önerilen HyperAttCov modelinin iş akışı (Burukanlı ve Yumuşak, 2024c).

Şekil **3.9**'da görüldüğü gibi, işlenmiş COVİD-19 veri seti ilk olarak dropout katmanından geçirilir (burada dropout katmanı aşırı öğrenmeyi engellemek için kullanıldı). Dropout katmanından geçirilen veri seti, girdi dikkat mekanizmasına girdi olarak verilir (tüm giriş dizisine uygulanır). Girdi dikkat mekanizmasından elde edilen veri seti, LSTM kodlayıcı mimarisine giriş olarak verilir. LSTM kodlayıcı mimarisinden elde edilen veri seti, HyperMixer token karıştırma mimarisine girdi olarak verilir. HyperMixer token karıştırma katmanına girdi olarak verilen işlenmiş veri seti, zamansal dikkat mekanizmasına girdi olarak verilir (HyperMixer token karıştırma katmanından elde edilen veri seti üzerine uygulanır). Daha sonra elde edilen veriler, sırasıyla doğrusal katmanı ve softmax katmanından geçirilir ve nihayetinde COVİD-19 mutasyon tahmini gerçekleştirilir.

3.4.1. Softmax fonksiyonu

Softmax fonksiyonu, derin öğrenme görevlerinde sıklıkla kullanılan bir aktivasyon fonksiyonu türüdür. Gerçek değerleri, 0-1 arasındaki olasılık değerlerine eşler. Bu fonksiyon son zamanlarda dikkat mekanizmalarında da kullanılmaya başlanmıştır. Softmax formülü denklem (3.7)'de verilmiştir (Banerjee ve ark., 2020).

$$softmax(x_a) = \frac{\exp(x_a)}{\sum_{b=1}^{G} \exp(x_b)} \tag{3.7}$$

Burada x_a, x giriş dizisinin a.ncı değerini ifade eder. x_b, x verilerindeki diğer dizileri belirtir. G, x dizisinin boyutudur.

3.4.2. Kayıp (loss) fonksiyonu

İkili sınıflandırma görevlerinde, tahmin edilen her bir olasılığı gerçek sınıf çıktısıyla karşılaştıran ve beklenen değerden uzaklığa bağlı olarak olasılıkları güncelleyen, ikili çapraz entropi (cross-entropy) adı verilen bir kayıp fonksiyonunu kullanır. Bu tez çalışmasında ele alınan görev iki sınıflı bir problemdir (mutasyon var, mutasyon yok). Bu nedenle, gerçek y_t ile tahmin edilen $\dot{y}_t$ arasındaki kayıp değerini hesaplamak için çapraz entropi kullanıyoruz. Kayıp fonksiyonu LF denklem (3.8)'deki formül kullanılarak hesaplanır.

$$LF = -\frac{1}{N}\Sigma_{i=1}^{N}\frac{1}{D^{(i)}-1}\Sigma_{t=1}^{D^{(i)}-1}\Sigma_{z}^{F}\{y_{t_{(z)}}^{D}\log\left(\dot{y}_{t_{(z)}}\right)+(1-y_{t_{(z)}}^{D})\log\left(1-\dot{y}_{t_{(z)}}\right)\} \tag{3.8}$$

Burada N, girdi örneklerinin sayısıdır ve F ise seçilen kalıntı bölgelerinin (residue sites) kümesidir. $D^{(i)}$, COVİD-19 mutasyon tahmini için i.inci eğitim örneklerindeki seçilen pozisyonlarının sayısıdır (Yin ve ark., 2020).

3.5. Bu Tez Çalışmasında Kullanılan Veri Setleri

Bu bölümde COVİD-19 mutasyon tahmini için kullanılan veri setlerinden bahsedilmiştir.

3.5.1. Önerilen TfrAdmCov modeli için kullanılan veri seti hakkında detaylı bilgiler

3.5.1.1. COVİD-19 S protein veri seti

COVİD-19 S protein veri seti, S protein dizilerinden oluşur. COVİD-19 S protein veri seti, toplamda 1273 amino setinden meydana gelmektedir (Anonim, 2023b)(Zhang ve ark., 2021). Bu tez çalışmasında 2020-2022 yılları arasında Anonim (2023b) nolu referans adresinden her yıl için toplam 15000 adet COVİD-19 S protein dizisi indirilmiştir. Tüm S protein dizileri indirildikten sonra tüm diziler, CLUSTAW Anonim (2023a) çoklu dizi hizalama (MSA) yöntemi kullanılarak yıllara göre hizalanmıştır.

3.5.1.2. COVİD-19 S protein veri setinin hazırlanması ve ön işleme adımları

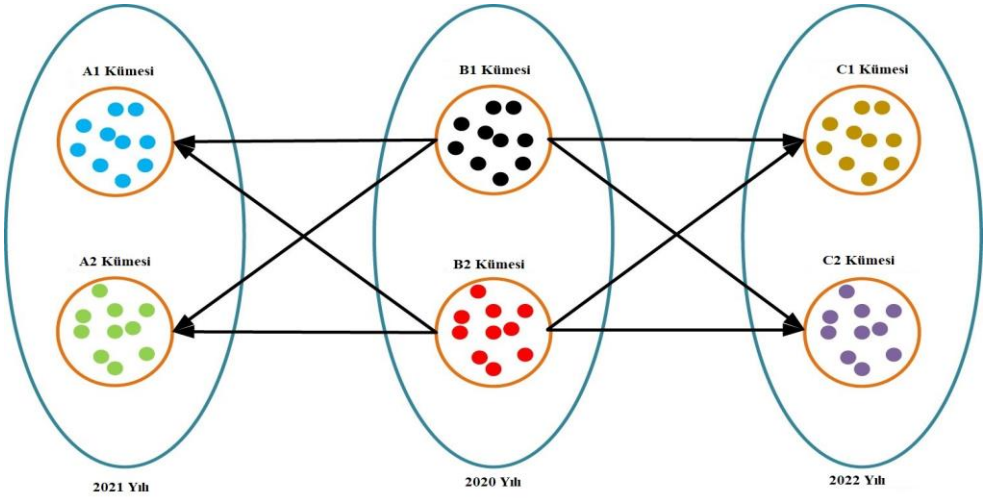
2020-2022 yılları arasında 20 evrensel genetik kod tarafından doğrudan kodlanan amino asitlerin dışında bazı suşlarda belirsiz birkaç amino asit bulunmaktadır. Bu suşlardaki belirsizliği ortadan kaldırmak amacıyla belirsiz 'B' harfi yerine rastgele 'D'

veya 'N' harflerinden biri atanmıştır. Belirsiz 'Z' harfi yerine rastgele 'E' veya 'Q' harflerinden biri atanmıştır. Son olarak, 20 evrensel amino asit arasında belirsiz 'X' harfi yerine rastgele bir amino asit ataması yapılmıştır. Bu şekilde tüm belirsizlikler giderilmiştir (Yin ve ark., 2020). Bu tez çalışmasında, Yin ve ark. (2020) tarafından sulunan veri seti oluşturma yöntemi kullanılmıştır. Bu yöntem KMeans kümeleme algoritması kullanılarak elde edildi. Bu tez çalışmasında ilk etapta KMeans kümeleme algoritmasını kullandık ancak makine öğrenmesi tabanlı algoritmaların performansını istenilen düzeyde yakalayamadık. Bu nedenle veri setleri oluşturma aşamasında KMeans kümeleme algoritması yerine başarı oranının arttırılmasında önemli bir faktör olan agglomerative kümeleme Sasirekha ve Baby (2013) algoritması tercih edilmiştir.

3.5.1.3. Agglomerative kümeleme tekniği

Agglomerative kümeleme tekniği, hiyerarşik kümeleme yönteminin bir çeşididir (Sasirekha ve Baby, 2013). Agglomerative kümeleme, parçadan bütüne veya aşağıdan yukarıya yaklaşımı olarakta ifade edilebilir. Tüm veri kümesindeki veri örnekleri, kümelere dönüştürülür. Daha sonra oluşturulan bu kümeler mesafeye bağlı olarak birbirine yakın olan kümelerle birleştirilerek yeni bir küme elde edilir (Sasirekha ve Baby, 2013). Eğitim veri setinin oluşturulması aşamasında, COVİD-19 suşları yıllara göre ayrıştırılmış ve her yıldaki suşların kümelere ayrılması için agglomerative kümeleme algoritması kullanılmıştır. Ayrıca agglomerative kümeleme algoritmasının parametresi ve bu parametrelerin değerleri Tablo A.1'de gösterilmiştir. Bu tez çalışmasında eğitim, test ve Kfold veri setleri kullanılmıştır. Yıllara göre eğitim, test ve Kfold veri setlerinin miktarları Tablo **3.1**'de gösterilmiştir. Tablo **3.1**'de görüldüğü gibi eğitim veri seti için 11250 COVİD-19 S protein suşu arasından her yıl için 30 suş rastgele seçilmiştir. Test veri seti için, 3750 COVİD-19 S protein suşu arasından her yıl için 10 suş rastgele seçilmiştir. Kfold veri seti için 15.000 COVİD-19 S protein suşu arasından her yıl için 40 suş rastgele seçilmiştir. Bu şekilde her veri seti için bu veri miktarlarını seçmemizin nedeni, GridSearchCV hiperparametre ayarlama yöntemini kullanmamızdır. Çünkü her makine öğrenmesi tabanlı modelin GridSearchCV yöntemi aracılığıyla en iyi parametre değerlerini seçerken sonuçlara ulaşmak çok zaman almaktadır. Bu tez çalışmasında kullanılan veri setleri için Şekil **3.10**'da da ifade edilen aglomeratif kümeleme algoritması kullanılarak her yıl için iki (2) küme oluşturulmuştur. Örneğin 2020 yılında B1 kümesinden seçilen bir suş, 2021 yılında bu suşa en düşük hamming mesafesine Norouzi ve ark. (2012) sahip olan A1

veya A2 kümesinden rastgele bir suş seçilir. Benzer şekilde 2020 yılında B1 kümesinden seçilen bir suş, 2022 yılında ise C1 veya C2 kümesinden bu suşa en düşük hamming mesafesine sahip rastgele bir suş seçilmiştir. Bu süreç tüm suşlar, veri setlerine dahil edilene kadar devam eder. Sonuçta farklı yıllara ait veriler tek tek bir araya getirilerek veri setleri elde edilir (Yin ve ark., 2020).



Şekil 3.10. COVİD-19 S proteini veri kümelerinin oluşturulmasına örnek (Yin et al., 2020)(Burukanlı ve Yumuşak, 2024a).

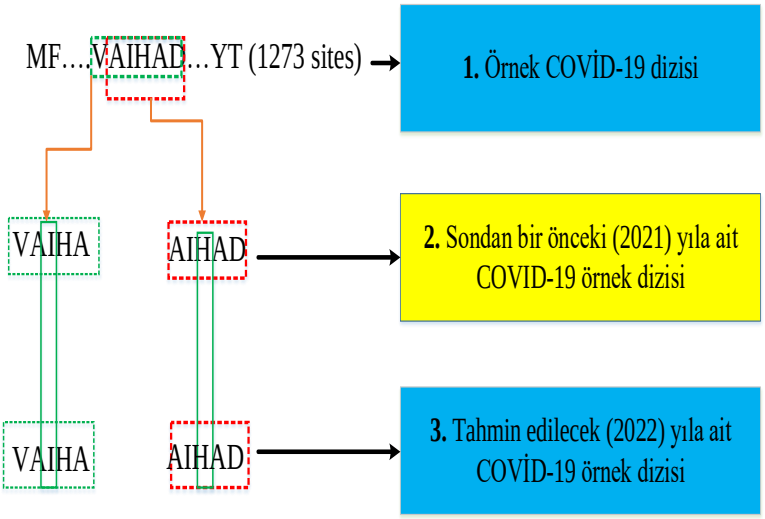
Tablo 3.1. Yıllara göre COVİD-19 S proteini veri kümelerinin suşlarının (strain) sayısı (Burukanlı ve Yumuşak, 2024a).

Yıl	Eğitim veri kümesi için suş sayısı	Test veri kümesi için suş sayısı	Kfold veri kümesi için suş sayısı
2020	30	10	40
2021	30	10	40
2022	30	10	40

Bu tez çalışmasında, 2020 ve 2021 yıllarında suşlar kullanılarak eğitim, test ve Kfold veri setleri için eğitim örnekleri oluşturulmuştur. Eğitim, test ve Kfold veri kümelerinin etiket/hedef veri örneklerini oluşturmak için 2021 (sondan bir önceki yıl) ve 2022 (tahmin edilecek yıl) yılları kullanılarak elde edilmiştir. Amacımız, 2020 ve 2021 suşlarını kullanarak COVİD-19 virüsünün 2022 yılındaki mutasyonlarını tahmin etmektir. Eğitim, test ve Kfold veri setleri için eğitim örnekleri oluşturma aşamaları Şekil **3.11**'de gösterilmiştir. Şekil **3.11**'de görüldüğü gibi, eğitim veri setleri oluşturulurken her bir amino asidi temsil edecek 5 bölge/kalıntı kullanılmıştır. Örneğin

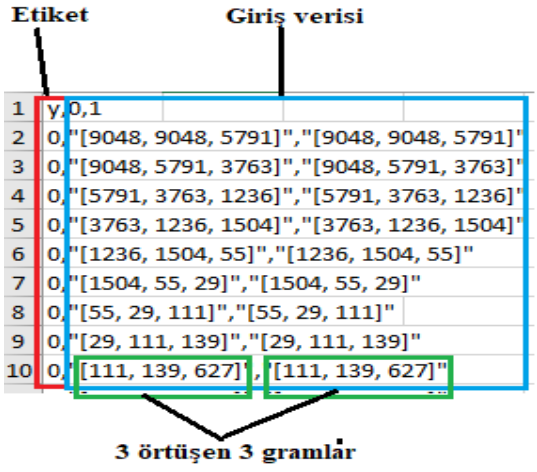
"VAIHA" dizisi "I" amino asidini temsil etmek için kullanılmıştır. “VAIHA” dizisi daha sonra üst üste örtüşen 3 gramlık (overlapping 3-gram) [VAI, AIH, IHA] küçük dizilere bölünmüştür. “AIHAD” dizisi “H” amino asidini temsil etmek için kullanılmıştır. “AIHAD” dizisi daha sonra üst üste örtüşen 3 gramlık [AIH, IHA, HAD] küçük dizilere bölünmüştür. Bu süreç, COVİD-19 S protein yapısının tamamında bulunan tüm bölgeler için bu şekilde sürdürülmüştür. Bu tez çalışmasında kullanılan tüm suşlar, örtüşen 3 gramlık 3 bölgeye bölündükten sonra, her 3 gram, Asgari ve ark. (Asgari & Mofrad, 2015) tarafından sunulan ProtVec'e dayalı 100 boyutlu bir gömme vektörü ile temsil edilir. Daha sonra 100 boyutlu 3 vektörün toplamı alınarak 100 boyutlu tek bir vektör elde edilir. Bu işlem, son suşa kadar devam edilir. Her yıla ait eğitim ve test veri kümelerinin etiket veri örneklerini oluşturmak için 2021 (sondan bir önceki yıl) ve 2022 (tahmin edilecek geçen yıl) kullanılmıştır. Eğitim ve test veri setleri için etiket örneklerinin oluşturulma aşaması Şekil **3.12**'de gösterilmiştir. Şekil **3.12**'de de görüldüğü üzere 2021'den (sondan bir önceki yıl) alınan I amino asidinin mutasyona uğrayıp uğramadığını kontrol etmek için merkez pozisyon (3. konum, “VAIHA” dizisinin I amino asidi) kontrol edilmiştir. 2022 yılında (son yıl) bu merkezi konumdaki I amino asidi değiştiyse mutasyon etiketi "1", değişmediyse mutasyon etiketi "0"dır. Benzer şekilde 2021'den (sondan bir önceki yıl) alınan H amino asidinin mutasyona uğrayıp uğramadığını kontrol etmek amacıyla “AIHAD” dizisinin merkez konumu (3. pozisyon, H amino asit) kontrol edilir. Bu merkezi konumdaki H amino asidi 2022 yılında (son yıl) değişmişse mutasyon etiketi "1", değişmediyse mutasyon etiketi "0"dır. Bu işlem, son suşa kadar devam ettirilir (Yin ve ark., 2020).

Şekil 3.11. Eğitim ve test veri kümeleri için eğitim ve test örneklerinin oluşturulması aşamaları (Burukanlı ve Yumuşak, 2024a).



Şekil 3.12. Veri kümelerinin eğitimi ve test edilmesi için etiket örneklerinin oluşturulma aşaması (Burukanlı ve Yumuşak, 2024a).

İşlenmiş COVİD-19 virüs veri seti ve detayları Şekil **3.13**'te gösterilmiştir. Şekil **3.13**'te de görüldüğü üzere işlenmiş COVİD-19 veri kümesi, etiket ve giriş verilerinden oluşmaktadır. Etiket değeri 1 ise "mutasyon var", 0 ise "mutasyon yok" anlamına gelmektedir. Her giriş verisi, 5 eğitim örneğinden oluşur (3 örtüşen - 3 gram). Her üç gram, Asgari ve ark. (2015) tarafından sunulan ProtVec'e dayalı 100 boyutlu bir gömme vektörüyle temsil edilir. Modelin eğitim aşamasında her üç gramlık 100 boyutlu vektör toplanarak 100 boyutlu tek bir vektör kullanılır.



Şekil 3.13. İşlenmiş COVİD-19 Veri Kümesi (Burukanlı ve Yumuşak, 2024a).

Bu tez çalışmasında derin öğrenme ve makine öğrenmesi tabanlı modellere girdi olarak verilen eğitim, test ve Kfold veri setlerindeki toplam veri miktarı sırasıyla Tablo **3.2**, Tablo **3.3** ve Tablo **3.4**'te olarak gösterilmiştir.

Tablo 3.2. Yıllara göre toplam eğitim veri seti miktarı (Burukanlı ve Yumuşak, 2024a).

Yıl	Suş sayısı X Kalıntı bölgelerinin sayısı	Toplam Eğitim veri kümesi miktarı
2020	30X1273	38190
2021	30X1273	38190
2022	30X1273	38190

Tablo 3.3. Yıllara göre toplam test veri seti miktarı (Burukanlı ve Yumuşak, 2024a).

Yıl	Suş sayısı X Kalıntı bölgelerinin sayısı	Toplam test veri kümesi miktarı
2020	10X1273	12730
2021	10X1273	12730
2022	10X1273	12730

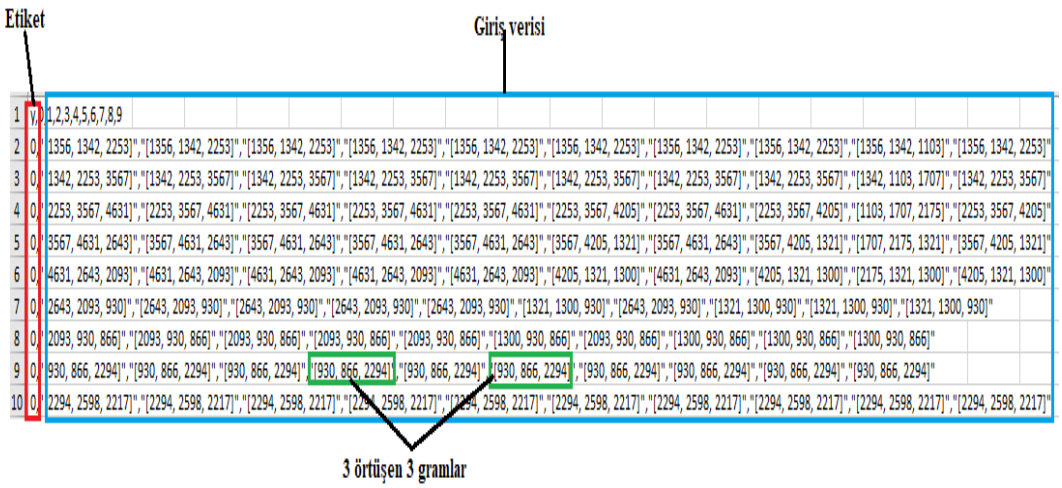
Tablo 3.4. Yıllara göre toplam Kfold veri seti miktarı (Burukanlı ve Yumuşak, 2024a).

Yıl	Suş sayısı X Kalıntı bölgelerinin sayısı	Toplam Kfold veri kümesi miktarı
2020	40X1273	50920
2021	40X1273	50920
2022	40X1273	50920

3.5.1.4. İnfluenza A/ H3N2 HA veri seti

Bu tez çalışmasında, önerilen TfrAdmCov modelinin performansını ölçmek için daha önce ortaya çıkmış influenza A/ H3N2 HA protein veri seti kullanılmıştır. Yin ve ark. (2020) tarafından sunulan infulenza A/H3N2 veri seti, 1991 ile 2016 yılları arasındaki HA protein dizilerinden oluşmaktadır. Bu veri seti toplam 132.000 dizi örneğinden oluşmaktadır (3'ü örtüşen 3 gram) (Yin ve ark., 2020). İşlenmiş influenza A/ H3N2

HA protein veri seti ve detayları Şekil **3.14**'te gösterilmiştir. Şekil **3.14**'te de görüldüğü üzere işlenmiş influenza A/ H3N2 HA protein veri seti, etiket ve giriş verilerinden oluşmaktadır. Etiket değeri 1 ise "mutasyon var", 0 ise "mutasyon yok" anlamına geliyor. Her giriş verisi, 5 eğitim örneğinden oluşur (3 örtüşen - 3 gram). Her üç gram, Asgari ve ark. (2015) tarafından sunulan ProtVec'e dayalı 100 boyutlu bir gömme vektörüyle temsil edilir. Modelin eğitim aşamasında her üç gramlık 100 boyutlu vektör toplanarak 100 boyutlu tek bir vektör kullanılır.



Şekil 3.14. İşlenmiş influenza A/ H3N2 HA protein veri seti (Burukanlı ve Yumuşak, 2024a).

Yin ve ark. (2020) tarafından sunulan influenza A/H3N2 HA protein veri kümesinin eğitim ve test veri kümeleri için sınıf miktarları Tablo **3.5**'te gösterilmiştir

Tablo 3.5. İnfluenza A/H3N2 HA protein veri kümesinin Eğitim ve Test veri kümeleri için sınıf miktarları ve toplam veri miktarı (Burukanlı ve Yumuşak, 2024a).

Veri seti	"Mutasyon var" sınıfı	"Mutasyon yok" sınıfı	Toplam veri
Eğitim	6325	99275	105600
Test	1630	24770	26400

3.5.1.5. Holdout yöntemi ile stratified 10 katlı çapraz doğrulama yöntemi

Bu tez çalışmasında, makine öğrenimi tabanlı modellerin performanslarını değerlendirmek için holdout ve stratified 10 katlı çapraz doğrulama teknikleri kullanılmıştır. Holdout tekniğinde Kohavi (1995), eğitim ve test veri kümeleri

kullanılır. Derin öğrenme ve makine öğrenimi tabanlı modeller, ilk olarak eğitim veri kümesinde eğitilir. Daha sonra daha önce hiç görmediği test veri seti üzerinde derin öğrenme ve makine öğrenmesi tabanlı modeller test edilir ve her algoritma için performans ölçümleri elde edilir. Stratified 10-katlı çapraz doğrulama tekniğinde Kohavi (1995) ise, K-katlı veri seti kullanılır. Veri seti, 10 parçaya bölünür. Bu durumda K=10 olarak ayarlanır. Derin öğrenme ve makine öğrenmesi tabanlı modeller için veri seti 9 parça eğitim 1 parça ise doğrulama/test için ayarlanır. Bu modeller, toplamda 10 defa test edilerek performans değerleri elde edilir. Elde edilen bu performans değerlerinin ortalaması alınarak nihai performans değeri elde edilir. Bu tez çalışmasında Tablo **3.6**'da da görüldüğü üzere, COVİD-19 S protein veri setlerinde sınıf dengesizliği bulunmaktadır. Model değerlendirmesinde holdout tekniği kullanıldığında tüm sınıflardan örnekler garanti edilmez. Bu büyük bir problem. Bu sorunun üstesinden gelmek için, COVİD-19 S protein veri setlerinin her bir sınıfının örnek yüzdelerini koruyarak verilerin kullanılmasına olanak tanıyan stratified 10 katlı çapraz doğrulama tekniği de tercih edilmiştir (Thölke ve ark., 2022)(Mbow ve ark., 2021). Tablo **3.6**'da holdout tekniğine ait eğitim ve test veri setinin yıllara göre toplam miktarları gösterilmiştir.

Tablo 3.6. Holdout tekniği için yıllara göre toplam eğitim ve test veri seti miktarı (Burukanlı ve Yumuşak, 2024a).

Yıl	Teknik	Veri seti	Toplam veri miktarı
2020	Holdout	Eğitim	38190
		Test	12730
2021	Holdout	Eğitim	38190
		Test	12730
2022	Holdout	Eğitim	38190
		Test	12730

Tablo **3.7**, stratified 10 kat çapraz doğrulama tekniği için yıllara göre COVİD-19 S proteini Kfold veri kümesinin miktarlarını göstermektedir.

Tablo 3.7. Stratified 10 katlı çapraz doğrulama tekniği için yıllara göre toplam K kat veri kümesi miktarı (Burukanlı ve Yumuşak, 2024a).

Yıl	Teknik	Veri seti	Toplam veri miktarı
2020	Stratified 10 katlı çapraz doğrulama	Kfold	50920
2021	Stratified 10 katlı çapraz doğrulama	Kfold	50920
2022	Stratified 10 katlı çapraz doğrulama	Kfold	50920

Tablo **3.8**, COVİD-19 virüsü için eğitim, test ve Kfold veri kümeleri için sınıf miktarlarını ve yaklaşık yüzdeleri gösterir.

Tablo 3.8. COVİD-19 virüsü için eğitim, test ve Kfold veri kümeleri için sınıf miktarları ve yaklaşık yüzdeleri (Burukanlı ve Yumuşak, 2024a).

Veri seti	"Mutasyon var" sınıfı	"Mutasyon yok" sınıfı	Toplam veri
Eğitim	1035 (2.71%)	37155 (97.29%)	38190 (100%)
Test	344 (2.70%)	12386 (97.30%)	12730 (100%)
Kfold	1374 (2.70%)	49546 (97.30%)	50920 (100%)

3.5.1.6. GridSearchCV hyperparametere ayarlama tekniği

Bu tez çalışmasında, herbir makine öğrenmesi modelinin en iyi hiperparametre değerlerinini seçmek için GridSearchCV algoritmasının varsayılan değerleri kullanılmıştır. Her makine öğrenmesi modelinin özellikleri arasında 3 rastgele özellik seçilmiştir. Tercih edilen her özellik için varsayılan değerler dahil toplam 5 parametre değeri seçilmiştir. Daha sonra GridSearchCV algoritması kullanılarak her bir özellik için en iyi parametre değerleri elde edilmiştir. Makine öğrenimi modellerine ilişkin hiperparametreler ve hiperparametrelerin değerleri Tablo **3.9**'da gösterilmiştir.

Tablo 3.9. Makine öğrenimi modelleri için hiper-parametreler (Burukanlı ve Yumuşak, 2024a).

Model	Hiper-parametreler ve değerleri
SVM	C = [1.0, 2.0,3.0,4.0,5.0] kernel =['linear', 'poly', 'rbf', 'sigmoid', 'precomputed'] probability =[True, False]
KNN	n_neighbors =[3,5,7,9,11] weights = ['uniform', 'distance'] algorithm = ['auto', 'ball_tree', 'kd_tree', 'brute']
XGBoost	booster=[‘gbtree’, ‘gblinear’,’dart’,None] learning_rate=[0.001,0.01,0.1,1,None] n_estimators=[50,100,150,200,250]
LR	C = np.linspace(1, 10, num=5) solver = ['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'] max_iter=[100, 1000, 10000, 100000, 1000000]

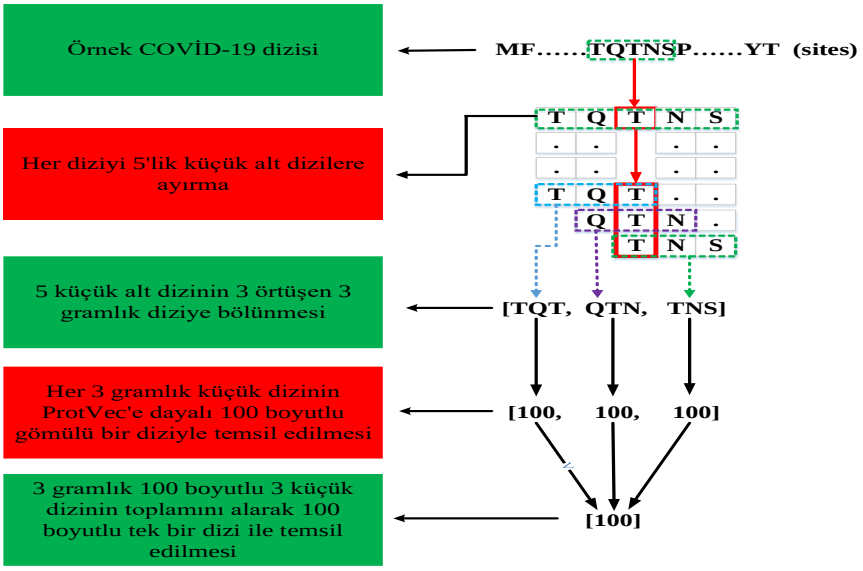
Tablo **3.9**'da da görüldüğü gibi her bir makine öğrenimi modeli için tüm hiperparametreler arasından rastgele üç hiperparametre seçilmiştir. Daha sonra bu hiperparametre değerleri arasından en iyi olan değerler GridSearchCV algoritması kullanılarak seçilmiştir. Her makine öğrenimi tabanlı model için seçilen hiperparametreler ve hiperparametrelerin değerleri, SVM modeli için Tablo A.2'de, KNN modeli için Tablo A. 3'te, XGBoost modeli için Tablo A. 4’te ve LR modeli için Tablo A.5'te gösterilmiştir (Burukanlı ve Yumuşak, 2024).

3.5.2. Önerilen StackGridCov ile HyperAttCov modeli için kullanılan veri seti hakkında detaylı bilgiler

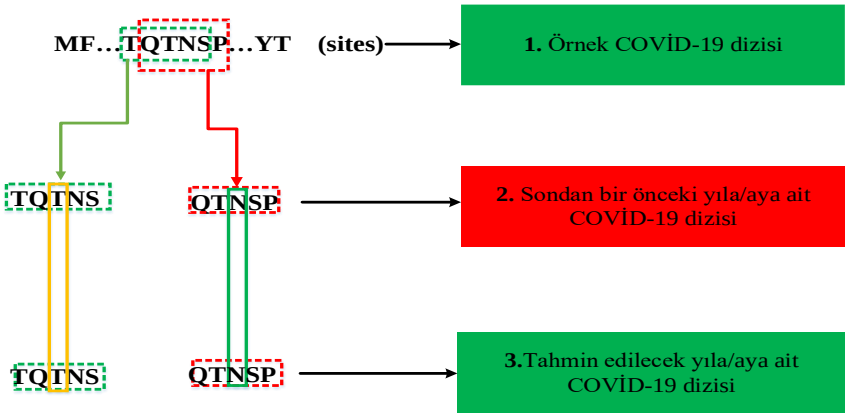
3.5.2.1. COVİD-19 (SARS-CoV-2) S protein veri seti

Bu tez çalışmasında Zhou ve ark. (2023a), tarafından hazırlanan COVİD-19 S protein veri seti Zhou ve ark. (2023b), COVİD-19 mutasyon tahmini için kullanılmıştır. Zhou ve ark. (2023a), Ocak 2020'den Şubat 2022'ye kadar GISAID veri tabanından toplam 8 milyondan fazla COVİD-19 S protein suşu/dizisini indirdiler (Shu ve McCauley, 2017). Onlar, 8 milyondan fazla COVİD-19 S protein dizisini, ön işleme

aşamalarından geçirdiler. Daha sonra tasarladıkları filogenetik ağaç tabanlı örnekleme yöntemini kullanarak toplam 5 uzunlukta 5758 (3 örtüşen-3 gram) eğitim ve test veri seti oluşturdular. Bu tez çalışmasında Zhou ve ark. (2023a) tarafından hazırlanan önceden işlenmiş 5758 adet (3 örtüşen-3 gram) COVİD-19 S protein verisi kullanılmıştır. Bu aşamada eğitim ve test veri kümelerinin önerilen StackGridCov ve HyperAttCov modelleri ile diğer modellerin eğitim ve test işlemleri için bazı ön işlem adımlarından geçirilmesi gerekmektedir. Bu nedenle biz eğitim ve test veri setlerini bazı ön işleme adımlarına tabi tuttuk. Eğitim ve test veri setleri için eğitim ve test veri örnekleri oluşturma aşamaları Şekil **3.15**'te ve etiket veri örnekleri oluşturma aşamaları ise Şekil **3.16**'da gösterilmiştir.

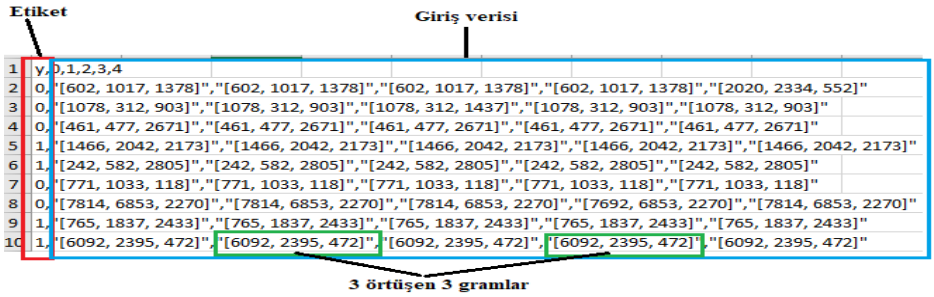


Şekil 3.15. Eğitim ve test veri kümeleri için eğitim ve test veri örnekleri oluşturma aşamaları (Burukanlı ve Yumuşak, 2024c).



Şekil 3.16. Veri kümelerini eğitmek ve test etmek için etiket veri örnekleri oluşturmanın aşamaları (Burukanlı ve Yumuşak, 2024b).

Bu tez çalışmasında kullanılan COVİD-19 veri seti iki sınıftan oluşmaktadır (“mutasyon” sınıfı ve “mutasyon yok” sınıfı). "mutasyon" sınıfında eğitim için 2314 veri (3 örtüşme-3 gram) ve test için 565 veri (3 örtüşme-3 gram) olmak üzere toplam 2879 veri (3 örtüşme-3 gram) bulunmaktadır. "mutasyon yok" sınıfında eğitim için 2292 veri (3 örtüşme-3 gram) ve test için 587 veri (3 örtüşme-3 gram) olmak üzere toplam 2879 veri (3 örtüşme-3 gram) bulunmaktadır. Bu veri setinde 5758 adet (3 örtüşen-3 gram), "mutasyon" sınıfında 2879, "mutasyon yok" sınıfında ise 2879 adet veri bulunmaktadır. Zhou ve ark. (2023a) tarafından sunulan işlenmiş COVİD-19 virüs veri seti ve detayları Şekil **3.17**'de gösterilmiştir. Şekil **3.17**'de de görüldüğü üzere işlenmiş COVİD-19 veri kümesi, etiket ve giriş verilerinden oluşmaktadır. Etiket değeri 1 ise “mutasyon var”, 0 ise “mutasyon yok” anlamına geliyor. Her giriş verisi, 5 eğitim örneğinden oluşur (3 örtüşen - 3 gram). Her üç gram, Asgari ve ark. (2015) tarafından sunulan ProtVec'e dayalı 100 boyutlu bir gömme vektörüyle temsil edilir. Modelin eğitim aşamasında her üç gramlık 100 boyutlu vektör toplanarak 100 boyutlu tek bir vektör kullanılır.



Şekil 3.17. işlenmiş COVİD-19 virüs veri seti.

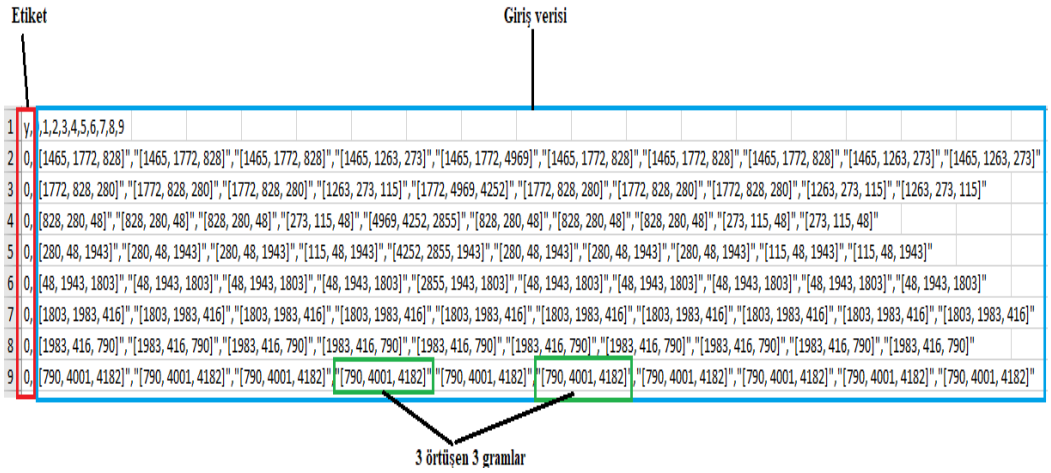
COVİD-19 veri setinin eğitim, test, Kfold ve toplam miktarları ve yaklaşık yüzdeleri Tablo 1'de gösterilmiştir.

Tablo 3.10. COVİD-19 veri setinin eğitim, test, Kfold ve toplam miktarları ve yaklaşık yüzdeleri (Burukanlı ve Yumuşak, 2024b) (Burukanlı ve Yumuşak, 2024c).

Veri seti	“Mutasyon var” sınıfı	“Mutasyon yok” sınıfı	Toplam veri
Eğitim	2314 (50.24%)	2292 (49.76%)	4606 (100%)
Test	565 (49.05%)	587 (50.95%)	1152 (100%)
Kfold	2879 (50%)	2879 (50%)	5758 (100%)

3.5.2.2. İnfluenza A/ H1N1 HA veri seti

Bu tez çalışmasında önerilen StackGridCov modelinin performansını ölçmek amacıyla daha önce ortaya çıkan influenza A/H1N1 HA virüs alt tipinin HA protein veri setleri üzerinde test edilmiştir. Yin ve ark. (2020) tarafından sunulan influenza veri kümeleri, 1991 ile 2016 yılları arasındaki HA protein dizilerinden oluşmaktadır. İnfluenza A/H1N1 HA veri kümesi, eğitim veri kümesinde 128.800 dizi örneğinden (3 gram örtüşen) ve test veri kümesinde 32.200 dizi örneğinden (3 gram örtüşen) olmak üzere toplamda 161.000 dizi örneğinden (3 gram örtüşen) oluşmaktadır (Yin et al., 2020). Eğitim veri seti, mutasyon sınıfında 18.886 örnek ve mutasyon olmayan sınıfta 109.914 örnek olmak üzere toplamda 128.800 örnekten meydana gelmektedir. Benzer şekilde, test veri seti mutasyon sınıfında 4.634 örnek ve mutasyon olmayan sınıfta 27.566 örnek olmak üzere toplamda 32.200 örnekten meyadana gelmektedir. İşlenmiş influenza A/H1N1 HA virüs veri seti ve detayları Şekil **3.18**'de gösterilmiştir. Şekil **3.18**'de de görüldüğü üzere işlenmiş COVİD-19 veri kümesi, etiket ve giriş verilerinden oluşmaktadır. Etiket değeri 1 ise "mutasyon var", 0 ise "mutasyon yok" anlamına geliyor. Her giriş verisi, 5 eğitim örneğinden oluşur (3 örtüşen - 3 gram). Her üç gram, Asgari ve ark. (2015) tarafından sunulan ProtVec'e dayalı 100 boyutlu bir gömme vektörüyle temsil edilir. Modelin eğitim aşamasında her üç gramlık 100 boyutlu vektör toplanarak 100 boyutlu tek bir vektör kullanılır.



Şekil 3.18. İşlenmiş Influenza A/H1N1 HA virüs veri seti.

Yin ve ark. (2020) tarafından sunulan influenza A/H1N1 HA protein veri setinin eğitim ve test veri setleri için sınıf miktarları, Tablo **3.11**'de gösterilmiştir.

Tablo 3.11. İnfluenza A/H1N1 HA protein veri kümesinin veri kümelerinin eğitimi ve test edilmesi için sınıf miktarları (Burukanlı ve Yumuşak, 2024b).

Veri seti	"Mutasyon var" sınıfı	"Mutasyon yok" sınıfı	Toplam veri
Eğitim	18886	109914	128800
Test	4634	27566	32200

4. ARAŞTIRMA BULGULARI VE TARTIŞMA

Bu tez çalışmasında önerilen TfrAdmCov Burukanlı ve Yumuşak (2024a), StackGridCov Burukanlı ve Yumuşak (2024b) ve HyperAttCov Burukanlı ve Yumuşak (2024c) modelleri ile diğer yapay zeka tabanlı modelerin hem COVİD-19 virüsü hemde inflenza virüsü üzerinde mutasyon tahmini için elde edilen bulgular tartışılmıştır.

4.1. ÖnerilenTfrAdmCov Modeli İçin Elde Edilen Bulgular

4.1.1. Uygulama detayları

Bu tez çalışmasında, derin öğrenme modellerinin (RNN, LSTM, GRU, Transformer) performansını en üst düzeye çıkarmak amacıyla hiperparametre değerleri (hidden size, dropout, batch size vb.) birçok kez (deneme yanılma) test edilmiştir ve en iyi hiper parametre değerleri seçilmiştir. Tüm derin öğrenme modellerinde (geleneksel makine öğrenme modelleri hariç), model optimizasyonu için batch size boyutu 32 (H3N2 HA veri seti için 256) olan Adam kullanılmıştır. Önerilen TfrAdmCov modelinin ve tüm derin öğrenme modellerinin kodlayıcısında öğrenme oranı 0.001 ve hidden size 128 olarak ayarlanmıştır. Amaç fonksiyonu olarak (kayıpları en aza indirmek için) çapraz entropi kullanılmıştır. Önerilen TfrAdmCov modelinin ve tüm derin öğrenme modellerinin eğitimi için 0.5 dropout değeri ve 500 epok değeri (H3N2 HA veri seti için 350 epok) kullanılmıştır. Ayrıca transformer kodlayıcı katmanında kullanılan çoklu kafa dikkatinin (multi head attention) sayısı (açık kaynak kütüphanesinde (Vaswani ve ark., 2017) varsayılan değeri dahil = 8) birçok kez test edilmiş ve en iyi hiperparametre değeri (çoklu kafa dikkati = 2) seçilmiştir. Tüm deneysel sonuçların, derin öğrenme modelleri için farklı rastgele tohumlara (seeds) sahip 10 rastgele denemenin ortalaması alınmıştır. Önerilen TfrAdmCov modelinin ve diğer modellerin hiperparametreleri ve bu hiperparemetrelerin değerleri Tablo **4.1**'de gösterilmiştir.

Tablo 4.1. Önerilen TfrAdmCov modelinin ve diğer modellerin hiperparametreleri (Burukanlı ve Yumuşak, 2024a).

Hiper-parametre adı	Değeri
Hidden Size	128
Dropout	0.5
Batch Size	32
Öğrenme oranı	0.001
Epok	500
Optimizer algoritması	Adam
Kayıp fonksiyonu	cross entropy
Çoklu kafa dikkat sayısı (Transformer kodlayıcı için)	2
Transformer kodlayıcı sayısı	1

4.1.2. Modellerin performanslarını değerlendirme

Bu tez çalışmasında eğitim, test ve Kfold veri setlerinin eğitim ve test işlemleri 2-çekirdekli Intel(R) Core(TM) i5 7200U CPU@ 2.5 GHz işlemcili, 12GB Ram ve Intel(R) HD Graphics 620 GPU sahip bir bilgisayarda gerçekleştirilmiştir. Tüm eğitim, test ve similasyonlar, makine öğrenmesi ve derin öğrenme modelleri için oldukça populer olan Python'un Scikit-learn Pedregosa ve ark. (2011) ve PyTorch Paszke ve ark. (2017) kütüphaneleri kullanılarak elde edilmiştir. Bu tez çalışmasında derin öğrenme ve makine öğrenmesi tabanlı modellerin (doğruluk (accuracy), kesinlik (precision), hassasiyet (recall), F1-skor ve matthews korelasyon katsayısı (MCC)) performans ölçümleri Tablo **4.2**'deki hata (confusion) matrisi kullanılarak elde edilmiştir.

Tablo 4.2. Hata matrisi (Luque ve ark., 2019).

		Tahmin edilen sınıf	
		Pozitif	Negatif
Gerçek sınıf	Pozitif	Gerçek Pozitif (TP)	Yanlış Negatif (FN)

Tablo 4.2. (Devamı) Hata matrisi (Luque ve ark., 2019).

		Tahmin edilen sınıf	
		Pozitif	Negatif
Gerçek sınıf	Negatif	Yanlış Pozitif (FP)	Gerçek Negatif (TN)

Gerçek Pozitif (TP), aslında pozitif (mutasyon) olan ve tahmin edildiğinde de pozitif (mutasyon) olarak sınıflandırılan numuneleri ifade eder. Yanlış Negatif (FN), gerçekte pozitif (mutasyon) olan ve tahmin edildiğinde de negatif (mutasyon yok) olarak sınıflandırılan numuneleri ifade eder. Yanlış Pozitif (FP), aslında negatif olan (mutasyon yok) ve tahmin edildiğinde de pozitif (mutasyon) olarak sınıflandırılan numuneleri ifade eder. Gerçek Negatif (TN), aslında negatif olan (mutasyon yok) ve tahmin edildiğinde negatif (mutasyon yok) olarak sınıflandırılan numuneleri ifade eder (Chicco ve Jurman, 2020). Bu tez çalışmasında kullanılan doğruluk, kesinlik, hassasiyet, F1-skor ve MCC performans değerlendirme metrikleri sırasıyla denklem (4.1), denklem (4.2), denklem (4.3), denklem (4.4), denklem (4.5)'te verilmiştir (Pacal, 2024b).

$$Doğruluk\ (Accuracy) = \frac{TP + TN}{TP + FN + FP + TN} \tag{4.1}$$

$$\text{Kesinlik (Precision)} = \frac{TP}{TP + FP} \tag{4.2}$$

$$Hassasiyet\ (Recall\ or\ Sensitivity) = \frac{TP}{TP + FN} \tag{4.3}$$

$$F1 - Skor = 2 * \frac{\text{Kesinlik } * Hassasiyet}{\text{Kesinlik } + Hassasiyet} \tag{4.4}$$

$$MCC = \frac{TP\ *\ TN\ -\ FP\ *\ FN}{\sqrt{(TP\ +\ FP)\ *\ (TP\ +\ FN)\ *\ (TN\ +\ FP)\ *\ (TN\ +\ FN)}} \tag{4.5}$$

4.1.3. Deneysel bulgular

GridSearchCV'li veya GridSearchCV'siz SVM modelinin performans değerleri, Tablo **4.3**'te gösterilmiştir.

Tablo 4.3. GridSearchCV'li veya GridSearchCV'siz SVM modelinin performans değerleri (Burukanlı ve Yumuşak, 2024a).

Model	Hyper-parametre ayarlama	Veri seti	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
SVM	GridSearchCV'li	Test	99.91	100.00	96.51	98.22	98.19
	GridSearchCV'siz	Test	99.90	100.00	96.22	98.07	98.04

Tablo **4.3**'te de görüldüğü üzere, GridSearchCV'li veya GridSearchCV'siz SVM modeli eğitim veri seti üzerinde eğitilmiş ve test veri seti üzerinde test edilmiştir. GridSearchCV yöntemine sahip SVM modelinin, test veri kümesinde GridSearchCV yöntemine sahip olmayan SVM modeline göre doğruluk değerini (%99.90'dan %99.91'e), hassasiyet değerini (%96.22'den %96.51'e), F1-skor değerini (%98.07'den %98.22'ye) ve MCC değerini (%98.04'ten %98.19'a) yükseltmiştir. Sonuç olarak, GridSearchCV yöntemine sahip olan SVM modelinin eğitim veri seti ve test veri seti üzerinde performansı önemli ölçüde arttırdığı gözlemlenmiştir. GridSearchCV'li veya GridSearchCV'siz KNN modelinin performans değerleri, Tablo **4.4**'te gösterilmiştir.

Tablo 4.4. GridSearchCV'li veya GridSearchCV'siz KNN modelinin performans değerleri (Burukanlı ve Yumuşak, 2024a).

Model	Hyper-parametre ayarlama	Veri seti	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
KNN	GridSearchCV'li	Test	99.91	99.40	97.09	98.24	98.19
	GridSearchCV'siz	Test	99.90	99.40	96.80	98.09	98.04

Tablo **4.4**'te de görüldüğü üzere, GridSearchCV'li veya GridSearchCV'siz KNN modeli eğitim veri seti üzerinde eğitilmiş ve test veri seti üzerinde test edilmiştir. GridSearchCV yöntemine sahip KNN modelinin test veri kümesinde GridSearchCV yöntemine sahip olmayan KNN modeline göre doğruluk değerini (%99,90'dan %99,91'e), hassasiyet değerini (%96.80'den %97.09'a), F1-skor değerini (%98.09'dan %98.24'e) ve MCC değerini (%98.04'ten %98.19'a) artmıştır. Sonuç olarak, GridSearchCV yöntemine sahip KNN modelinin eğitim veri seti ve test veri seti üzerinde performansı önemli ölçüde arttırdığı gözlemlenmiştir. GridSearchCV'li veya

GridSearchCV'siz XGBoost modelinin performans değerleri, Tablo **4.5**'te gösterilmiştir.

Tablo 4.5. GridSearchCV'li veya GridSearchCV'siz XGBoost modelinin performans değerleri (Burukanlı ve Yumuşak, 2024a).

Model	Hyper-parametre ayarlama	Veri seti	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
XGBoost	GridSearchCV'li	Test	99.91	100.00	96.51	98.22	98.19
	GridSearchCV'siz	Test	99.90	99.70	96.51	98.08	98.04

Tablo **4.5**'te de görüldüğü üzere, GridSearchCV'li veya GridSearchCV'siz XGBoost modeli eğitim veri seti üzerinde eğitilmiş ve test veri seti üzerinde test edilmiştir. GridSearchCV yöntemine sahip XGBoost modelinin test veri seti üzerinde GridSearchCV yöntemi olmayan XGBoost modeline göre doğruluk değerini (%99.90'dan %99.91'e), hassasiyet değerini (%99.70'ten %100.00'a), F1-skor değerini (%98.08'den %98.22'ye) ve MCC değerini (%098.04'ten %98.19'a) yükseltmiştir. Sonuç olarak, GridSearchCV yöntemi ile XGBoost modelinin test veri seti üzerinde performansı önemli ölçüde arttırdığı gözlemlenmiştir. GridSearchCV'li veya GridSearchCV'siz LR modelinin performans değerleri, Tablo **4.6**'da gösterilmiştir.

Tablo 4.6. GridSearchCV'li veya GridSearchCV'siz LR modelinin performans değerleri (Burukanlı ve Yumuşak, 2024a).

Model	Hyper-parametre ayarlama	Veri seti	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
LR	GridSearchCV'li	Test	99.79	98.18	93.90	95.99	95.90
	GridSearchCV'siz	Test	99.57	98.65	85.17	91.42	91.46

Tablo **4.6**'da da görüldüğü üzere, GridSearchCV'li veya GridSearchCV'siz LR modeli eğitim veri seti üzerinde eğitilmiş ve test veri seti üzerinde test edilmiştir. GridSearchCV yöntemine sahip LR modelinin test veri seti üzerinde GridSearchCV yöntemi olmayan LR modeline göre doğruluk değerini (%99.57'den -%99.79'a), hassasiyet değerini (%85.17'den %93.90'a), F1-skor değerini (%91.42'den %95.99'a) ve MCC değerini (%91.64'ten %95.90'a) yükseltmiştir. Sonuç olarak, GridSearchCV yöntemine sahip LR modelinin eğitim veri seti ve test veri seti üzerinde performansı

önemli ölçüde arttırdığı gözlemlenmiştir. GridSearchCV'li veya GridSearchCV'siz makine öğrenimi tabanlı modellerin performans değerlerinin karşılaştırılması, Tablo **4.7**'de gösterilmiştir.

Tablo 4.7. GridSearchCV'li veya GridSearchCV'siz Makine öğrenimi tabanlı modellerin performans değerlerinin karşılaştırılması (Burukanlı ve Yumuşak, 2024a).

Model	Veri seti	Hyper-parametre ayarlama	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
SVM	Kfold	GridSearchCV'li	**99.893**	**99.92**	96.14	**97.97**	**97.95**
		GridSearchCV'siz	99.84	**99.92**	94.32	97.00	96.98
KNN	Kfold	GridSearchCV'li	99.89	99.69	**96.22**	97.91	97.88
		GridSearchCV'siz	99.86	99.75	94.91	97.24	97.21
XGBoost	Kfold	GridSearchCV'li	99.88	99.62	96.00	97.75	97.72
		GridSearchCV'siz	99.88	99.69	95.93	97.75	97.72
LR	Kfold	GridSearchCV'li	99.79	97.14	94.90	95.99	95.89
		GridSearchCV'siz	99.42	98.27	79.76	88.04	88.25

Tablo **4.7**'de de görüldüğü gibi, GridSearchCV'li ve GridSearchCV'siz makine öğrenimi tabanlı modeller (SVM, KNN, XGBoost, LR) kullanılarak Kfold veri seti üzerinde doğruluk, kesinlik, hassasiyet, F1-skor ve MCC değerlerinin karşılaştırılması yapılmıştır. GridSearchCV yöntemine sahip SVM modeli, Kfold veri seti üzerinde %99.893 doğruluk değeriyle en yüksek performansı elde etmiştir. GridSearchCV'li veya GridSearchCV'siz SVM modeli, Kfold veri kümesinde %99.92 kesinlik değeriyle en yüksek performansı elde etmiştir. GridSearchCV yöntemine sahip SVM modeli, Kfold veri kümesinde %97.97 F1-skor değeriyle en yüksek performansı elde etmiştir. GridSearchCV yöntemine sahip KNN modeli ise Kfold veri seti üzerinde %96.22 F1-skor değeriyle en yüksek performansı elde etti. GridSearchCV yöntemine sahip SVM modeli, Kfold veri kümesinde %97.95 MCC değeriyle en yüksek performansı elde etti. Sonuçlar detaylı analiz edildiğinde, GridSearchCV hyperparametre ayarlama metodu modellerin başarımlarını arttırmada çıddi oranda

etkili olmuştur. Test veri kümesi üzerinde önerilen TfrAdmCov modeli ile derin öğrenme modellerinin performans karşılaştırması, Tablo **4.8**'de gösterilmiştir.

Tablo 4.8. Test veri kümesi üzerinde önerilen TfrAdmCov modeli ile derin öğrenme modellerinin performans karşılaştırması (Burukanlı ve Yumuşak, 2024a).

Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
RNN	99.92	**100.00**	97.09	98.53	98.50
LSTM	99.91	**100.00**	96.80	98.38	98.34
GRU	99.91	**100.00**	96.80	98.38	98.34
TfrAdmCov	**99.93**	**100.00**	**97.38**	**98.67**	**98.65**

Tablo **4.8**'de görüldüğü üzere, önerilen TfrAdmCov modeli, COVİD-19 test veri setinde %99.93 ile doğruluk, %97.38 ile hassasiyet, %98.67 ile F1-skor ve %98.65 ile MCC değeri açısından diğer modellere göre daha iyi sonuçlar elde etmiştir. Ayrıca önerilen TfrAdmCov modeli, RNN, LSTM, GRU, COVİD-19 test veri setinde %100.00 ile kesinlik açısından aynı sonuçları elde etmiştir. Test veri seti üzerinde Adam, RMSprop, AdamW optimizasyon algoritmasına sahip önerilen TfrAdmCov modelinin performans karşılaştırması, Tablo **4.9**'da gösterilmiştir.

Tablo 4.9. Test veri seti üzerinde Adam, RMSprop, AdamW optimizasyon algoritmasına sahip önerilen TfrAdmCov modelinin performans karşılaştırması (Burukanlı ve Yumuşak, 2024a).

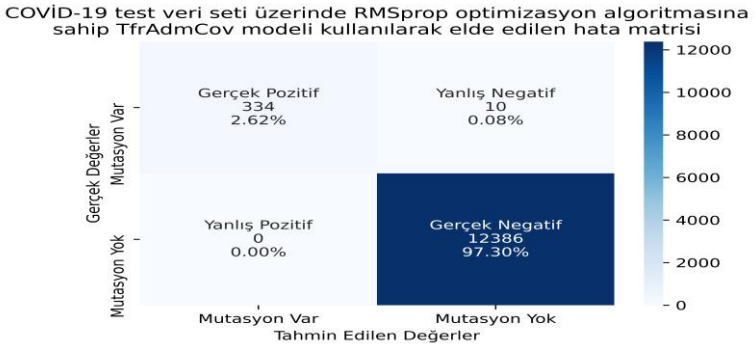
Optimizasyon algoritması	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
Adam	**99.93**	**100**	**97.383**	**98.67**	**98.65**
RMSprop	99.92	**100**	97.09	98.53	98.50
AdamW	99.92	**100**	97.09	98.53	98.50

Tablo **4.9**, önerilen TfrAdmCov modeli için üç optimizasyon algoritması (Adam, RMSprop, AdamW) arasından en iyi performansı elde eden optimizasyon algoritmayı seçmeyi amaçlamaktadır. Tablo **4.9**'da da görüldüğü üzere, Adam optimizasyon algoritmasına sahip TfrAdmCov modeli, RMSprop, AdamW optimizasyon

algoritmalarına sahip TfrAdmCov modelinden daha iyi performans göstermiştir. Bu nedenle, önerilen TfrAdmCov modeli için Adam optimizasyon algoritması tercih edilmiştir. COVİD-19 test veri seti üzerinde Adam optimizasyon algoritmasına sahip TfrAdmCov modeli kullanılarak elde edilen hata matrisi, Şekil **4.1**'de gösterilmiştir.

Şekil 4.1. COVİD-19 test veri kümesinde Adam optimizasyon algoritmasına sahip TfrAdmCov modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024a).

Şekil **4.1**'de görüldüğü gibi, COVİD-19 test veri setinde, Adam optimizasyon algoritmasına sahip TfrAdmCov modeli, "mutasyon" sınıfındaki 344 örnekten 335 örneği doğru tahmin ederken, "mutasyon" sınıfındaki 344 örnekten 9 örneği hatalı tahmin etmiştir. Ek olarak, Adam optimizasyon algoritmasına sahip önerilen TfrAdmCov modeli, "mutasyon yok" sınıftaki 12386 örnekten tüm örnekleri doğru bir şekilde tahmin etmiştir. COVİD-19 test veri kümesinde RMSprop optimizasyon algoritmasına sahip TfrAdmCov modeli kullanılarak elde edilen hata matrisi, Şekil **4.2**'de gösterilmiştir.



Şekil 4.2. COVİD-19 test veri kümesinde RMSprop optimizasyon algoritmasına sahip TfrAdmCov modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024a).

Şekil **4.2**'de görüldüğü gibi üzere COVİD-19 test veri setinde, RMSprop optimizasyon algoritmasına sahip önerilen TfrAdmCov modeli, "mutasyon" sınıfındaki 344 örnekten 334 örneği doğru tahmin ederken, "mutasyon" sınıfındaki 344 örnekten 10 örneği hatalı tahmin etmiştir. Ek olarak, RMSprop optimizasyon algoritmasına sahip önerilen TfrAdmCov modeli, "mutasyon yok" sınıftaki 12386 örnekten tüm örnekleri doğru bir şekilde tahmin etmiştir. COVİD-19 test veri kümesinde AdamW optimizer algoritmasına sahip TfrAdmCov modeli kullanılarak elde edilen hata matrisi, Şekil **4.3**'te gösterilmiştir.

Şekil 4.3. COVİD-19 test veri kümesinde AdamW optimizer algoritmasına sahip TfrAdmCov modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024a).

Şekil **4.3**'te de görüldüğü gibi, COVİD-19 test veri setinde, AdamW optimizasyon algoritmasına sahip önerilen TfrAdmCov modeli, "mutasyon" sınıfındaki 344 örnekten 334 örneği doğru tahmin ederken, "mutasyon" sınıfındaki 344 örnekten 10 örneği hatalı tahmin etmiştir. Ek olarak, AdamW optimizasyon algoritmasına sahip önerilen TfrAdmCov modeli, "mutasyon yok" sınıftaki 12386 örnekten tüm örnekleri doğru bir şekilde tahmin etmiştir. Test veri kümesi üzerinde farklı rastgele tohumlara (different random seeds) için 10 rastgele denemeye (10 random trail) sahip önerilen TfrAdmCov modelli ile RNN, LSTM, GRU modellerinin performans karşılaştırması, Tablo **4.10**'da gösterilmiştir.

Tablo 4.10. Test veri kümesi üzerinde farklı rastgele tohumlara (different random seeds) için 10 rastgele denemeye (10 random trail) sahip önerilen TfrAdmCov modelli ile RNN, LSTM, GRU modellerinin performans karşılaştırması (Burukanlı ve Yumuşak, 2024a).

Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
RNN	99.918	**100.00**	96.95	98.45	98.42
LSTM	99.916	**100.00**	96.89	98.42	98.39
GRU	99.914	**100.00**	96.83	98.39	98.36
TfrAdmCov	**99.924**	**100.00**	**97.18**	**98.57**	**98.54**

Tablo **4.10**'da görüldüğü gibi, önerilen TfrAdmCov modeli, COVİD-19 test veri setinde %99.924 ile doğruluk, %97.18 hassasiyet, %98.57 ile f1-skor ve %98.54 ile MCC açısından diğer modellere göre daha iyi sonuçlar elde etmiştir. Ayrıca önerilen TfrAdmCov modeli, RNN, LSTM, GRU, COVİD-19 test veri seti üzerinde %100.00 kesinlik açısından aynı sonuçları elde etmiştir. Önerilen TfrAdmCov modeli ile derin öğrenme modelleri ve GridSearchCV yöntemine sahip makine öğrenmesi tabanlı modellerin test veri seti üzerindeki performans karşılaştırmaları, Tablo **4.11**'de gösterilmiştir.

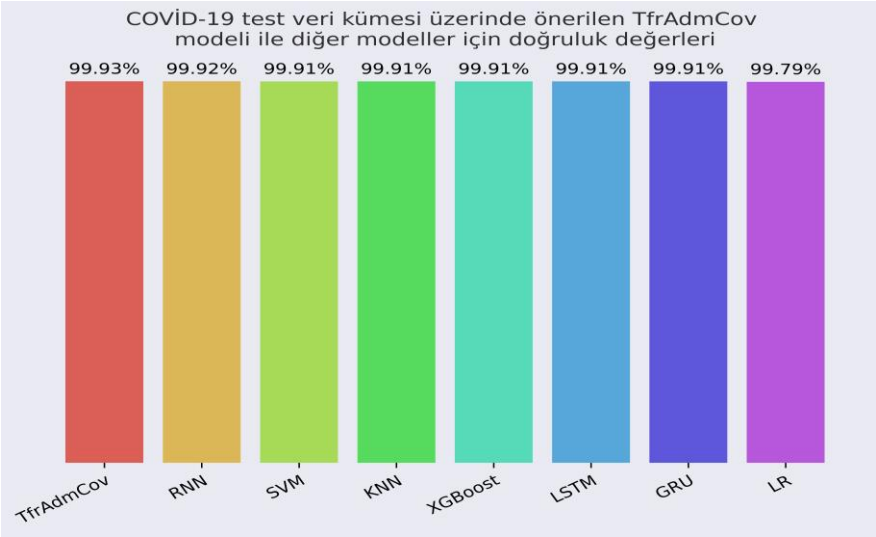
Tablo 4.11. Önerilen TfrAdmCov modeli ile derin öğrenme modelleri ve GridSearchCV yöntemine sahip makine öğrenmesi tabanlı modellerin test veri seti üzerindeki performans karşılaştırmaları (Burukanlı ve Yumuşak, 2024a).

Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
SVM	99.91	**100.00**	96.51	98.22	98.19
KNN	99.91	99.40	97.09	98.24	98.19
XGBoost	99.91	**100.00**	96.51	98.22	98.19
LR	99.79	98.18	93.90	95.99	95.90
RNN	99.92	**100.00**	97.09	98.53	98.50

Tablo 4.11. (Devamı) Önerilen TfrAdmCov modeli ile derin öğrenme modelleri ve GridSearchCV yöntemine sahip makine öğrenmesi tabanlı modellerin test veri seti üzerindeki performans karşılaştırmaları (Burukanlı ve Yumuşak, 2024a).

Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
LSTM	99.91	**100.00**	96.80	98.38	98.34
GRU	99.91	**100.00**	96.80	98.38	98.34
TfrAdmCov	**99.93**	**100.00**	**97.38**	**98.67**	**98.65**

Tablo **4.11**'de de görüldüğü gibi, önerilen TfrAdmCov modeli, COVİD-19 test veri setinde %99.93 ile doğruluk, %97.38 ile hassasiyet, %98.67 ile f1-skor ve %98.65 ile MCC değeri açısından diğer modellere göre daha iyi sonuçlar elde etmiştir. Ayrıca önerilen TfrAdmCov modeli, SVM, XGBoost, RNN, LSTM, GRU, COVİD-19 test veri seti üzerinde %100.00 kesinlik açısından aynı sonuçları elde etmiştir. Öte yandan LR modeli, COVİD-19 test veri setinde %99.79 ile doğruluk, %98.18 ile kesinlik, %93,90 ile hassasiyet, %95.99 ile f1-skor ve %95.90 ile MCC değeri açısından diğer modellere göre daha kötü sonuçlar elde etmiştir. Sonuç olarak, önerilen TfrAdmCov modeli, dizi bazlı COVİD-19 veri seti üzerinde oldukça başarılı sonuçlar elde etmiştir. Önerilen TfrAdmCov modeli ile diğer modellerin COVİD-19 test veri kümesi üzerindeki doğruluk değerleri, Şekil **4.4**'te gösterilmiştir.



Şekil 4.4. Önerilen TfrAdmCov modeli ile diğer modellerin COVİD-19 test veri kümesi üzerindeki doğruluk değerleri (Burukanlı ve Yumuşak, 2024a).

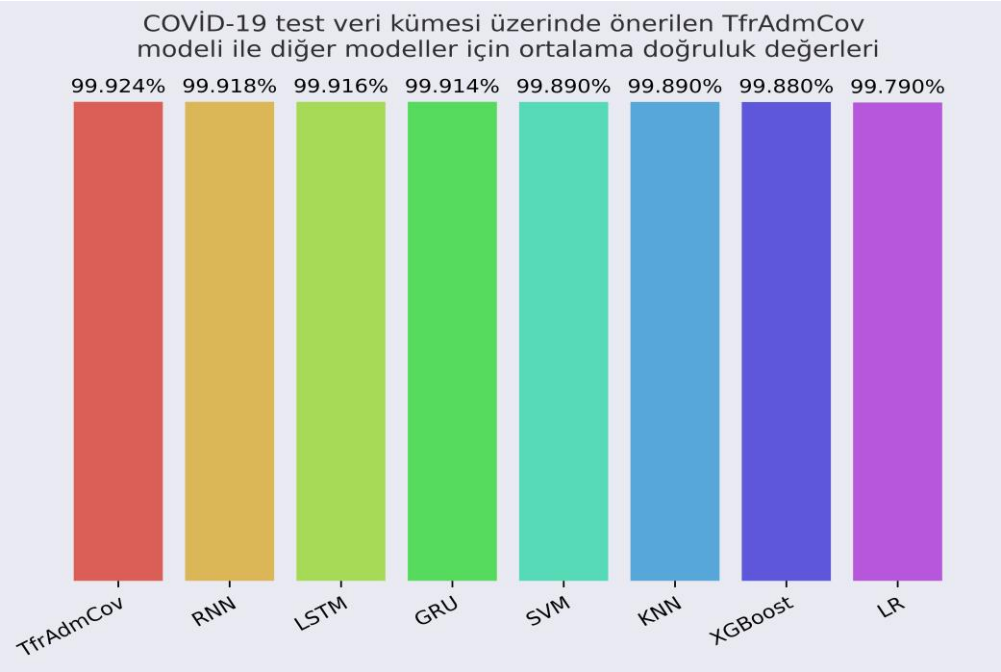
Şekil **4.4**'te de görüldüğü gibi önerilen TfrAdmCov modeli %99.93 ile en iyi doğruluk değerine ulaşırken, LR modeli %99.79 ile en kötü doğruluk değerine ulaşmıştır. Farklı rastgele tohumlar (different random seeds) için 10 rastgele denemeye (random trail) sahip önerilen TfrAdmCov modeli ile derin öğrenme modelleri ve stratified 10 kat çapraz doğrulama tekniğine sahip makine öğrenmesi algoritmalarının test veri kümesi üzerinde ortalama değerlerinin karşılaştırılması, Tablo **4.12**'de gösterilmiştir.

Tablo 4.12. Farklı rastgele tohumlar (different random seeds) için 10 rastgele denemeye (random trail) sahip önerilen TfrAdmCov modeli ile derin öğrenme modelleri ve stratified 10 kat çapraz doğrulama tekniğine sahip makine öğrenmesi algoritmalarının test veri kümesi üzerinde ortalama değerlerinin karşılaştırılması (Burukanlı ve Yumuşak, 2024a).

Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
SVM	99.89	99.92	96.14	97.97	97.95
KNN	99.89	99.69	96.22	97.91	97.88
XGBoost	99.88	99.62	96.00	97.75	97.72
LR	99.79	97.14	94.90	95.99	95.89
RNN	99.918	**100.00**	96.95	98.45	98.42
LSTM	99.916	**100.00**	96.89	98.42	98.39
GRU	99.914	**100.00**	96.83	98.39	98.36
TfrAdmCov	**99.924**	**100.00**	**97.18**	**98.57**	**98.54**

Tablo **4.12**'de de görüldüğü gibi, önerilen TfrAdmCov modeli, COVİD-19 test veri seti üzerinde %99.924 ile ortalama doğruluk, %97.18 ile ortalama hassasiyet, %98.57 ile ortalama F1-skor ve %98.54 ile ortalama MCC değeri açısından diğer modellerden daha iyi sonuçlar elde etmiştir. Ayrıca önerilen TfrAdmCov modeli, SVM, XGBoost, RNN, LSTM, GRU, COVİD-19 test veri seti üzerinde %100.00 ile ortalama kesinlik açısından aynı sonuçları elde etmiştir. Öte yandan LR modeli, COVİD-19 test veri setinde %99.79 ile ortalama doğruluk, %98.14 ortalama kesinlik, %94.90 ile ortalama hassasiyet, %95.99 ile ortalama F1-skor ve %95.89 ile ortalama MCC değeri açısından diğer modellere göre daha kötü sonuçlar elde etmiştir. Sonuç olarak, önerilen

TfrAdmCov modeli dizi bazlı COVİD-19 veri seti üzerinde oldukça başarılıdır. Önerilen TfrAdmCov modeli ile diğer modellerin COVİD-19 test veri kümesi üzerinde ortalama doğruluk değerleri, Şekil **4.5**'te gösterilmiştir.



Şekil 4.5. Önerilen TfrAdmCov modeli ile diğer modellerin COVİD-19 test veri kümesi üzerinde ortalama doğruluk değerleri (Burukanlı ve Yumuşak, 2024a).

Şekil **4.5**'te de görüldüğü gibi, önerilen TfrAdmCov modeli %99.924 ile en iyi ortalama doğruluk değerine ulaşırken, LR modeli %99.79 ile en kötü ortalama doğruluk değerine ulaşmıştır.

4.1.4. Önerilen TfrAdmCov modeli ile derin öğrenme modelleri için istatistiksel analizler

Bu tez çalışmasında elde edilen sonuçlar, hem önerilen TfrAdmCov modeli hem de derin öğrenme modelleri için farklı rastgele tohumlara sahip 10 rastgele denemenin ortalaması alınarak elde edilmiştir. Ortalama, standart sapma, medyan, minimum ve maksimum gibi istatistiksel ölçümler kullanılarak her model için doğruluk, kesinlik, hassasiyet, F1-puanı ve MCC performans ölçüm metriği açısından detaylı analizler yapılmıştır. Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip önerilen TfrAdmCov modelinin istatistiksel analizi, Tablo **4.13**'te gösterilmiştir.

Tablo 4.13. Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip önerilen TfrAdmCov modelinin istatistiksel analizi (Burukanlı ve Yumuşak, 2024a).

	Performans değerlendirme metrikleri				
İstatiksel Ölçüm	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC
Ortalama	0.999238	1.000000	0.971802	0.985699	0.985414
Standart sapma	0.000036	0.000000	0.001332	0.000685	0.000694
Medyan	0.999214	1.000000	0.970930	0.985251	0.984960
Minimum	0.999214	1.000000	0.970930	0.985251	0.984960
Maksimum	0.999293	1.000000	0.973837	0.986745	0.986474

Tablo **4.13**'te de görüldüğü gibi, önerilen TfrAdmCov modeli, test veri kümesi üzerinde elde edilen 10 doğruluk değeri arasında ortalama 0.999238, standart sapma 0.000036, medyan 0.999214, minimum 0.999214 ve maksimum 0.999293 olarak elde edilmiştir. Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip önerilen TfrAdmCov modelinin istatistiksel analizi, Tablo **4.14**'te gösterilmiştir.

Tablo 4.14. Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip RNN modelinin istatistiksel analizi (Burukanlı ve Yumuşak, 2024a).

	Performans değerlendirme metrikleri				
İstatiksel Ölçüm	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC
Ortalama	0.999175	1.000000	0.969476	0.984502	0.984202
Standart sapma	0.000039	0.000000	0.001453	0.000750	0.000757
Medyan	0.999175	1.000000	0.969476	0.984501	0.984202
Minimum	0.999136	1.000000	0.968023	0.983752	0.983445
Maksimum	0.999214	1.000000	0.970930	0.985251	0.984960

Tablo **4.14**'te de görüldüğü gibi, önerilen RNN modeli, elde edilen 10 doğruluk değeri arasında ortalama 0.999175, standart sapma 0.000039, medyan 0.999175, minimum 0.999136 ve maksimum 0.999214 olarak elde edilmiştir. Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip RNN modelinin istatistiksel analizi, Tablo **4.15**'te gösterilmiştir.

Tablo 4.15. Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip LSTM modelinin istatistiksel analizi (Burukanlı ve Yumuşak, 2024a).

	Performans değerlendirme metrikleri				
İstatiksel Ölçüm	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC
Ortalama	0.999159	1.000000	0.968895	0.984202	0.983899
Standart sapma	0.000036	0.000000	0.001332	0.000687	0.000694
Medyan	0.999136	1.000000	0.968023	0.983752	0.983445
Minimum	0.999136	1.000000	0.968023	0.983752	0.983445
Maksimum	0.999214	1.000000	0.970930	0.985251	0.984960

Tablo **4.15**'te de görüldüğü gibi, önerilen LSTM modeli, elde edilen 10 doğruluk değeri arasında ortalama 0.999159, standart sapma 0.000036, medyan 0.999136, minimum 0.999136 ve maksimum 0.999214 olarak elde edilmiştir. Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip LSTM modelinin istatistiksel analizi, Tablo **4.16**'da gösterilmiştir.

Tablo 4.16. Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip GRU modelinin istatistiksel analizi (Burukanlı ve Yumuşak, 2024a).

	Performans değerlendirme metrikleri				
İstatiksel Ölçüm	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC
Ortalama	0.999144	1.000000	0.968314	0.983902	0.983596
Standart sapma	0.000023	0.000000	0.000872	0.000450	0.000454
Medyan	0.999136	1.000000	0.968023	0.983752	0.983445

Tablo 4.16. (Devamı) Test veri kümesi üzerinde farklı rastgele tohumlar için 10 rastgele denemeye sahip GRU modelinin istatistiksel analizi (Burukanlı ve Yumuşak, 2024a).

	Performans değerlendirme metrikleri				
İstatiksel Ölçüm	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC
Minimum	0.999136	1.000000	0.968023	0.983752	0.983445
Maksimum	0.999214	1.000000	0.970930	0.985251	0.984960

Tablo **4.16**'da da görüldüğü gibi önerilen GRU modeli, elde edilen 10 doğruluk değeri arasından ortalama 0.999144, standart sapma 0.000023, medyan 0.999136, minimum 0.999136 ve maksimum 0.999214 olarak elde edilmiştir.

4.1.5. Eğitim, test ve Kfold veri setlerinin oluşturulmasında kmeans kümeleme algoritmasının yerine agglomerative kümeleme algoritmasının tercih edilmesinin nedeni

Bu tez çalışmasında eğitim, test ve Kfold veri kümelerini oluşturmak için ilk olarak Kmeans algoritmasını kullandık. Ancak Tablo **4.17**'de de görüldüğü gibi önerilen TfrAdmCov modelinin performansı agglomerative kümeleme algoritması tercih edildiğinde daha yüksek başarımlar elde edilmiştir. Bu nedenle eğitim, test ve Kfold veri setlerini oluşturmak için agglomerative kümeleme algoritması tercih edilmiştir. Önerilen TfrAdmCov modeli için kmeans ve agglomerative kümeleme algoritmaları kullanılarak oluşturulan test veri kümesi üzerinde performans karşılaştırması, Tablo **4.17**'de gösterilmiştir.

Tablo 4.17. Önerilen TfrAdmCov modeli için kmeans ve agglomerative kümeleme algoritmaları kullanılarak oluşturulan test veri kümesi üzerinde performans karşılaştırması (Burukanlı ve Yumuşak, 2024a).

Kümeleme algoritması	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
KMeans	99.62	88.52	96.38	92.28	92.17
Agglomerative	**99.93**	**100.00**	**97.38**	**98.67**	**98.65**

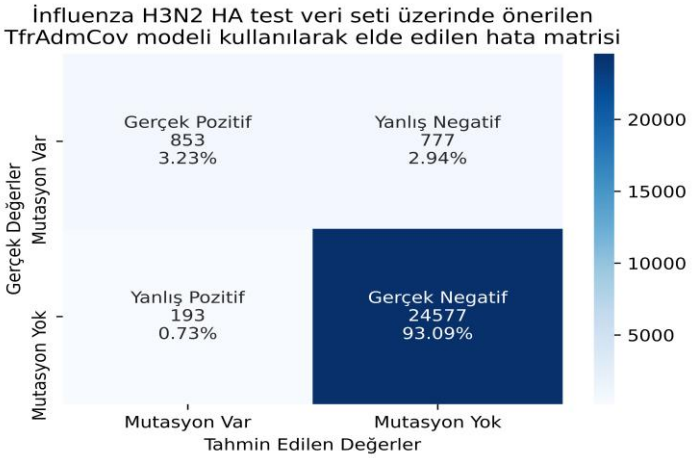
4.1.6. Önerilen TfrAdmCov modelinin influenza A/ H3N2 HA veri seti üzerinde performans değerlendirmesi

Önerilen TfrAdmCov modeli ile diğer modellerin H3N2 HA test veri kümesindeki performans değerleri, Tablo **4.18**'de gösterilmiştir.

Tablo 4.18. Önerilen TfrAdmCov modeli ile diğer modellerin H3N2 HA test veri kümesi üzerindeki performans değerleri (Burukanlı ve Yumuşak, 2024a).

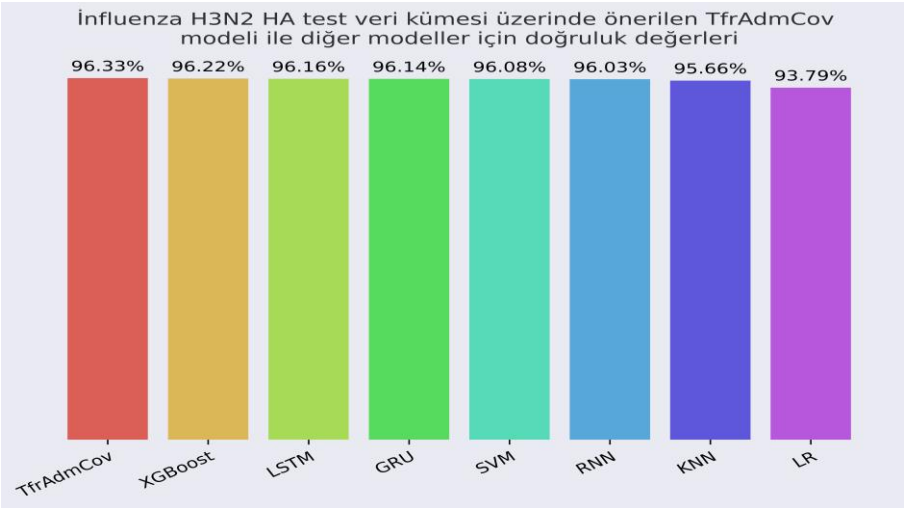
Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
SVM	96.08	80.96	47.30	60.05	60.39
KNN	95.66	70.11	51.66	59.48	57.99
XGBoost	96.22	81.54	50.12	62.08	62.19
LR	93.79	46.98	4.29	7.87	12.77
RNN	96.03	77.12	50.67	61.16	60.62
LSTM	96.16	78.68	51.84	62.50	62.03
GRU	96.14	79.79	50.12	61.57	61.44
TfrAdmCov	**96.33**	**81.55**	**52.33**	**63.75**	**63.61**

Tablo **4.18**'de de görüldüğü gibi, önerilen TfrAdmCov modeli, H3N2 HA test veri seti üzerinde %96.33 ile doğruluk, %81.55 ile kesinlik, %52.33 hassasiyet, %63.75 ile F1-skor ve %63.61 ile MCC değerlerinde diğer modellere göre daha iyi sonuçlar elde etmiştir. Diğer taraftan, LR modeli, test veri seti üzerinde %93.79 ile doğruluk, %46.98 ile kesinlik, %4.29 ile hassasiyet, %7.87 ile F1-skor ve %12.77 ile MCC değeri bakımından en kötü sonucu elde etmiştir. İnfluenza H3N2 HA test veri seti üzerindeki sonuçlar, önerilen TfrAdmCov modelinin oldukça sağlam olduğunu göstermiştir. H3N2 HA test veri seti üzerinde önerilen TfrAdmCov modeli kullanılarak elde edilen hata matrisi, Şekil **4.6**'da gösterilmiştir.



Şekil 4.6. H3N2 HA test veri seti üzerinde önerilen TfrAdmCov modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024a).

Şekil **4.6**'da da görüldüğü gibi, H3N2 HA test veri setinde önerilen TfrAdmCov modeli, "mutasyon" sınıfındaki 1630 örnekten 853 örneği doğru tahmin ederken, "mutasyon" sınıfındaki 1630 örnekten 777'sini hatalı tahmin etmiştir. Ayrıca önerilen TfrAdmCov modeli, "mutasyon yok" sınıfındaki ise 24770 örnekten 24577 örneği doğru tahmin ederken, "mutasyon yok" sınıfındaki 24770 örnekten 193 örneği hatalı tahmin etmiştir. Önerilen TfrAdmCov modeli ile diğer modellerin H3N2 HA test veri kümesi üzerindeki doğruluk değerleri, Şekil **4.7**'de gösterilmiştir.



Şekil 4.7. Önerilen TfrAdmCov modeli ile diğer modellerin H3N2 HA test veri kümesi üzerindeki doğruluk değerleri (Burukanlı ve Yumuşak, 2024a).

Şekil **4.7**'de de görüldüğü gibi önerilen TfrAdmCov modeli %96.33 ile en iyi doğruluk değerine ulaşırken, LR modeli ise %93.79 ile en kötü doğruluk değerine ulaşmıştır.

Önerilen TfrAdmCov modelinin son teknoloji (literatür) çalışmalarla karşılaştırılması, Tablo **4.19**'da gösterilmiştir.

Tablo 4.19. Önerilen TfrAdmCov modelinin son teknoloji (literatür) çalışmalarla karşılaştırılması (Burukanlı ve Yumuşak, 2024a).

Makale	Model	Veri seti	Doğruluk (%)
Mohamed ve ark. (2021)	LSTM	İnfluenza genom dizisi (DNA) (H1N1)	98.99
Haimed ve ark. (2021)	LSTM	COVİD-19 ORF7a Protein dizisi	72
Yin ve ark. (2020)	TEMPEL	İnfluenza Protein dizisi (H5N1)	99.1
Cai ve ark. (2024)	FluPMT	İnfluenza Protein dizisi (H1N1)	Rouge value (98.7)
Li ve ark. (2023)	GraphLncLoc	RNA dizisi	61.2
Yin ve ark. (2022)	IAV-CNN	İnfluenza Protein disisi (H1N1)	91.7
Zhou ve ark. (2023a)	TEMPO	COVİD-19 S Protein dizisi	**65.5**
Bizim çalışma	**TfrAdmCov**	COVİD-19 S Protein dizisi (test veri seti üzerinde)	**99.93**
Bizim çalışma	**TfrAdmCov**	COVİD-19 S Protein dizisi (farklı rastgele tohumlarla sahip 10 rastgele denemenin ortalaması alınarak)	**99.924**

Tablo **4.19**'da da görüldüğü gibi, önerilen TfrAdmCov modeli test veri kümesi üzerinde %99.93 doğruluk değerini elde etmiştir. Ayrıca önerilen TfrAdmCov modeli, farklı rastgele tohumlarla sahip 10 rastgele denemenin ortalamasını alındığında da %99.924 doğruluk değerine ulaşmıştır. Ayrıca, Tablo **4.19**'da da görüldüğü gibi önerilen TfrAdmCov modeli, en son teknolojiye sahip çalışmalardan daha iyi performans göstermiştir. Literatürde yayınlanan çalışmaların çoğunluğu ya COVİD-19 virüsünün diğer yönleriyle ya da diğer virüslerin mutasyonlarıyla ilgilidir.

Literatürdeki bu eksikliği bir nebze olsun gidererek bu çalışmayı gerçekleştirdik. Sonuç olarak, önerilen TfrAdmCov modeli, COVİD-19 veri seti üzerinde mutasyon tahminini başarıyla gerçekleştirebilmektedir.

4.2. Önerilen StackGridCov Modeli İçin Elde Edilen Bulgular

4.2.1. Uygulama detayları

Bu tez çalışmasında, tüm modeller için (makine öğrenimi ve topluluk öğrenme modelleri hariç) COVİD-19 S protein veri kümesinde, her biri için minimum batch size 256 olan Adam optimizasyonu (influenza A/H1N1 HA veri kümesi için RAdam) kullanılmıştır. Modellerin kodlayıcısında öğrenme oranı 0.001 olarak ayarlanmış ve hiden size 128 olarak seçilmiştir. Kaybı (loss) en aza indiren amaç fonksiyonu olarak çapraz entropi seçilmiştir. Tüm derin öğrenme modellerinin eğitimi için dropout =0.5 ve epok =100 (influenza A/H1N1 HA veri kümesi için epok =50) kullanılmıştır. Derin öğrenme modelleri için tüm deneysel sonuçlar, tüm epoklar için elde edilen değerlerin (doğruluk, kesinlik, hatırlama, F1-skor, MCC) ortalaması alınarak hesaplanmıştır. Önerilen StackGridCov modelinin ve diğer algoritmanın performansını ölçmek için alıcı işletim karakteristik eğrisi altındaki alan (AUC) kullanılmıştır. Hesaplanan AUC değeri 1'e ne kadar yakınsa hesaplanan performans o kadar iyidir (Fan et al., 2006).

4.2.2. Elde edilen bulgular

Bu bölümde önerilen StackGridCov modeli ile diğer modellerin hem COVİD-19 S protein veri seti hem de influenza A/H1N1 HA veri seti üzerinde elde edilen sonuçlar ve bu sonuçların detaylı analizi tartışılmaktadır.

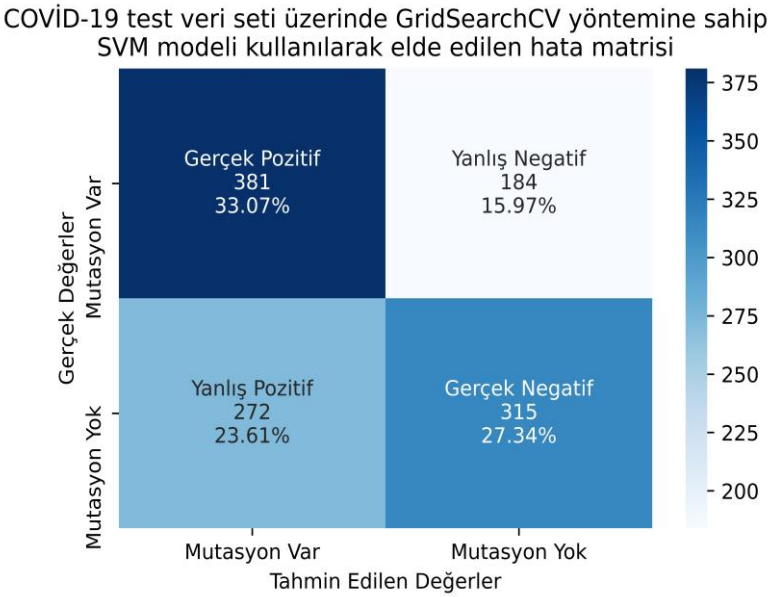
4.2.3. Önerilen StackGridCov modeli ile diğer modellerin COVİD-19 S protein veri seti üzerinde performans analizi

Her bir öğrenme algoritmasının (makine öğrenmesi, topluluk öğrenmesi ve derin öğrenme) performansı analiz edilmiştir. Ek olarak, GridSearchCV hiperparametre ayarlama tekniğine sahip olan veya olmayan her bir makine öğrenmesi ve topluluk öğrenme algoritmasının hata matrisleri şekillerle gösterilmiştir. GridSearchCV'li veya GridSearchCV'siz SVM algoritmasının test veri seti üzerindeki performans değerleri, Tablo **4.20**'de gösterilmiştir.

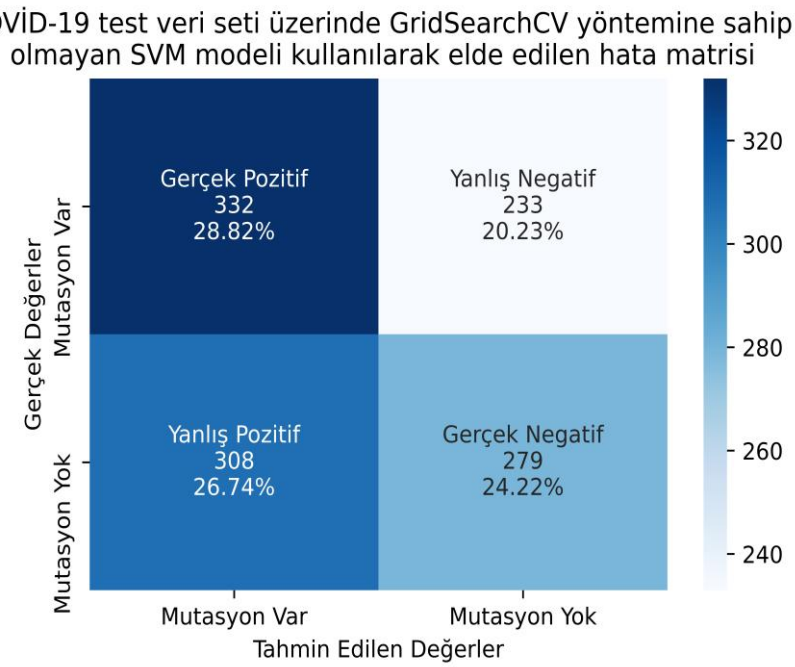
Tablo 4.20. GridSearchCV'li veya GridSearchCV'siz SVM algoritmasının test veri seti üzerindeki performans değerleri (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
SVM	GridSearchCV'li	0.6042	0.5835	0.6743	0.6256	0.2128	0.6114
	GridSearchCV'siz	0.5304	0.5188	0.5877	0.5510	0.0633	0.5443

Tablo **4.20**'de de görüldüğü gibi, GridSearchCV yöntemine sahip SVM algoritması, test veri kümesinde GridSearchCV yöntemine sahip olmayan SVM algoritmasından daha iyi performans göstermektedir. GridSearchCV yöntemine sahip SVM algoritması doğruluk değerini (0.5304'ten 0.6042'ye), kesinlik değerini (0.5188'den 0.5835'e), hassasiyet değerini (0.5877'den 0.6743'e), F1-Skor değerini (0.5510'dan 0.6256'ya), MCC değerini (0.0633'ten 0.2128'e) ve AUC değerini (0.5443'ten 0.6114'e kadar) arttırmıştır. Sonuç olarak, GridSearchCV yöntemine sahip SVM algoritmasının test veri seti üzerinde performansı önemli ölçüde arttırdığı gözlemlenmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip SVM modeli kullanılarak elde edilen hata matrisi, Şekil **4.8**'de gösterilmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan SVM modeli kullanılarak elde edilen hata matrisi, Şekil **4.9**'da gösterilmiştir.



Şekil 4.8. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip SVM modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).



Şekil 4.9. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan SVM modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

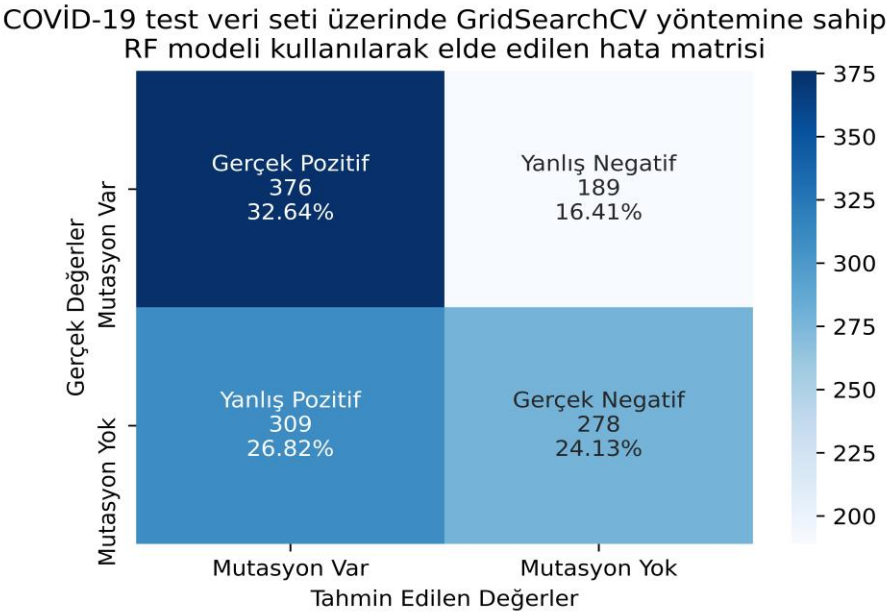
GridSearchCV'li veya GridSearchCV'siz RF algoritmasının performans değerleri, Tablo **4.21**'de gösterilmiştir.

Tablo 4.21. GridSearchCV'li veya GridSearchCV'siz RF algoritmasının performans değerleri (Burukanlı ve Yumuşak, 2024b).

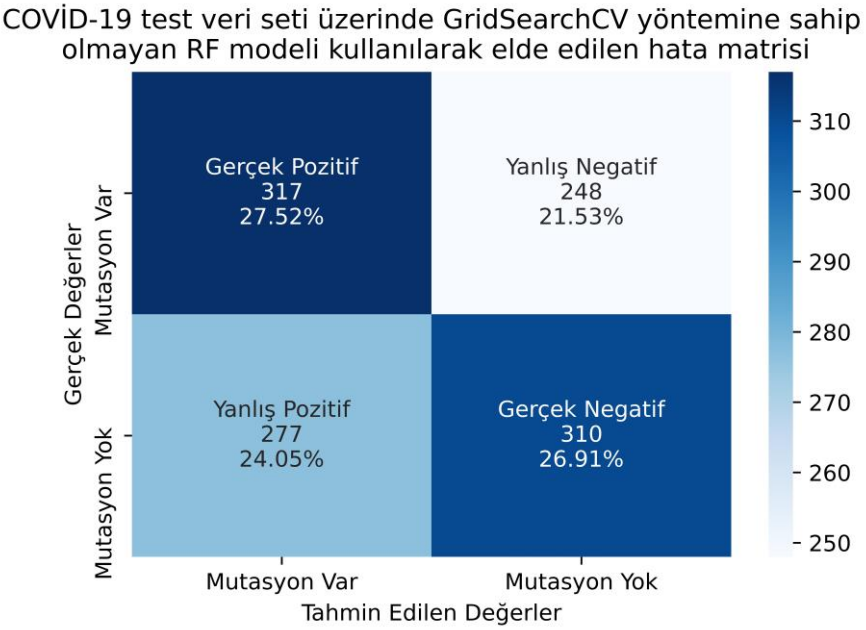
Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
RF	GridSearchCV'li	0.5677	0.5489	0.6655	0.6016	0.1416	0.5725
	GridSearchCV'siz	0.5443	0.5337	0.5611	0.5470	0.0892	0.5609

Tablo **4.21**'de de görüldüğü gibi, GridSearchCV yöntemine sahip RF algoritması, test veri kümesinde GridSearchCV yöntemine sahip olmayan RF algoritmasından daha iyi performans göstermektedir. GridSearchCV yöntemine sahip RF algoritması doğruluk değerini (0.5443'ten 0.5677'ye), kesinlik değerini (0.5337'den 0.5489'a), hassasiyet değerini (0.5611'den 0.6655'e), F1-skor değerini (0.5470'den 0.6016'ya), MCC değerini (0.0892'den 0.1416'ya) ve AUC değerini (0.5609'dan 0.5725'e) arttırmıştır. Sonuç olarak, GridSearchCV yöntemine sahip RF algoritmasının test veri seti üzerinde performansı önemli ölçüde arttırdığı gözlemlenmiştir. COVİD-19 test veri kümesi

üzerinde GridSearchCV yöntemine sahip RF modeli kullanılarak elde edilen hata matrisi, Şekil **4.10**'da gösterilmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan RF modeli kullanılarak elde edilen hata matrisi, Şekil **4.11**'de gösterilmiştir.



Şekil 4.10. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip RF modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).



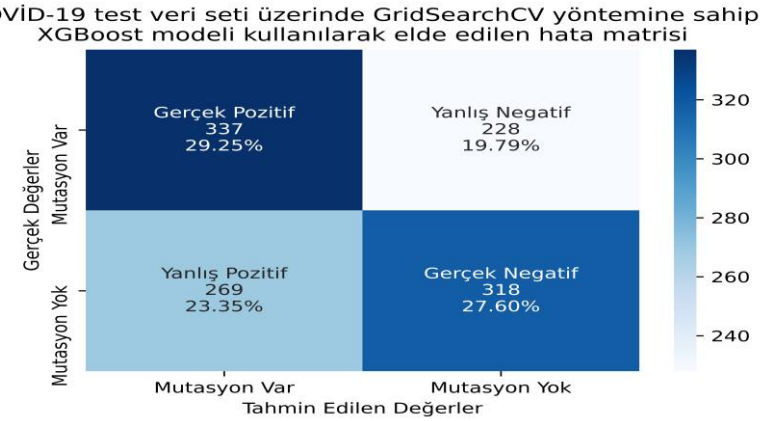
Şekil 4.11. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan RF modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

GridSearchCV'li veya GridSearchCV'siz XGBoost algoritmasının performans değerleri, Tablo **4.22**'de gösterilmiştir.

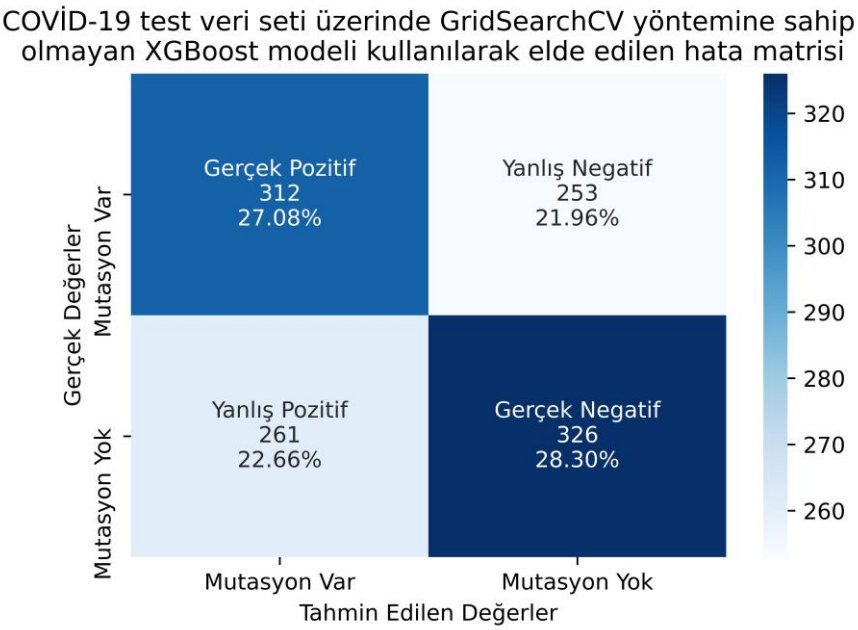
Tablo 4.22. GridSearchCV'li veya GridSearchCV'siz XGBoost algoritmasının performans değerleri (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
XGBoost	GridSearchCV'li	0.5686	0.5561	0.5965	0.5756	0.1384	0.5763
	GridSearchCV'siz	0.5538	0.5445	0.5522	0.5483	0.1076	0.5609

Tablo **4.22**'de de görüldüğü gibi, GridSearchCV yöntemine sahip XGBoost algoritması, test veri kümesinde GridSearchCV yöntemine sahip olmayan XGBoost algoritmasından daha iyi performans göstermektedir. GridSearchCV yöntemine sahip XGBoost algoritması doğruluk değerini (0.5538'den 0.5538'e), kesinlik değerini (0.5445'ten 0.5561'e), hassasiyet değerini (0.5522'den 0.5965'e), F1-Skor değerini (0.5483'ten 0.5756'ya), MCC değerini (0.1076'dan 0.1384'e) ve AUC değerini (0.5609'dan 0.5763'e kadar) yükseltmiştir. Sonuç olarak, GridSearchCV yöntemine sahip XGBoost algoritmasının test veri seti üzerinde performansı önemli ölçüde arttırdığı gözlemlenmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip XGBoost modeli kullanılarak elde edilen hata matrisi, Şekil **4.12**'de gösterilmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan XGBoost modeli kullanılarak elde edilen hata matrisi, Şekil **4.13**'te gösterilmiştir.



Şekil 4.12. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip XGBoost modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).



Şekil 4.13. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan XGBoost modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

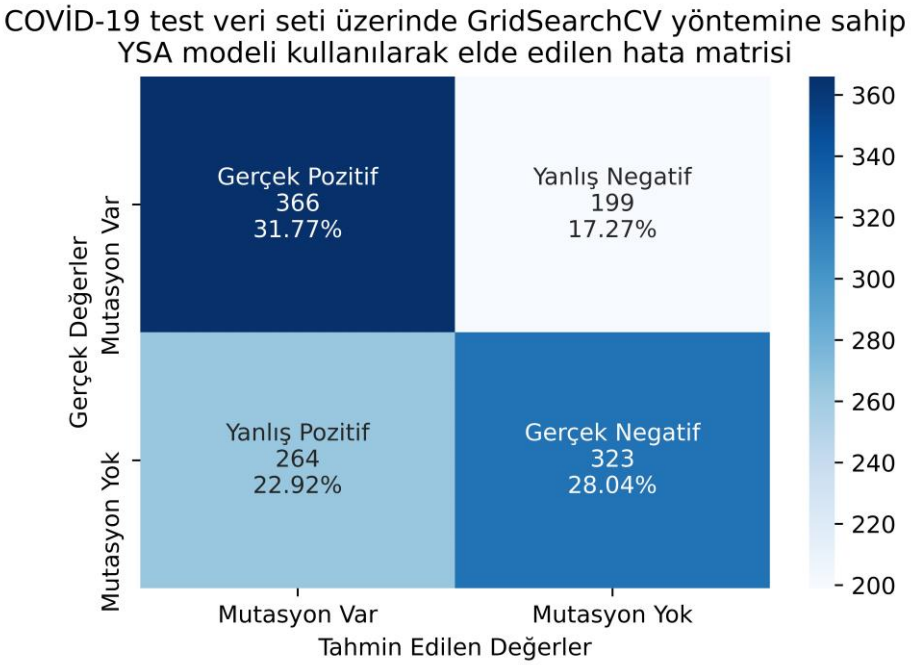
GridSearchCV'li veya GridSearchCV'siz YSA algoritmasının performans değerleri, Tablo **4.23**'te gösterilmiştir.

Tablo 4.23. GridSearchCV'li veya GridSearchCV'siz YSA algoritmasının performans değerleri (Burukanlı ve Yumuşak, 2024b).

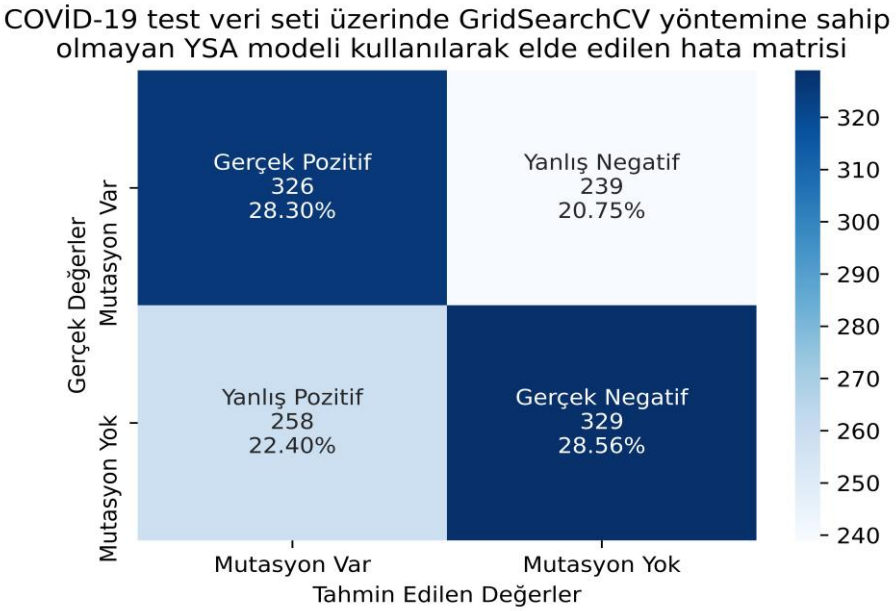
Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
YSA	GridSearchCV'li	0.5981	0.5810	0.6478	0.6126	0.1989	0.6054
	GridSearchCV'siz	0.5686	0.5582	0.5770	0.5675	0.1375	0.5836

Tablo **4.23**'te de görüldüğü gibi, GridSearchCV yöntemine sahip YSA algoritması, test veri kümesinde GridSearchCV yöntemine sahip olmayan YSA algoritmasından daha iyi performans göstermektedir. GridSearchCV yöntemine sahip YSA algoritması, doğruluk değerini (0.5686'dan 0.5981'e), kesinlik değerini (0.5582'den 0.5810'a), hassasiyet değerini (0.5770'den 0.6478'e), F1-skor değerini(0.5675'ten 0.6126'ya), MCC değerini (0.1375'ten 0.1989'a) ve AUC değerini (0.5836'dan 0.6054'e) artırmıştır. Sonuç olarak GridSearchCV yöntemine sahip YSA algoritmasının test veri seti üzerindeki performansı önemli ölçüde arttırdığı gözlemlenmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip YSA modeli kullanılarak elde

edilen hata matrisi, Şekil **4.14**'te gösterilmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan YSA modeli kullanılarak elde edilen hata matrisi, Şekil **4.15**'te gösterilmiştir.



Şekil 4.14. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip YSA modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).



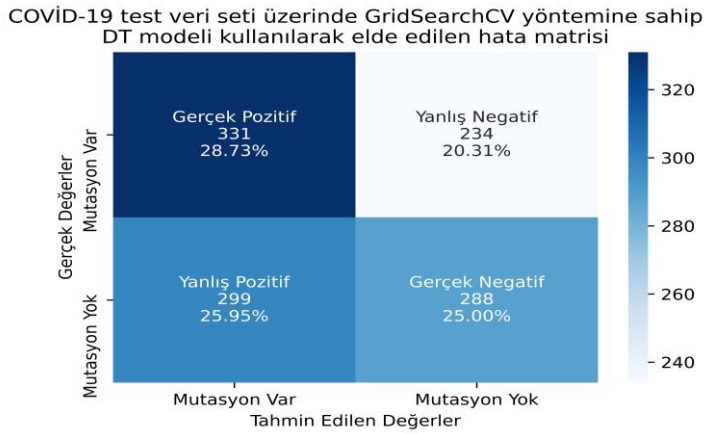
Şekil 4.15. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan YSA modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

GridSearchCV'li veya GridSearchCV'siz DT algoritmasının performans değerleri, Tablo **4.24**'te gösterilmiştir.

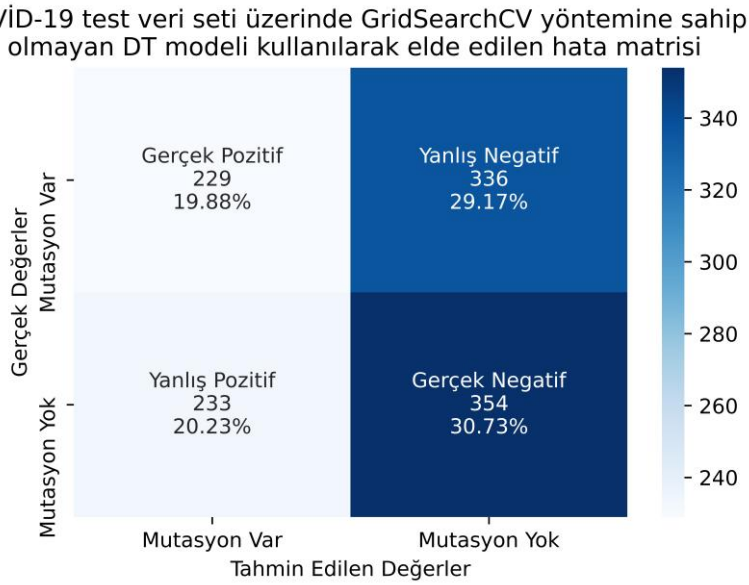
Tablo 4.24. GridSearchCV'li veya GridSearchCV'siz DT algoritmasının performans değerleri (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
DT	GridSearchCV'li	0.5373	0.5254	0.5858	0.5540	0.0768	0.5676
	GridSearchCV'siz	0.5061	0.4957	0.4053	0.4460	0.0085	0.5435

Tablo **4.24**'te de görüldüğü gibi, GridSearchCV yöntemine sahip DT algoritması, test veri kümesinde GridSearchCV yöntemine sahip olmayan DT algoritmasından daha iyi performans göstermektedir. GridSearchCV yöntemine sahip DT algoritması doğruluk değerini (0.5061'den 0.5373'e), kesinlik değerini (0.4957'den 0.5254'e), hassasiyet değerini (0.4053'ten 0.5858'e), F1-skor değerini (0.4460'dan 0.5540'a), MCC değerini (0.0085'ten 0.0768'e) ve AUC değerini (0.5435'ten 0.5676'ya) arttırmıştır. Sonuç olarak, GridSearchCV yöntemine sahip DT algoritmasının test veri seti üzerindeki performansı önemli ölçüde arttırdığı gözlemlenmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip DT modeli kullanılarak elde edilen hata matrisi, Şekil **4.16**'da gösterilmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan DT modeli kullanılarak elde edilen hata matrisi, Şekil **4.17**'de gösterilmiştir.



Şekil 4.16. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip DT modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).



Şekil 4.17. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan DT modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

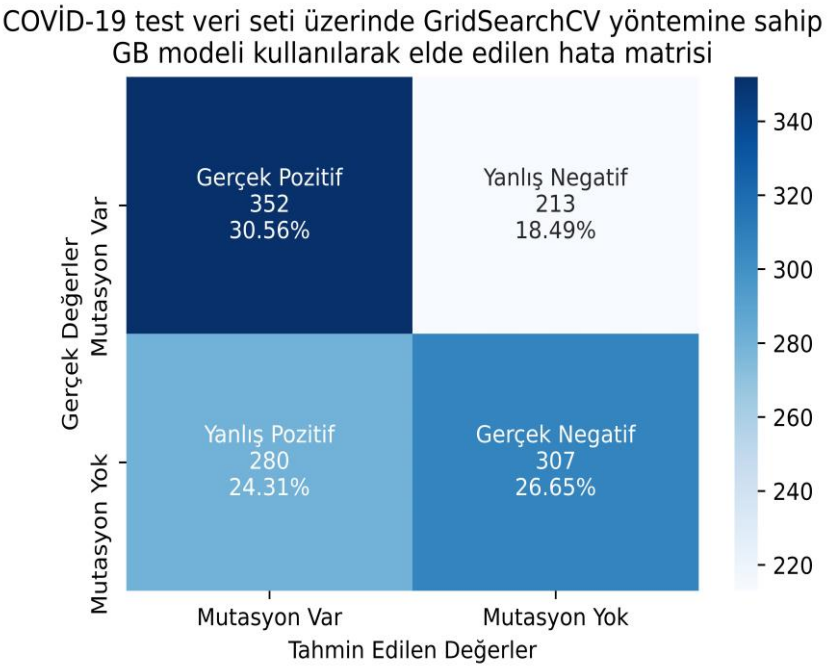
GridSearchCV'li veya GridSearchCV'siz GB algoritmasının performans değerleri, Tablo **4.25**'te gösterilmiştir.

Tablo 4.25. GridSearchCV'li veya GridSearchCV'siz GB algoritmasının performans değerleri (Burukanlı ve Yumuşak, 2024b).

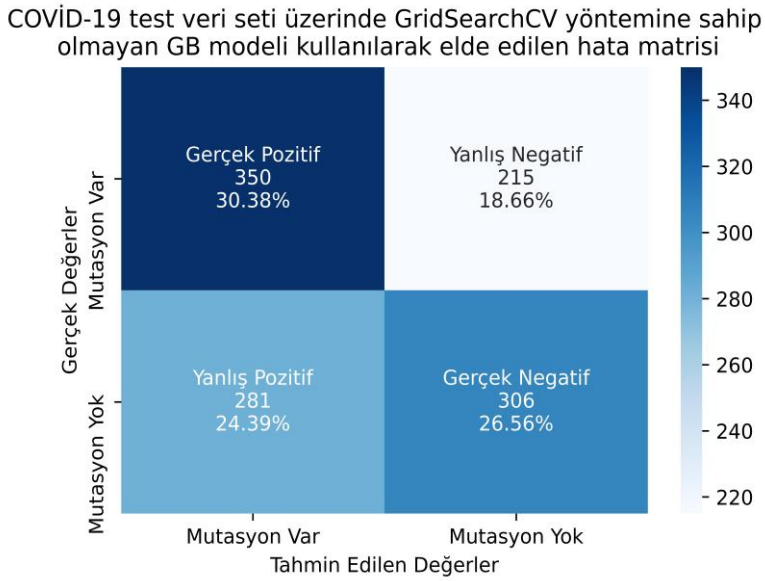
Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
GB	GridSearchCV'li	0.5720	0.5570	0.6230	0.5881	0.1467	0.5781
	GridSearchCV'siz	0.5694	0.5547	0.6195	0.5853	0.1414	0.5840

Tablo **4.25**'te de görüldüğü gibi, GridSearchCV yöntemine sahip GB algoritması, test veri kümesinde GridSearchCV yöntemine sahip olmayan GB algoritmasından daha iyi performans göstermektedir. GridSearchCV yöntemine sahip GB algoritması, doğruluk değerini (0.5694'ten 0.5720'ye), kesinlik değerini (0.5547'den 0.5570'e), hassasiyet değerini (0.6195'ten 0.6230'a), F1-skor değerini (0.5853'ten 0.5881'e) ve MCC değerini (0.1414'ten 0.1467'ye) artırmıştır. Sonuç olarak, GridSearchCV yöntemine sahip GB algoritmasının test veri seti üzerindeki performansı önemli ölçüde arttırdığı gözlemlenmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip GB modeli kullanılarak elde edilen hata matrisi, Şekil **4.18**'de gösterilmiştir.

COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan GB modeli kullanılarak elde edilen hata matrisi, Şekil **4.19**'da gösterilmiştir.



Şekil 4.18. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip GB modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).



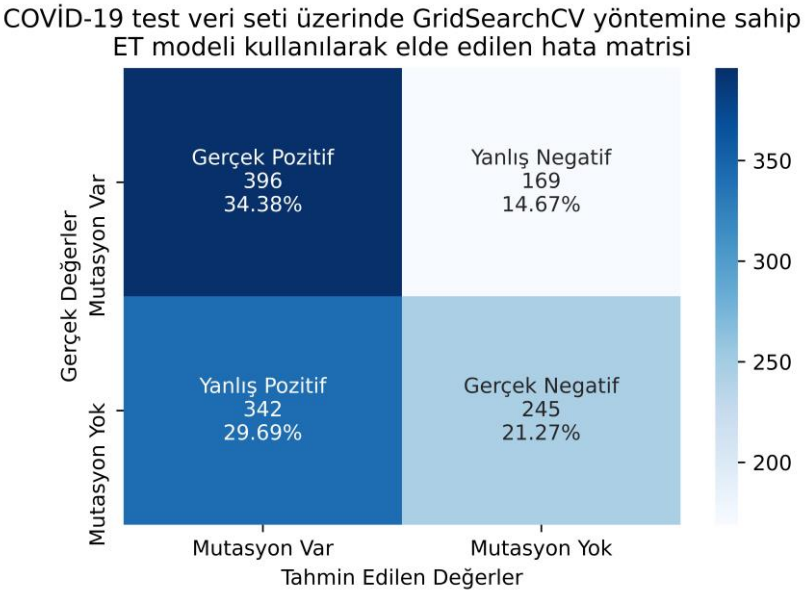
Şekil 4.19. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan GB modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

GridSearchCV'li veya GridSearchCV'siz ET algoritmasının performans değerleri, Tablo **4.26**'da gösterilmiştir.

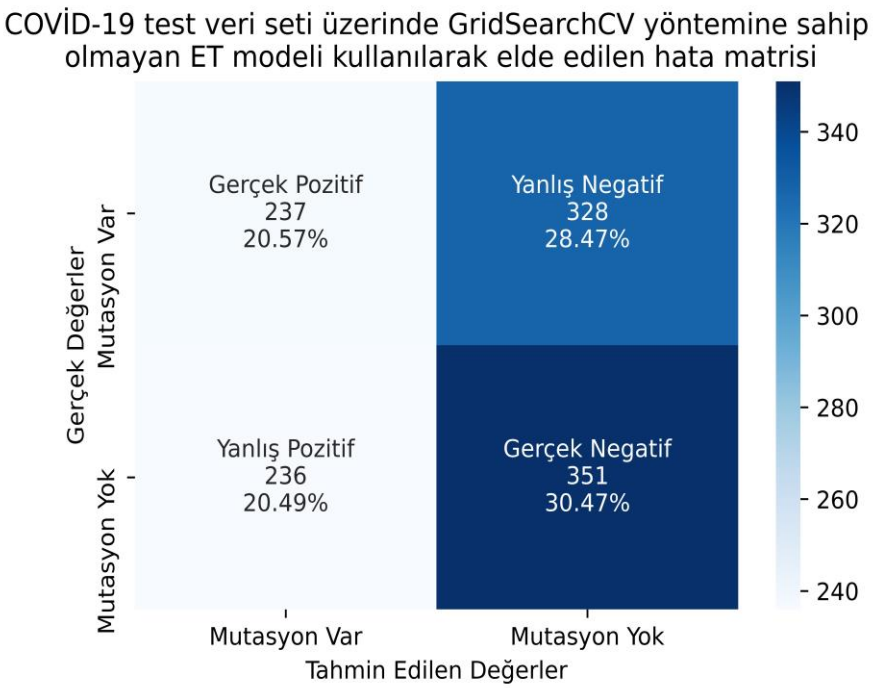
Tablo 4.26. GridSearchCV'li veya GridSearchCV'siz ET algoritmasının performans değerleri (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
ET	GridSearchCV'li	0.5564	0.5366	0.7009	0.6078	0.1232	0.5628
	GridSearchCV'siz	0.5104	0.5011	0.4195	0.4566	0.0177	0.5593

Tablo **4.26**'da da görüldüğü gibi, GridSearchCV yöntemine sahip ET algoritması, test veri kümesinde GridSearchCV yöntemine sahip olmayan ET algoritmasından daha iyi performans göstermektedir. GridSearchCV yöntemine sahip ET algoritması doğruluk değerini (0.5104'ten 0.5564'e), kesinlik değerini (0.5011'den 0.5366'ya), hassasiyet değerini (0.4195'ten 0.7009'a), F1-skor değerini (0.4566'dan 0.6078'e), MCC değerini (0.0177'den 0.1232'ye) ve AUC değerini (0.5593'ten 0.5628'e) arttırmıştır. Sonuç olarak, GridSearchCV yöntemine sahip ET algoritmasının test veri seti üzerindeki performansı önemli ölçüde arttırdığı gözlemlenmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip ET modeli kullanılarak elde edilen hata matrisi, Şekil **4.20**'de gösterilmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan ET modeli kullanılarak elde edilen hata matrisi, Şekil **4.21**'de gösterilmiştir.



Şekil 4.20. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip ET modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).



Şekil 4.21. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan ET modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

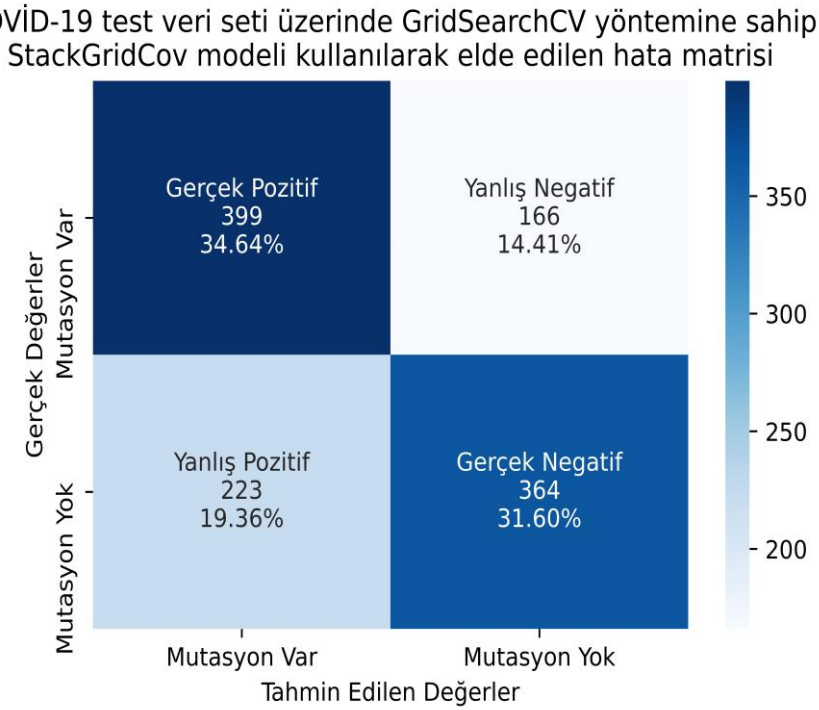
GridSearchCV'li veya GridSearchCV'siz StackGridCov algoritmasının performans değerleri, Tablo **4.27**'de gösterilmiştir.

Tablo 4.27. GridSearchCV'li veya GridSearchCV'siz StackGridCov algoritmasının performans değerleri (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
StackGridCov	GridSearch CV'li	0.6623	0.6415	0.7062	0.6723	0.3273	0.7018
	GridSearch CV'siz	0.6016	0.5833	0.6566	0.6178	0.2063	0.6133

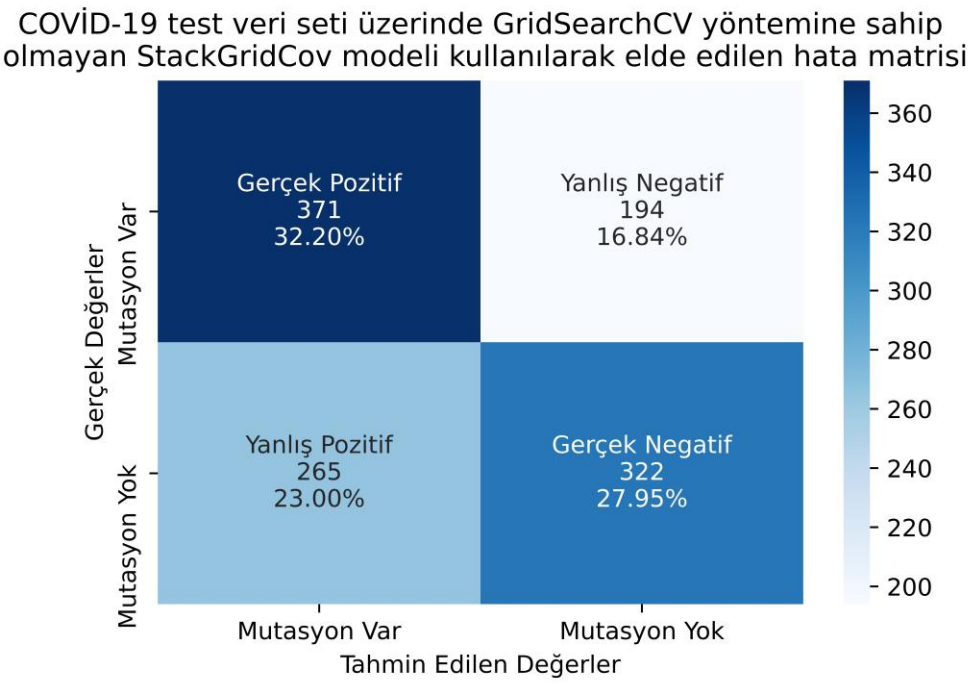
Tablo **4.27**'de de görüldüğü gibi, GridSearchCV tekniğine sahip önerilen StackGridCov algoritması, test veri kümesinde GridSearchCV sahip olmayan StackGridCov algoritmasından daha iyi performans göstermektedir. GridSearchCV yöntemine sahip önerilen StackGridCov algoritması doğruluk değerini (0.6016'dan 0.6623'e), kesinlik değerini (0.5833'ten 0.6415'e), hassasiyet değerini (0.6566'dan

0.7062'ye), F1-skor değerini (0.6178'den 0.6723'e), MCC değerini (0.2063'ten 0.3273'e) ve AUC değerini (0.6133'ten 0.7018'e) yükseltmiştir. Sonuç olarak, GridSearchCV yöntemine sahip önerilen StackGridCov algoritmasının test veri seti üzerindeki performansı önemli ölçüde arttırdığı gözlemlenmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip StackGridCov modeli kullanılarak elde edilen hata matrisi, Şekil **4.22**'de gösterilmiştir.



Şekil 4.22. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip StackGridCov modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

Şekil **4.22**'de de görüldüğü üzere, COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip StackGridCov modeli, "mutasyon" sınıfındaki 565 örnekten 399 örneği doğru tahmin ederken, "mutasyon" sınıfındaki 565 örnekten sadece 166 örneği hatalı tahmin etmiştir. Ek olarak, GridSearchCV yöntemine sahip StackGridCov modeli, "mutasyon yok" sınıftaki 587 örnekten 223 örneneği doğru bir şekilde tahmin ederken, "mutasyon yok" sınıftaki 587 örnekten 364 örneneği hatalı tahmin etmiştir. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan StackGridCov modeli kullanılarak elde edilen hata matrisi, Şekil **4.23**'te gösterilmiştir.



Şekil 4.23. COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan StackGridCov modeli kullanılarak elde edilen hata matrisi (Burukanlı ve Yumuşak, 2024b).

Şekil **4.23**'te de görüldüğü üzere, COVİD-19 test veri kümesi üzerinde GridSearchCV yöntemine sahip olmayan StackGridCov modeli, "mutasyon" sınıfındaki 565 örnekten 371 örneği doğru tahmin ederken, "mutasyon" sınıfındaki 565 örnekten sadece 194 örneği hatalı tahmin etmiştir. Ek olarak, GridSearchCV yöntemine sahip olmayan StackGridCov modeli, "mutasyon yok" sınıftaki 587 örnekten 265 örneneği doğru bir şekilde tahmin ederken, "mutasyon yok" sınıftaki 587 örnekten 322 örneneği hatalı tahmin etmiştir. Derin öğrenme modellerinin (RNN, LSTM, GRU ve Transformer) test veri kümesi üzerindeki performans değerleri, Tablo **4.28**'de gösterilmiştir.

Tablo 4.28. Derin öğrenme modellerinin (RNN, LSTM, GRU ve Transformer) test veri kümesi üzerindeki performans değerleri (Burukanlı ve Yumuşak, 2024b).

Model	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
RNN	0.5889	0.5648	**0.7101**	0.6285	0.1882	**0.5418**
LSTM	0.6218	0.6040	0.6825	**0.6378**	0.2503	0.5177
GRU	0.6327	0.6294	0.6246	0.6254	0.2663	0.4863
Transformer	**0.6395**	**0.6601**	0.5708	0.6067	**0.2837**	0.4605

Tablo **4.28**'de de görüldüğü gibi Transformer modeli, test veri setinde 0.6395 doğruluk değeri, 0.6601 kesinlik değeri ve 0.2837 MCC değeri açısından RNN modeli, LSTM modeli ve GRU modelinden daha iyi performans göstermektedir. RNN modeli, test veri setinde 0.7101 hassasiyet değeri ve 0.5418 AUC değeri açısından Transformer modeli, LSTM modeli ve GRU modelinden daha iyi performans göstermektedir. Ayrıca LSTM modeli, test veri setinde 0.6378 F1-skor değeri açısından Transformer modeli, RNN modeli ve GRU modelinden daha iyi performans göstermektedir. RNN modeli, 0.5889 doğruluk değeri, 0.5648 kesinlik değeri ve 0.1882 MCC değeri açısından diğer derin öğrenme modellerine göre daha kötü performans göstermiştir. Benzer şekilde transformer modeli, 0.5708 hassasiyet değeri, 0.6067 F1-skor değeri ve 0.4605 AUC değeri açısından diğer derin öğrenme modellerine göre daha kötü performans göstermiştir. Önerilen StackGridCov modeli ile diğer modellerin test veri kümesi üzerindeki performans karşılaştırmaları, Tablo **4.29**'da gösterilmiştir.

Tablo 4.29. Önerilen StackGridCov modeli ile diğer modellerin test veri seti üzerindeki performans karşılaştırmaları (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
SVM	GridSearch CV'li	0.6042	0.5835	0.6743	0.6256	0.2128	0.6114
	GridSearch CV'siz	0.5304	0.5188	0.5877	0.5510	0.0633	0.5443
RF	GridSearch CV'li	0.5677	0.5489	0.6655	0.6016	0.1416	0.5725
	GridSearch CV'siz	0.5443	0.5337	0.5611	0.5470	0.0892	0.5609
XGBoost	GridSearch CV'li	0.5686	0.5561	0.5965	0.5756	0.1384	0.5763
	GridSearch CV'siz	0.5538	0.5445	0.5522	0.5483	0.1076	0.5609

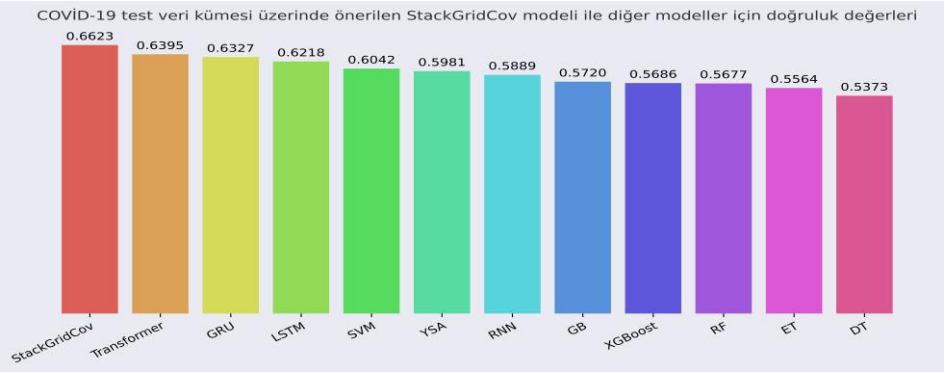
Tablo 4.29. (Devamı) Önerilen StackGridCov modeli ile diğer modelelerin test veri kümesi üzerindeki performans karşılaştırmaları (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
YSA	GridSearch CV'li	0.5981	0.5810	0.6478	0.6126	0.1989	0.6054
	GridSearch CV'siz	0.5686	0.5582	0.5770	0.5675	0.1375	0.5836
DT	GridSearch CV'li	0.5373	0.5254	0.5858	0.5540	0.0768	0.5676
	GridSearch CV'siz	0.5061	0.4957	0.4053	0.4460	0.0085	0.5435
GB	GridSearch CV'li	0.5720	0.5570	0.6230	0.5881	0.1467	0.5781
	GridSearch CV'siz	0.5694	0.5547	0.6195	0.5853	0.1414	0.5840
ET	GridSearch CV'li	0.5564	0.5366	0.7009	0.6078	0.1232	0.5628
	GridSearch CV'siz	0.5104	0.5011	0.4195	0.4566	0.0177	0.5593
RNN	GridSearch CV'siz	0.5889	0.5648	**0.7101**	0.6285	0.1882	0.5418
LSTM	GridSearch CV'siz	0.6218	0.6040	0.6825	0.6378	0.2503	0.5177
GRU	GridSearch CV'siz	0.6327	0.6294	0.6246	0.6254	0.2663	0.4863
Transformer	GridSearch CV'siz	0.6395	**0.6601**	0.5708	0.6067	0.2837	0.4605

Tablo 4.29. (Devamı) Önerilen StackGridCov modeli ile diğer modelelerin test veri kümesi üzerindeki performans karşılaştırmaları (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC	AUC
StackGridCov	GridSearch CV'li	**0.6623**	0.6415	0.7062	**0.6723**	**0.3273**	**0.7018**
	GridSearch CV'siz	0.6016	0.5833	0.6566	0.6178	0.2063	0.6133

Tablo **4.29**'da da görüldüğü gibi, GridSearchCV tekniğine sahip önerilen StackGridCov, test veri setinde 0.6623 doğruluk değeri, 0.6723 F1-skor değeri, 0.3273 MCC değeri ve 0.7018 AUC değeri ile diğer algoritmalardan daha iyi performans gösterimiştir. Transformer modeli, test veri seti üzerinde 0.6601 kesinlik deeğri açısından diğer algoritmalardan daha iyi performans göstermiştir. Öte yandan GridSearchCV yöntemine sahip olmayan DT modeli, test veri setinde diğer algoritmalara göre 0.5061 doğruluk değeri, 0.4957 kesinlik değeri, 0.4053 hassasiyet değeri, 0.4460 F1-skor değeri ve 0.0085 MCC değeri açısından daha kötü performans göstermiştir. Ayrıca transformer modeli, test veri seti üzerinde 0.4605 AUC değeri açısından diğer algoritmalara göre daha kötü performans göstermiştir. Sonuç olarak GridSearchCV hiperparametre tekniğinin kullanılmasının genel olarak önerilen StackGridCov modeli ile diğer yöntemlerin performansını arttırdığı gözlemlenmiştir. Önerilen StackGridCov modeli ile diğer modelerin COVİD-19 test veri kümesindeki doğruluk değerleri, Şekil **4.24**'te gösterilmiştir.



Şekil 4.24. Önerilen StackGridCov modeli ile diğer modelerin COVİD-19 test veri kümesindeki doğruluk değerleri (Burukanlı ve Yumuşak, 2024b).

Önerilen StackGridCov modeli ile diğer modellerin KFold veri seti üzerindeki performans değerlerinin karşılaştırılması, Tablo **4.30**'da gösterilmiştir.

Tablo 4.30. Önerilen StackGridCov modeli ile diğer modellerin KFold veri seti üzerindeki performans değerlerinin karşılaştırılması (Burukanlı ve Yumuşak, 2024b).

Model	Hyper-parametre ayarlama	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC
SVM	GridSearchCV'li	0.5934	0.5811	**0.6721**	0.6232	0.1892
	GridSearchCV'siz	0.5590	0.5526	0.6200	0.5843	0.1190
RF	GridSearchCV'li	0.5818	0.5696	0.6707	0.6159	0.1663
	GridSearchCV'siz	0.5398	0.5379	0.5703	0.5533	0.0798
XGBoost	GridSearchCV'li	0.5573	0.5554	0.5832	0.5684	0.1149
	GridSearchCV'siz	0.5417	0.5414	0.5544	0.5475	0.0835
YSA	GridSearchCV'li	0.5669	0.5644	0.5884	0.5759	0.1340
	GridSearchCV'siz	0.5488	0.5470	0.5679	0.5565	0.0981
DT	GridSearchCV'li	0.5486	0.5401	0.6617	0.5926	0.1013
	GridSearchCV'siz	0.5092	0.5076	0.4456	0.4722	0.0189
GB	GridSearchCV'li	0.5681	0.5638	0.6065	0.5840	0.1367
	GridSearchCV'siz	0.5667	0.5630	0.6020	0.5813	0.1339
ET	GridSearchCV'li	0.5637	0.5550	0.6457	0.5968	0.1292
	GridSearchCV'siz	0.4927	0.4922	0.4161	0.4505	-0.0146
StackGridCov	GridSearchCV'li	**0.6610**	**0.6614**	0.6613	**0.6607**	**0.3226**
	GridSearchCV'siz	0.5969	0.5884	0.6481	0.6165	0.1951

Tablo **4.30**'da da görüldüğü gibi önerilen StackGridCov modeli, KFold veri setinde 0.6610 doğruluk değeri, 0.6614 kesinlik değeri, 0.6607 F1-skor değeri ve 0.3226 MCC değeri ile diğer yaklaşımlardan daha iyi başarım elde etmiştir. Ayrıca SVM modeli,

KFold veri kümesinde 0.6721 hassasiyet değeri açısından diğer algoritmalardan daha iyi performans göstermiştir. Öte yandan, GridSearchCV yöntemine sahip olmayan ET modeli, diğer algoritmalarla karşılaştırıldığında KFold veri setinde 0.4927 doğruluk değeri, 0.4922 kesinlik değeri, 0.4161 hassasiyet değeri, 0.4505 F1-skor değeri ve -0.0146 MCC değeri açısından daha kötü performans göstermiştir. Sonuç olarak, GridSearchCV hiperparametre tekniğinin kullanılmasının genel olarak önerilen StackGridCov modeli ile diğer yöntemlerin performansını arttırdığı gözlemlenmiştir.

4.2.4. Önerilen StackGridCov modeli ile diğer modellerin influenza A/H1N1 HA veri seti üzerindeki performans analizi

Bu bölümde önerilen StackGridCov modelinin ve diğer modellerin performansını değerlendirmek için influenza A/H1N1 HA veri seti kullanılmıştır. Elde edilen sonuçlar incelendiğinde, önerilen StackGridCov modelinin sadece COVİD-19 virüsü üzerindeki mutasyonu tahmin etmekle kalmayıp aynı zamanda influenza A/H1N1 virüsü üzerindeki mutasyonu da tahmin ettiği görülmektedir. Bu da önerilen StackGridCov modelin ne kadar sağlam olduğunu kanıtlamaktadır. Önerilen StackGridCov modeli ile diğer modellerin influenza A/H1N1 HA test veri kümesi üzerindeki performans karşılaştırmaları, Tablo **4.31**'de gösterilmiştir.

Tablo 4.31. Önerilen StackGridCov modeli ile diğer modellerin influenza A/H1N1 HA test veri kümesi üzerindeki performans karşılaştırmaları (Burukanlı ve Yumuşak, 2024b).

Model	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC
SVM	0.9450	0.8269	0.7816	0.8036	0.7721
RF	0.9426	0.8140	0.7790	0.7961	0.7630
XGBoost	0.9444	0.8216	0.7842	0.8025	0.7705
YSA	0.9404	0.8026	0.7773	0.7897	0.7552
DT	0.9373	0.7960	0.7585	0.7768	0.7406
GB	0.9396	**0.8295**	0.7305	0.7768	0.7441
ET	0.9414	0.8138	0.7687	0.7906	0.7570
RNN	0.9354	0.7881	0.7534	0.7696	0.7328

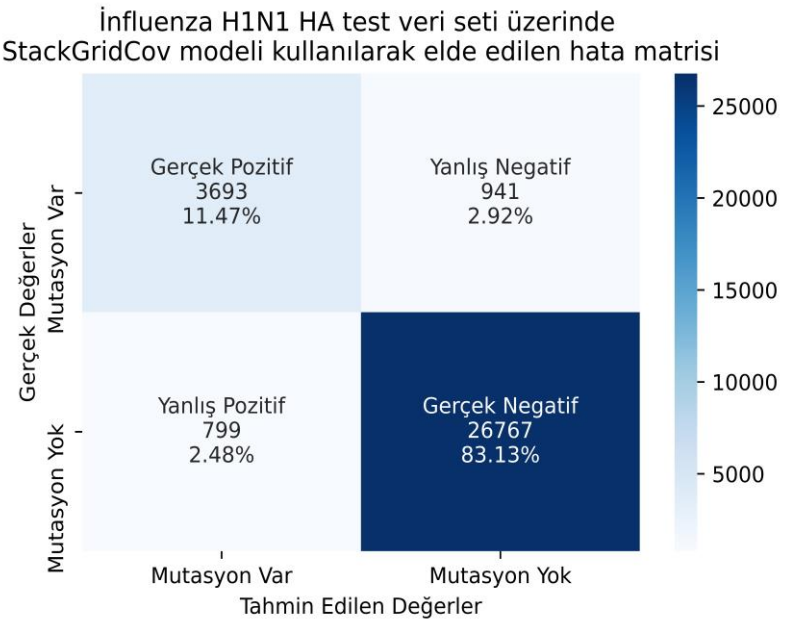
Tablo 4.31. (Devamı) Önerilen StackGridCov modeli ile diğer modellerin influenza A/H1N1 HA test veri kümesi üzerindeki performans karşılaştırmaları (Burukanlı ve Yumuşak, 2024b).

Model	Doğruluk	Kesinlik	Hassasiyet	F1-Skor	MCC
LSTM	0.9418	0.8132	0.7739	0.7928	0.7595
GRU	0.9403	0.8116	0.7613	0.7851	0.7514
Transformer	0.9452	0.8219	0.7903	0.8058	0.7741
StackGridCov	0.**9460**	0.8221	**0.7969**	0.**8093**	0.**7780**

Tablo **4.31**'de de görüldüğü gibi önerilen StackGridCov modeli, influenza A/H1N1 HA test veri setinde 0.9460 doğruluk değeri, 0.7969 hassasiyet değeri, 0.8093 F1-skor değeri ve 0.7780 MCC değeri açısından diğer modellerden daha iyi performans göstermiştir. Ayrıca, GB modeli, influenza A/H1N1 HA test veri setinde 0.8295 kesinlik değeri açısından diğer algoritmalardan daha iyi performans göstermiştir. Öte yandan RNN modeli, diğer algoritmalarla karşılaştırıldığında influenza A/H1N1 HA test veri setinde 0.9354 doğruluk değeri, 0.7881 kesinlik değeri, 0.7696 F1-skor deeğri ve 0.7328 MCC değeri açısından daha kötü performans göstermiştir. Benzer şekilde GB modeli, diğer algoritmalarla karşılaştırıldığında influenza A/H1N1 HA test veri setinde 0.7305 hassasiyet değeri açısından daha kötü performans göstermiştir. Sonuç olarak, önerilen StackGridCov modelinin, hem COVİD-19 test veri kümesinde hem de İnfluenza A/H1N1 HA test veri kümesinde mutasyon tahmini görevinde genel olarak diğer modellerden daha iyi performans elde etmiştir. Önerilen StackGridCov modeli ile diğer modellerin influenza A/H1N1 HA test veri kümesi üzerindeki doğruluk değerleri, Şekil **4.25**'te gösterilmiştir.

Şekil 4.25. Önerilen StackGridCov modeli ile diğer modellerin influenza A/H1N1 HA test veri kümesi üzerindeki doğruluk değerleri (Burukanlı ve Yumuşak, 2024b).

İnfluenza A/H1N1 HA test veri kümesi üzerinde elde edilen önerilen StackGridCov modelinin hata matrisi, Şekil **4.26**'da gösterilmiştir.



Şekil 4.26. İnfluenza A/H1N1 HA test veri kümesi üzerinde elde edilen önerilen StackGridCov modelinin hata matrisi (Burukanlı ve Yumuşak, 2024b).

Önerilen StackGridCov modelinin literatürle karşılaştırılması, Tablo **4.32**'de gösterilmiştir.

Tablo 4.32. Önerilen StackGridCov modelinin literatürle karşılaştırılması (Burukanlı ve Yumuşak, 2024b).

Araştırma makalesi	Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
Zhou ve ark. (2023a)	TEMPO	65.5	**65.8**	61.4	63.6	30.9
Bizim model	GridSearchCV'li StackGridCov	**66.23**	64.15	**70.62**	**67.23**	**32.73**

Tablo **4.32**'de de görüldüğü gibi, GridSearchCV yöntemine sahip önerilen StackGridCov modeli, test veri kümesinde %66.23 doğruluk değeri, %70.62 hassasiyet değeri, %67.23 F1-skor değeri ve %32.73 MCC değeri ile TEMPO modelinden daha iyi performans göstermiştir. Öte yandan, TEMPO modeli, test veri seti üzerinde yalnızca %65.8 kesinlik değeri açısından önerilen StackGridCov

modelinden daha iyi performans göstermiştir. Sonuç olarak, test veri seti üzerinde GridSearchCV modeline sahip önerilen StackGridCov modeli, TEMPO modeli ile karşılaştırıldığında, performansın %1.1 oranında doğruluk değeri, %15 oranında hassasiyet değeri, %5.7 oranında F1-skor değeri ve %5.9 oranında MCC değerini arttığı görülmüştür. Sonuç olarak önerilen StackGridCov modeli, test veri setinde (%65.8 kesinlik değeri hariç) son teknoloji çalışma TEMPO modelinden daha iyi performans göstermiştir. Nihai olarak, GridSearchCV hiperparametre ayarlama algoritmasının hem önerilen StackGridCov modelinin hemde diğer modellerin performansını önemli ölçüde iyileştirdiği ve GridSearchCV algoritmasına sahip önerilen StackGridCov modelinin, hem COVİD-19 virüs veri seti üzerindeki mutasyonları tahmin etmede hemde influenza A/H1N1 HA virüsü veri kümesi üzerindeki mutasyonları tahmin etmede oldukça başarılı olduğu gözlemlenmiştir.

4.3. Önerilen HyperAttCov Modeli İçin Elde Edilen Bulgular

Bu tez çalışmasında, derin öğrenme modellerinin (RNN, LSTM vb.) performansını en üst düzeye çıkarmak amacıyla hiperparametre değerleri (hidden size, dropout, batch size vb.) birçok kez (deneme yanılma) test edilmiş ve en iyi hiper parametre değerleri seçilmiştir. Tüm modeller için (SVM ve LR modelleri hariç), model optimizasyonu için minimum batch size değeri 32 olan Adam optimizasyon algoritması kullanılmıştır. Önerilen HyperAttCov modeli ile diğer modellerin kodlayıcısında (encoder'da) öğrenme oranı 0,0015 ve hidden size değeri 128 olarak ayarlanmıştır. Amaç fonksiyonu olarak (kayıpları en aza indirmek için) çapraz entropi (cross entropy) kullanılmıştır. Tüm modellerin eğitimi için dropout değeri 0.4 ve epok değeri 120 tercih edilmiştir. Ayrıca transformer kodlayıcı katmanında kullanılan çoklu kafa dikkat sayısı (multi head attention) (varsayılan değerleri dahil) birçok kez test edilmiş ve en iyi hiper parametre değeri olan çoklu kafa dikkat = 2 olarak seçilmiştir. Önerilen HyperAttCov modeli için (varsayılan değerler dahil) birçok kez test edilmiş ve en iyi hiperparametre değeri num_heads sayısı = 2 olarak seçilmiştir. Önerilen HyperAttCov modelinin ve diğer modellerin hiperparametre ve bu hiperparametrelerin değerleri, Tablo **4.33**'te verilmiştir.

Tablo 4.33. Önerilen HyperAttCov modelinin ve diğer modellerin hiperparametre ve değerleri (Burukanlı ve Yumuşak, 2024c).

Hyper-parametre adı	Değeri
Hidden Size	128
Dropout	0.4
Batch Size	32
Öğrenme oranı	0.0015
Epok	120
Optimizasyon algoritması	Adam
Kayıp fonksiyonu	cross entropy
Önerilen HyperAttCov için num_heads sayısı	2
HyperMixer için num_heads sayısı	1
Transformer kodlayıcı sayısı	1
Çoklu kafalı dikkat sayısı	2

4.3.1. Elde edilen bulgular

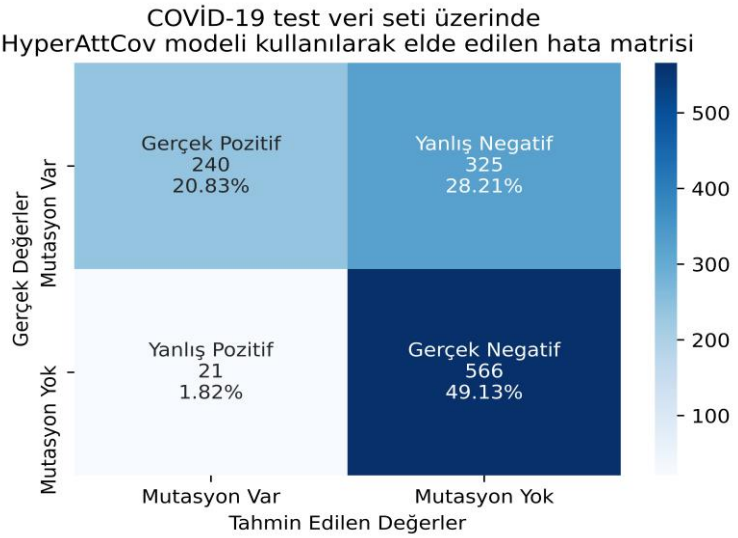
Bu tez çalışmasında önerilen HyperAttCov modeli ile diğer modellerin COVİD-19 test veri seti üzerinde (holdout tekniği kullanılarak) test edilerek performans değerleri elde edilmiştir. Holdout tekniği kullanılması durumunda veri seti düzgün dağılmış olabilme ihtimaline karşın, önerilen HyperAttCov modelinin performansı ile diğer modellerin performanslarını adil bir şekilde değerlendirmek için, tüm deneysel sonuçlar farklı rastgele tohumlara sahip 10 rastgele denemenin ortalaması alınarak hesaplanmıştır. Ayrıca tüm derin öğrenme modellerinin performansı hesaplanırken en iyi doğruluk değerleri kullanılarak tüm derin öğrenme modellerinin performans değerleri elde edilmiştir. Önerilen HyperAttCov modeli ile diğer modellerin COVİD-19 test veri seti üzerinde elde edilen performans değerleri, Tablo **4.34**'te gösterilmiştir.

Tablo 4.34. Önerilen HyperAttCov modeli ile diğer modellerin COVİD-19 test veri seti üzerinde elde edilen performans değerleri (Burukanlı ve Yumuşak, 2024c).

Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
SVM	53.0	51.9	**58.8**	55.1	6.3
LR	54.2	53.0	57.5	55.2	8.5
RF	54.4	53.4	56.1	54.7	8.9
RNN	63.5	65.9	53.1	58.8	27.3
LSTM	63.9	64.8	57.9	**61.1**	27.8
GRU	66.3	73.1	49.6	59.1	34.0
Transformer	64.8	73.7	44.1	55.1	31.7
HyperMixer (Mai et al., 2023)	60.1	59.5	58.4	58.9	20.1
HyperAttCov	**70.0**	**92.0**	42.5	58.1	**46.5**

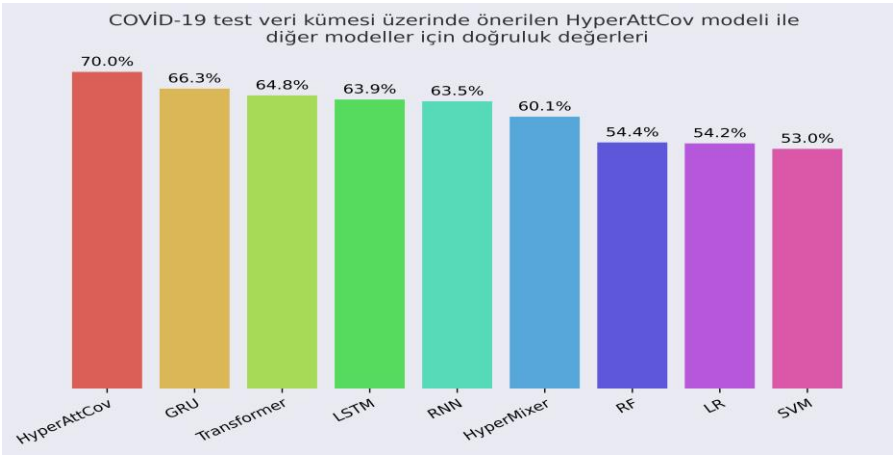
Tablo **4.34**'te de görüldüğü üzere önerilen HyperAttCov modeli, COVİD-19 S protein testi veri seti üzerinde %70.0 ile doğruluk değeri, %92.0 ile kesinlik değeri ve %46,5 ile MCC değeri açısından diğer modellere göre daha iyi sonuçlar elde etmiştir. Öte yandan, SVM modeli, COVİD-19 test veri setinde %58.8 ile hassasiyet değeri bakımından diğer yaklaşımlarından göre daha iyi performans elde etmiştir. Ek olarak, LSTM modeli, COVİD-19 test veri setinde %61.1 ile F1-Skor değeri ile diğer modellere göre daha iyi performans göstermiştir. Sonuç olarak deneysel sonuçlar detaylı olarak incelendiğinde derin öğrenmeye dayalı modellerin genellikle diğer geleneksel makine öğrenimi modellerinden daha iyi performans elde etmiştir. Önerilen HyperAttCov modeli, COVİD-19 test veri seti üzerinde standart HyperMixer (Mai et al., 2023) modeliyle karşılaştırıldığında performans değerlerinde doğrulukta %16.47, kesinlikte %54.62 ve MCC'de %131.34 artış olduğu görülmüştür. Ayrıca önerilen HyperAttCov modeli, COVİD-19 test veri seti üzerinde standart LSTM modeli ile karşılaştırıldığında, performans değerlerinin ortalaması doğrulukta %9.55, kesinlikte %41.98 ve MCC değerinde %67.27 artış olduğu gözlemlenmiştir. Önerilen

HyperAttCov modelinin COVİD-19 test veri seti üzerinde elde edilen hata matrisi, Şekil **4.27**'de gösterilmiştir.



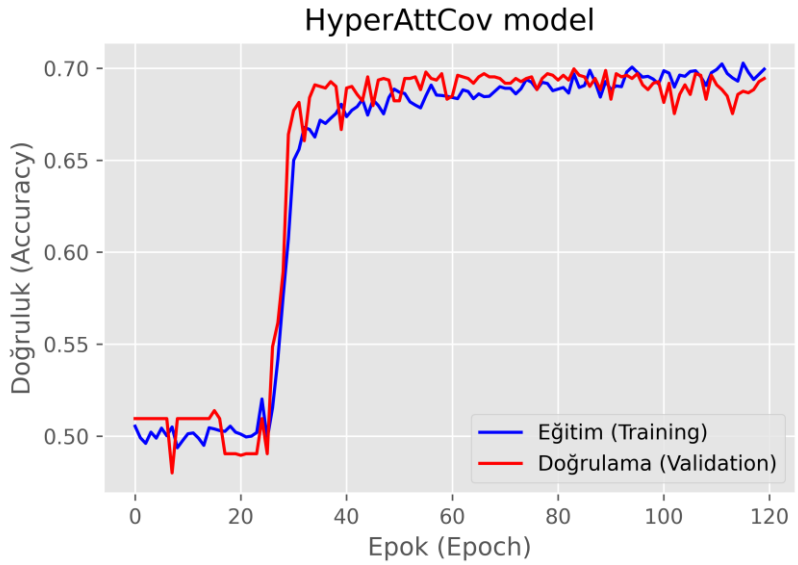
Şekil 4.27. Önerilen HyperAttCov modelinin COVİD-19 test veri seti üzerinde elde edilen hata matrisi

Şekil **4.27**'de de görüldüğü üzere, COVİD-19 test veri setinde önerilen HyperAttCov modeli, "mutasyon" sınıfındaki 565 örnekten 240 örneği doğru tahmin ederken, "mutasyon" sınıfındaki 565 örnekten 325 örneği hatalı tahmin etmiştir. Ayrıca önerilen HyperAttCov modeli, "mutasyon yok" sınıfta ise 587 örnekten 566 örneği doğru tahmin ederken, "mutasyonun yok" sınıfından 587 örnekten sadece 21 örneği hatalı tahmin etmiştir. Önerilen HyperAttCov modeli ile diğer modellerin COVİD-19 test veri kümesi üzerindeki doğruluk değerleri, Şekil **4.28**'de gösterilmiştir.

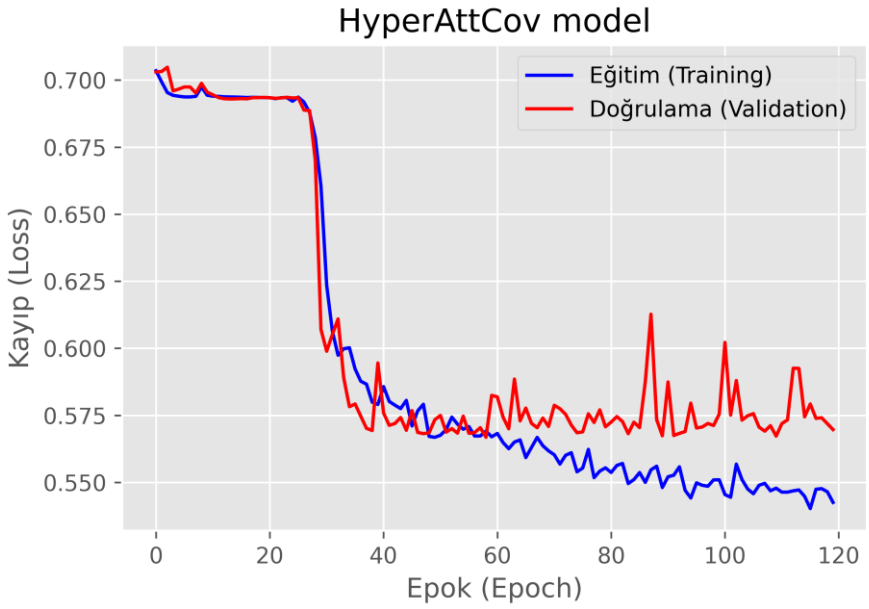


Şekil 4.28. Önerilen HyperAttCov modeli ile diğer modellerin COVİD-19 test veri kümesi üzerindeki doğruluk değerleri (Burukanlı ve Yumuşak, 2024c).

COVİD-19 test veri seti üzerinde önerilen HyperAttCov modeli için elde edilen doğruluk-epok eğrisi, Şekil **4.29**'de gösterilmiştir. COVİD-19 test veri setinde önerilen HyperAttCov modeli için elde edilen kayıp-epok eğrisi, Şekil **4.30**'da gösterilmiştir.



Şekil 4.29. COVİD-19 test veri seti üzerinde önerilen HyperAttCov modeli için elde edilen doğruluk-epok eğrisi (Burukanlı ve Yumuşak, 2024c).



Şekil 4.30. COVİD-19 test veri setinde önerilen HyperAttCov modeli için elde edilen kayıp-epok eğrisi (Burukanlı ve Yumuşak, 2024c).

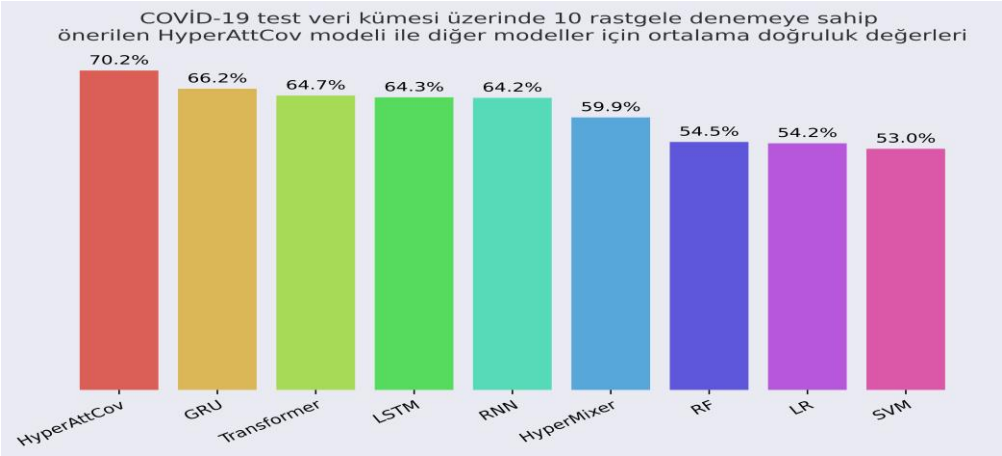
Önerilen HyperAttCov modeli ile diğer modeller için COVİD-19 test veri kümesinde farklı rastgele tohumlara sahip 10 rastgele deneme ile elde edilen ortalama performans değerleri, Tablo **4.35**'te gösterilmiştir.

Tablo 4.35. Önerilen HyperAttCov modeli ile diğer modeller için COVİD-19 test veri kümesinde farklı rastgele tohumlara sahip 10 rastgele deneme ile elde edilen ortalama performans değerleri (Burukanlı ve Yumuşak, 2024c).

Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
SVM	53.0	51.9	58.8	55.1	6.3
LR	54.2	53.0	57.5	55.2	8.5
RF	54.5	53.4	56.1	54.7	9.0
RNN	64.2	67.8	51.9	58.7	29.0
LSTM	64.3	65.2	58.3	**61.6**	28.5
GRU	66.2	73.4	48.8	58.6	33.9
Transformer	64.7	70.4	49.3	57.6	30.8
HyperMixer (Mai et al., 2023)	59.9	59.0	**60.4**	60.0	19.9
HyperAttCov	**70.2**	**90.4**	43.9	59.1	**46.2**

Tablo **4.35**'te de görüldüğü gibi önerilen HyperAttCov modeli, COVİD-19 S protein testi veri setinde %70.2 ile ortalama doğruluk değeri, %90.4 ile ortalama kesinlik değerive %46.2 ile ortalama MCC değeri açısından diğer modellere göre daha iyi sonuçlar elde etmiştir. Öte yandan, HyperMixer (Mai et al., 2023) modeli COVİD-19 test veri setinde %60.4 ile ortalama hassasiyet değeri ile diğer modellere göre daha iyi performans göstermiştir. Ayrıca, LSTM modeli, COVİD-19 test veri setinde %61.6 ile ortalama F1-skor değeri ile diğer modellere göre daha iyi performans göstermiştir. Öte yandan, SVM modeli, ortalama doğruluk, ortalama hassasiyet ve ortalama MCC değerleri bazında en kötü performansı göstermiştir. Deneysel sonuçlar, derin öğrenmeye dayalı modellerin genellikle diğer geleneksel makine öğrenimi modellerinden daha iyi performans elde ettiğini göstermiştir. Önerilen HyperAttCov modeli, COVİD-19 test veri seti üzerinde standart HyperMixer [86] modeli ile

karşılaştırıldığında, performans değerlerinin ortalama olarak %17.20 ile doğruluk, %53.22 ile kesinlik ve %132.16 ile MCC artış görülmüştür. Ayrıca önerilen HyperAttCov modeli, COVİD-19 test veri seti üzerinde standart LSTM modeli ile karşılaştırıldığında, performans değerlerinin ortalama olarak %9.18 ile doğruluk, %38.65 ile kesinlik ve %62.11 ile MCC artışı göstermiştir. Önerilen HyperAttCov modeli ile diğer modellerin COVİD-19 test veri kümesi üzerinde ortalama 10 rastgele deneme için elde edilen doğruluk değerleri, Şekil **4.31**'de gösterilmiştir.



Şekil 4.31. Önerilen HyperAttCov modeli ile diğer modellerin COVİD-19 test veri kümesi üzerinde 10 rastgele deneme için elde edilen ortalama doğruluk değerleri (Burukanlı ve Yumuşak, 2024c).

Önerilen HyperAttCov modelinin COVİD-19 test veri seti üzerine literatürdeki (TEMPO) çalışma ile performans karşılaştırması, Tablo **4.36**'da gösterilmiştir.

Tablo 4.36. Önerilen HyperAttCov modelinin COVİD-19 test veri seti üzerine literatürdeki (TEMPO) çalışma ile performans karşılaştırması (Burukanlı ve Yumuşak, 2024c).

Araştırma makalesi	Model	Doğruluk (%)	Kesinlik (%)	Hassasiyet (%)	F1-Skor (%)	MCC (%)
Zhou ve ark. (2023a)	TEMPO	65.5	65.8	61.4	63.6	30.9
Bizim model	**HyperAttCov**	**70.0**	**92.0**	42.5	58.1	**46.5**
Bizim model (10 rastgele denemenin ortalaması)	**HyperAttCov**	**70.2**	**90.4**	43.9	59.1	**46.2**

Tablo **4.36**'da da görüldüğü gibi önerilen HyperAttCov modeli, COVİD-19 test veri setinde doğruluk, kesinlik ve MCC değeri açısından TEMPO modelinden daha iyi performans göstermiştir. Deneysel sonuçlar göstermiştir ki, önerilen HyperAttCov modeli TEMPO modeli ile karşılaştırıldığında, COVİD-19 test veri setinde performans değerleri %6.87 oranında doğruluk değeri, %39.82 oranında kesinlik değeri, %50.49 oranında MCC değeri artışı gözlemlenmiştir. Benzer şekilde, 10 rastgele denemenin ortalaması alınması durumda ise önerilen HyperAttCov modeli, TEMPO modeliyle karşılaştırıldığında, COVİD-19 test veri setinde performans değerlerinin ortalaması olarak %7.18 oranında doğruluk değeri, %37.39 oranında kesinlik değeri ve %49.51 oranında ise MCC değeri artırılmıştır. Sonuç olarak, HyperMixer ve dikkat mekanizmalarının kullanılması, önerilen HyperAttCov modelinin performansını önemli ölçüde arrtırmıştır.

5. SONUÇ VE ÖNERİLER

COVİD-19 virüsünün mutasyonlarıyla baş etmede aşı ve ilaçların geliştirilmesi oldukça önemlidir. Günümüzde pek çok aşı gelişrilmiş olup, COVİD-19 virüsünün mutasyona sık sık mutasyona uğraması sonucu etkindikleri önemli oranda azalmıştır. Gelişen teknoloji ile beraber başarısı kanıtlanmış yapay zeka tabanlı modeler oldukça sık sağlık alanına uygulanmıştır. Bu tez çalışmasında, COVİD-19 virüsünün yapısında meydana gelebilecek mutasyonları tahmin etmek için üç adet yapay zeka tabanlı model (TfrAdmCov, StackGridCov ve HyperAttCov) önerdik.

Önerilen TfrAdmCov modelini kullanarak 2022 yılında COVİD-19 S (Spike) proteininde meydana gelebilecek mutasyonları tahmin etmeyi amaçladık. Veri kümelerini oluşturmak için aglomeratif kümeleme algoritmasını kullandık. Ayrıca makine öğrenimi tabanlı modellerin performansını artırmak için GridSearchCV hiperparametre ayarlama yöntemini kullandık. Her makine öğrenimi tabanlı modelin performansını değerlendirmek için holdout tekniği ve stratified 10 katlı çapraz doğrulama tekniği kullanılmıştır. Önerilen TfrAdmCov modelinin ve derin öğrenmeye dayalı modellerin performansını doğrulamak için istatistiksel analizler de gerçekleştirdik. Deneysel sonuçlar incelendiğinde, önerilen TfrAdmCov modelinin hem temel hem de bazı son teknoloji yöntemlerden daha iyi performans gösterdiği görülmüştür. Önerilen TfrAdmCov modeli, COVİD-19 test veri kümesinde %99.93 ile doğruluk değerine, %100.00 kesinlik değerine, %97.38 hassasiyet değerine, %98.67 F1-skor değerine ve %9.65 MCC değerine ulaşmıştır. Benzer şekilde, önerilen TfrAdmCov modeli, influenza A/H3N2 HA protein veri seti üzerinde %96.33 ile doğruluk değeri, %81.55 ile kesinlik değeri, %52.33 ile hassasiyet değeri, %63.75 ile F1-skor değeri ve %63.61 ile MCC değeri açısından diğer modellerden daha iyi sonuçlar elde etmiştir. Sonuç olarak, önerilen TfrAdmCov modeli, hem COVİD-19 veri kümesinde hem de influenza A/H3N2 HA veri kümesinde meydana gelen mutasyonları başarılı bir şekilde tahmin etmiştir.

Ayrıca, bu tez çalışmasında, COVİD-19 virüsünün mutasyon tahmini için sağlam bir StackGridCov modeli önerdik. Önerilen StackGridCov modelinin ve diğer

algoritmaların performansını artırmak için GridSearchCV hiperparametre ayarlama algoritmasını kullandık. Önerilen StackGridCov modelinin ve diğer algoritmaların performansını değerlendirmek için, holdout tekniğinin yanı sıra stratified 10-katlı çapraz doğrulama tekniğini de kullandık. Önerilen StackGridCov modeli ile diğer algoritmaları doğruluk, kesinlik, hassasiyet, F1-skor, MCC ve AUC değerleri açısından karşılaştırdık. Deneysel sonuçlar incelendiğinde, önerilen StackGridCov modelinin test veri seti üzerinde 0.6623 ile doğruluk değeri, 0.6723 ile F1-skor değeri, 0.3273 ile MCC değeri ve 0.7018 ile AUC değeri ile diğer algoritmalardan daha iyi performans gösterdiği görülmüştür. Ayrıca önerilen StackGridCov modeli, KFold veri setinde 0.6610 ile doğruluk değeri, 0.6614 ile hassasiyet değeri, 0.6607 ile F1-skor değeri ve 0.3226 ile MCC değeri açısından diğer algoritmalardan daha iyi performans göstermiştir. Sonuçlar detaylı incelendiğinde, önerilen StackGridCov modelinin literatürdeki çalışmaya (TEMPO) göre daha iyi performans gösterdiği görülmektedir. Ek olarak, önerilen StackGridCov modelinin ve diğer modellerin performansını değerlendirmek için daha önce ortaya çıkan influenza A/H1N1 HA virüsü veri seti üzerinde mutasyon tahmini gerçekleştirlmiştir. Önerilen StackGridCov modeli, influenza A/H1N1 HA test veri setinde 0.9460 ile doğruluk değeri, 0.7969 ile hassasiyet değeri, 0.8093 ile F1-skor değeri ve 0.7780 ile MCC değeri ile diğer algoritmalardan daha iyi performans göstermiştir. Sonuç olarak, önerilen StackGridCov modeli, hem COVİD-19 virüsü veri kümelerindeki hem de influenza A/H1N1 HA virüsü veri kümesindeki mutasyonları tahmin etmede oldukça başarılı olmuştur.

Ek olarak, bu tez çalışmasında, COVİD-19 mutasyon tahmini için HyperAttCov modelini önerdik. Önerilen HyperAttCov modeli, birçok yöntemden daha iyi performans göstermiştir. Deneysel sonuçlar, önerilen HyperAttCov modelinin, COVİD-19 test veri setinde %70.0 doğruluk değerine, %92.0 kesinlik değerine ve %46.5 MCC değerine ulaştığını göstermiştir. Benzer şekilde önerilen HyperAttCov modeli, ortalama 10 rastgele deneme ile COVİD-19 test veri setinde %70.2 doğruluk değerine, %90.4 hassasiyet değerine ve %46.2 MCC değerine ulaşmıştır. Ayrıca deneysel sonuçlar, önerilen HyperAttCov modelinin TEMPO modeliyle karşılaştırıldığında, performans değerlerinin COVİD-19 test veri setinde %6.87 doğruluk, %39.82 kesinlik ve MCC %50.49 oranında arttığını göstermiştir. Benzer şekilde önerilen 10 rastgele deneme ile HyperAttCov modeli TEMPO modeliyle

karşılaştırıldığında, performans değerlerinin ortalaması, COVİD-19 test veri setinde doğruluk %7.18, kesinlik %37.39, MCC %49.51 oranında arttırmıştır. Elde edilen bu sonuçlar, önerilen HyperAttCov modelinin COVİD-19 mutasyon tahmini açısından oldukça başarılı olduğunu göstermiştir. Bu sonuçlar COVİD-19 virüsüyle mücadele açısından umut vericidir.

COVİD-19 virüsüne yönelik aşı ve ilaç geliştirilmesinde yardımcı olmak veya fikir vermek oldukça önemlidir. Sonuç olarak önerilen modellerin COVİD-19 S proteininde meydana gelecek mutasyonları başarılı bir şekilde tahmin edebildiğini gözlemledik. GridSearchCV algoritması ile her makine öğrenme algoritmasının sadece 3 hiperparametresi üzerinden en uygun hiperparametre değerlerini elde ettik. Önerilen StackGridCov'un tüm hiperparametreleri ve her makine öğrenimi algoritması üzerinde parametre ayarlaması yapmadık çünkü bu çok zaman alıcıydı. Bu nedenle, gelecekteki çalışmada önerilen StackGridCov modelinin performansını daha da artırmak için öncelikle bu tez çalışmasında test edilmeyen diğer hiperparametreler araştırılabilir. İkinci olarak 2023 ve 2024 yılları için yeni COVİD-19 veri seti üzerinde önerilen StackGridCov modelini kullanarak COVİD-19 virüsü üzerinde mutasyon tahmini yapilabilir. Ayrıca, önerilen HyperAttCov modelinin genel performansının iyileştirilmesine ve hassasiyet probleminin çözülmesine odaklanılabilir. Ek olarak, önerilen TfrAdmCov, StackGridCov ve HyperAttCov modelleri, daha önce ortaya çıkmış ve yeni ortaya çıkan virüsler üzerinde mutasyon tahmini gerçekleştirilebilir. Önerilen bu üç model kullanılarak, COVİD-19, infulenza vb. virüslerin daha büyük veri setleri üzerinde mutasyona tahmini yapılabilir. Son olarak önerilen TfrAdmCov, StackGridCov ve HyperAttCov modelleri kullanılarak, COVİD-19 virüsünün diğer proteinleri (M proteini, N proteini vb.) üzerinde mutasyon tahminini gerçekleştirilebilir.

KAYNAKLAR

Abbas, M. E., Chengzhang, Z., Fathalla, A., & Xiao, Y. (2022). End-to-end antigenic variant generation for H1N1 influenza HA protein using sequence to sequence models. *PLoS ONE*, *17*(3), e0266198. https://doi.org/10.1371/journal.pone.0266198

Adaboost, M., Zhu, J., Zou, H., Rosset, S., & Hastie, T. (2009). Multi-class AdaBoost ∗. *Statistics and Its Interface*, *2*(3), 349–360.

Agatonovic-Kustrin, S., & Beresford, R. (2000). Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *Journal of Pharmaceutical and Biomedical Analysis*, *22*(5), 717–727. https://doi.org/10.1016/S0731-7085(99)00272-1

Ahmed, I., & Jeon, G. (2021). Enabling Artificial Intelligence for Genome Sequence Analysis of COVID - 19 and Alike Viruses. *Interdisciplinary Sciences: Computational Life Sciences*, *0123456789*.

Ahmed, I., & Jeon, G. (2022). Enabling Artificial Intelligence for Genome Sequence Analysis of COVID-19 and Alike Viruses. *Interdisciplinary Sciences: Computational Life Sciences*, *14*(2), 504–519. https://doi.org/10.1007/s12539-021-00465-0

Anonim. (2023a, 17 Kasım). *clustalw*. https://www.genome.jp/tools-bin/clustalw

Anonim. (2023b, 15 Kasım). *COVID-19 S protein dataset*. https://www.ncbi.nlm.nih.gov/datasets/taxonomy/2697049/

Asgari, E., & Mofrad, M. R. K. (2015). Continuous distributed representation of biological sequences for deep proteomics and genomics. *PLoS ONE*, *10*(11), 1–16. https://doi.org/10.1371/journal.pone.0141287

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer Normalization. *ArXiv Preprint ArXiv:1607.06450*. http://arxiv.org/abs/1607.06450

Banerjee, K., Vishak Prasad, C., Gupta, R. R., Vyas, K., Anushree, H., & Mishra, B. (2020). Exploring alternatives to softmax function. *ArXiv Preprint ArXiv:2011.11538*. https://doi.org/10.5220/0010502000810086

Barnes, C. O., West, A. P., Huey-Tubman, K. E., Hoffmann, M. A. G., Sharaf, N. G., Hoffman, P. R., Koranda, N., Gristick, H. B., Gaebler, C., Muecksch, F., Lorenzi, J. C. C., Finkin, S., Hägglöf, T., Hurley, A., Millard, K. G., Weisblum, Y., Schmidt, F., Hatziioannou, T., Bieniasz, P. D., … Bjorkman, P. J. (2020). Structures of Human Antibodies Bound to SARS-CoV-2 Spike Reveal Common Epitopes and Recurrent Features of Antibodies. *Cell*, *182*(4), 828-842. https://doi.org/10.1016/j.cell.2020.06.025

Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32.

Cai, C., Li, J., Xia, Y., & Li, W. (2024). FluPMT: Prediction of Predominant Strains of Influenza A Viruses Via Multi-task Learning. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1–11. https://doi.org/10.1109/TCBB.2024.3378468

Chakraborty, G. S., Singh, D., Rakhra, M., Batra, S., & Singh, A. (2022). Covid-19 and Diabetes Risk Prediction for Diabetic Patient using Advance Machine Learning Techniques and Fuzzy Inference System. *Proceedings of 5th International Conference on Contemporary Computing and Informatics, IC3I 2022*, *2022*, 1212–1219. https://doi.org/10.1109/IC3I56241.2022.10073256

Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, *21*(1), 1–13. https://doi.org/10.1186/s12864-019-6413-7

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *ArXiv:1412.3555*, 1–9. http://arxiv.org/abs/1412.3555

Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, *20*, 273–297.

Cui, J., Li, F., & Shi, Z. L. (2019). Origin and evolution of pathogenic coronaviruses. *Nature Reviews Microbiology*, *17*(3), 181–192. https://doi.org/10.1038/s41579-018-0118-9

de Wit, J. J., & Cook, J. K. A. (2020). Spotlight on avian coronaviruses. *Avian Pathology*, 313–316. https://doi.org/10.1080/03079457.2020.1761010

Dietterich, T. G. (2002). Ensemble Learning. *The Handbook of Brain Theory and Neural Networks*, *2*(1), 110–125.

Divina, F., Gilson, A., Goméz-Vela, F., Torres, M. G., & Torres, J. F. (2018). Stacking ensemble learning for short-term electricity consumption forecasting. *Energies*, *11*(4), 1–32. https://doi.org/10.3390/en11040949

Dong, X., Yu, Z., Cao, W., Shi, Y., & Ma, Q. (2020). A survey on ensemble learning. *Frontiers of Computer Science*, *14*(2), 241–258. https://doi.org/10.1007/s11704-019-8208-z

ElAraby, M. E., Elzeki, O. M., Shams, M. Y., Mahmoud, A., & Salem, H. (2022). A novel Gray-Scale spatial exploitation learning Net for COVID-19 by crawling Internet resources. *Biomedical Signal Processing and Control*, *73*(January), 103441. https://doi.org/10.1016/j.bspc.2021.103441

Elzeki, O. M., Elfattah, M. A., Salem, H., Hassanien, A. E., & Shams, M. (2021). A novel perceptual two layer image fusion using deep learning for imbalanced COVID-19 dataset. *PeerJ Computer Science*, *7*, 1–35. https://doi.org/10.7717/PEERJ-CS.364

Elzeki, O. M., Shams, M., Sarhan, S., Elfattah, M. A., & Hassanien, A. E. (2021). COVID-19: a new deep learning computer-aided model for classification. *PeerJ Computer Science*, *7*, 1–33. https://doi.org/10.7717/peerj-cs.358

Fan, J., Upadhye, S., & Worster, A. (2006). Understanding receiver operating characteristic (ROC) curves. *CJEM*, *8*(01), 19–20. https://doi.org/10.1017/S1481803500013336

Feng, J., Xu, H., Mannor, S., & Yan, S. (2014). Robust logistic regression and classification. *Advances in Neural Information Processing Systems*, *1*(January), 253–261.

Friedman, J. (2001). Greedy Function Approximation : A Gradient Boosting Machine *Annals of statistics*, 29(5), 1189-1232.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics and Data Analysis*, *38*(4), 367–378. https://doi.org/10.1016/S0167-9473(01)00065-2

Gage, A., Brunson, K., Morris, K., Wallen, S. L., Dhau, J., Gohel, H., & Kaushik, A. (2021). Perspectives of Manipulative and High-Performance Nanosystems to Manage Consequences of Emerging New Severe Acute Respiratory Syndrome Coronavirus 2 Variants. *Frontiers in Nanotechnology*, *3*(June), 1–7. https://doi.org/10.3389/fnano.2021.700888

Galassi, A., Lippi, M., & Torroni, P. (2021). Attention in Natural Language Processing. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(10), 4291–4308. https://doi.org/10.1109/TNNLS.2020.3019893

Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN Model-Based Approach in Classification. *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, Italy*, 2888, 986–996. https://doi.org/10.1007/978-3-540-39964-3_62

Ha, D., Dai, A., & Le, Q. V. (2016). HyperNetworks. *ArXiv:1609.09106v4*. http://arxiv.org/abs/1609.09106

Hai-Dong, L., Ya-Juan, Y., & Lu, L. (2022). In the context of COVID-19: the impact of employees' risk perception on work engagement. *Connection Science*, *34*(1), 1367–1383. https://doi.org/10.1080/09540091.2022.2071839

Haimed, A. M. A., Saba, T., Albasha, A., Rehman, A., & Kolivand, M. (2021). Viral reverse engineering using Artificial Intelligence and big data COVID-19 infection with Long Short-term Memory (LSTM). *Environmental Technology & Innovation*, *22*, 101531. https://doi.org/10.1016/j.eti.2021.101531

Hassan, E., Shams, M. Y., Hikal, N. A., & Elmougy, S. (2023). COVID-19 Diagnosis-Based Deep Learning Approaches for COVIDx Dataset: A Preliminary Survey. In Artificial Intelligence for Disease Diagnosis and Prognosis in Smart Healthcare (1st ed., pp. 107–122). CRC Press

Hassan, E., Shams, M. Y., Hikal, N. A., & Elmougy, S. (2024). Detecting COVID-19 in chest CT images based on several pre-trained models. *Multimedia Tools and Applications*, *Dl*. https://doi.org/10.1007/s11042-023-17990-3

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 770–778. https://doi.org/10.1109/CVPR.2016.90

Hendrycks, D., & Gimpel, K. (2016). Gaussian Error Linear Units (GELUs). *ArXiv Preprint ArXiv:1606.08415*, 1–10. http://arxiv.org/abs/1606.08415

Hochreiter, S., & Schmidhuber, J. (1997). LONG SHORT-TERM MEMORY. *Neural Computation*, *9*(8), 1735–1780.

Hossain, M. K., Hassanzadeganroudsari, M., & Apostolopoulos, V. (2021). The emergence of new strains of SARS-CoV-2. What does it mean for COVID-19 vaccines? *Expert Review of Vaccines*, *20*(6), 635–638. https://doi.org/10.1080/14760584.2021.1915140

Hossain, M. S., Pathan, A. Q. M. S. U., Islam, M. N., Tonmoy, M. I. Q., Rakib, M. I., Munim, M. A., Saha, O., Fariha, A., Al Reza, H., Roy, M., Bahadur, N. M., & Rahaman, M. M. (2021). Genome-wide identification and prediction of SARS-CoV-2 mutations show an abundance of variants: Integrated study of bioinformatics and deep neural learning. *Informatics in Medicine Unlocked*, *27*, 100798. https://doi.org/10.1016/j.imu.2021.100798

Huang, Y., Yang, C., Xu, X. feng, Xu, W., & Liu, S. wen. (2020). Structural and functional properties of SARS-CoV-2 spike protein: potential antivirus drug development for COVID-19. *Acta Pharmacologica Sinica*, *41*(9), 1141–1149. https://doi.org/10.1038/s41401-020-0485-4

Jackson, C. B., Farzan, M., Chen, B., & Choe, H. (2022). Mechanisms of SARS-CoV-2 entry into cells. *Nature Reviews Molecular Cell Biology*, *23*(1), 3–20. https://doi.org/10.1038/s41580-021-00418-x

Jaimes, J. A., André, N. M., Chappie, J. S., Millet, J. K., & Whittaker, G. R. (2020). Phylogenetic Analysis and Structural Modeling of SARS-CoV-2 Spike Protein Reveals an Evolutionary Distinct and Proteolytically Sensitive Activation Loop. In *Journal of Molecular Biology*, *432*(10), 3309-3325. https://doi.org/10.1016/j.jmb.2020.04.009

Kalyan, K. S., Rajasekharan, A., & Sangeetha, S. (2021). AMMUS : A Survey of Transformer-based Pretrained Models in Natural Language Processing. *ArXiv Preprint ArXiv:2108.05542.*, 1–42. http://arxiv.org/abs/2108.05542

Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *International Joint Conference of Artificial Intelligence*, 14(2), 1137-1145.

Kotsiantis, S. B. (2013). Decision trees: A recent overview. *Artificial Intelligence Review*, *39*(4), 261–283. https://doi.org/10.1007/s10462-011-9272-4

Li, M., Zhao, B., Yin, R., Lu, C., Guo, F., & Zeng, M. (2023). GraphLncLoc: Long non-coding RNA subcellular localization prediction using graph convolutional networks based on sequence to graph transformation. *Briefings in Bioinformatics*, *24*(1), 1–12. https://doi.org/10.1093/bib/bbac565

Lopez-Rincon, A., Perez-Romero, C. A., Tonda, A., Mendoza-Maldonado, L., Claassen, E., Garssen, J., & Kraneveld, A. D. (2021). Design of Specific Primer Sets for the Detection of B.1.1.7, B.1.351 and P.1 SARS-CoV-2 Variants using Deep Learning. *BioRxiv*, *70*, 2021.01.20.427043. https://doi.org/10.1101/2021.01.20.427043

Luque, A., Carrasco, A., Martín, A., & de las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognition*, *91*, 216–231. https://doi.org/10.1016/j.patcog.2019.02.023

Madhi, S. A., Kwatra, G., Myers, J. E., Jassat, W., Dhar, N., Mukendi, C. K., Nana, A. J., Blumberg, L., Welch, R., Ngorima-Mabhena, N., & Mutevedzi, P. C. (2022). Population Immunity and Covid-19 Severity with Omicron Variant in South Africa. *New England Journal of Medicine*, *386*(14), 1314–1326. https://doi.org/10.1056/nejmoa2119658

Mai, F., Pannatier, A., Fehr, F., Chen, H., Marelli, F., Fleuret, F., & Henderson, J. (2023). HyperMixer: An MLP-based Low Cost Alternative to Transformers. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, Canada, *1*(2020), 15632–15654. https://doi.org/10.18653/v1/2023.acl-long.871

Mbow, M., Koide, H., & Sakurai, K. (2021). An Intrusion Detection System for Imbalanced Dataset Based on Deep Learning. *2021 9th International Symposium on Computing and Networking (CANDAR)*, Japan, 38–47. https://doi.org/10.1109/CANDAR53791.2021.00013

Memon, N., Patel, S. B., & Patel, D. P. (2019). Comparative Analysis of Artificial Neural Network and XGBoost Algorithm for PolSAR Image Classification. *In International Conference on Pattern Recognition and Machine Intelligence*, , Germany, 452–460. https://doi.org/10.1007/978-3-030-34869-4_49

Mohamed, T., Sayed, S., Salah, A., & Houssein, E. H. (2021). Long Short-Term Memory Neural Networks for RNA Viruses Mutations Prediction. *Mathematical Problems in Engineering*, 2021(1), 9980347. https://doi.org/10.1155/2021/9980347

Nawaz, M. S., Fournier-Viger, P., Shojaee, A., & Fujita, H. (2021). Using artificial intelligence techniques for COVID-19 genome analysis. *Applied Intelligence*, *51*(5), 3086–3103. https://doi.org/10.1007/s10489-021-02193-w

Norouzi, M., Fleet, D. J., & Salakhutdinov, R. (2012). Hamming distance metric learning. *Advances in Neural Information Processing Systems*, *2*, 1061–1069.

Pacal, I. (2024a). A novel Swin transformer approach utilizing residual multi-layer perceptron for diagnosing brain tumors in MRI images. *International Journal of Machine Learning and Cybernetics*, 1-19. https://doi.org/10.1007/s13042-024-02110-w

Pacal, I. (2024b). MaxCerVixT: A novel lightweight vision transformer-based Approach for precise cervical cancer detection. *Knowledge-Based Systems*, *289*(October 2023). https://doi.org/10.1016/j.knosys.2024.111482

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., & ... (2017). Automatic differentiation in pytorch. *31st Conference on Neural Information Processing Systems (NIPS 2017)*, USA, 1–4. https://openreview.net/forum?id=BJJsrmfCZ

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, douard. (2011). Scikit-learn: Machine Learning in Python Fabian. *Journal Of Machine Learning Research*, *12*, 2825–2830.

Peng, F., Xia, Y., & Li, W. (2023). Prediction of Antigenic Distance in Influenza A Using Attribute Network Embedding. *Viruses*, *15*(7), 1–20. https://doi.org/10.3390/v15071478

Pirjatullah, Kartini, D., Nugrahadi, D. T., Muliadi, & Farmadi, A. (2021). Hyperparameter Tuning using GridsearchCV on the Comparison of the Activation Function of the ELM Method to the Classification of Pneumonia in Toddlers. *2021 4th International Conference of Computer and Informatics Engineering(IC2IE)*, Indonesia, 390–395. https://doi.org/10.1109/IC2IE53219.2021.9649207

Post, P., Complications, C., Gupta, A., Jain, V., & Singh, A. (2021). Stacking Ensemble - Based Intelligent Machine Learning. *New Generation Computing*, *0123456789*, 987–1007. https://doi.org/10.1007/s00354-021-00144-0

Qin, L., Ding, X., Li, Y., Chen, Q., Meng, J., & Jiang, T. (2021). Co-mutation modules capture the evolution and transmission patterns of SARS-CoV-2. *Briefings in Bioinformatics*, *22*(6), 1–10. https://doi.org/10.1093/bib/bbab222

Saha, I., Ghosh, N., Maity, D., Sharma, N., Sarkar, J. P., & Mitra, K. (2020). Genome-wide analysis of Indian SARS-CoV-2 genomes for the identification of genetic mutation and SNP. *Infection, Genetics and Evolution*, *85*, 104457. https://doi.org/10.1016/j.meegid.2020.104457

Salama, M. A., Hassanien, A. E., & Mostafa, A. (2016). The prediction of virus mutation using neural networks and rough set techniques. *Eurasip Journal on Bioinformatics and Systems Biology*, *2016*(1), 1–11. https://doi.org/10.1186/s13637-016-0042-0

Sasirekha, K., & Baby, P. (2013). Agglomerative Hierarchical Clustering Algorithm-A Review. *International Journal of Scientific and Research Publications*, *3*(3), 83–85.

Serena Low, W. C., Chuah, J. H., Tee, C. A. T. H., Anis, S., Shoaib, M. A., Faisal, A., Khalil, A., & Lai, K. W. (2021). An Overview of Deep Learning Techniques on Chest X-Ray and CT Scan Identification of COVID-19. *Computational and Mathematical Methods in Medicine*, *2021(1)*, 5528144. https://doi.org/10.1155/2021/5528144

Sewell, M. (2011). Ensemble Learning. *Research Note*, *11*(02), 1–12.

Shaikh, F., Andersen, M. B., Sohail, M. R., Mulero, F., Awan, O., Dupont-Roettger, D., Kubassova, O., Dehmeshki, J., & Bisdas, S. (2021). Current Landscape of Imaging and the Potential Role for Artificial Intelligence in the Management of COVID-19. *Current Problems in Diagnostic Radiology*, *50*(3), 430-435. https://doi.org/10.1067/j.cpradiol.2020.06.009

Sharma, A., Ahmad Farouk, I., & Lal, S. K. (2021). COVID-19: A review on the novel coronavirus disease evolution, transmission, detection, control and prevention. *Viruses*, *13*(2), 202. https://doi.org/10.3390/v13020202

Shereen, M. A., Khan, S., Kazmi, A., Bashir, N., & Siddique, R. (2020). COVID-19 infection: Emergence, transmission, and characteristics of human coronaviruses. *Journal of Advanced Research*, *24*, 91–98. https://doi.org/10.1016/j.jare.2020.03.005

Shiehzadegan, S., Alaghemand, N., Fox, M., & Venketaraman, V. (2021). Analysis of the Delta Variant B.1.617.2 COVID-19. *Clinics and Practice*, *11*(4), 778–784. https://doi.org/10.3390/clinpract11040093

Shrestha, H., Dhasarathan, C., Kumar, M., Nidhya, R., Shankar, A., & Kumar, M. (2022). A Deep Learning Based Convolution Neural Network-DCNN Approach to Detect Brain Tumor. *Proceedings of Academia-Industry Consortium for Data Science: AICDS 2020,* Singapore, 1411, 115–127. https://doi.org/10.1007/978-981-16-6887-6_11

Shu, Y., & McCauley, J. (2017). GISAID: Global initiative on sharing all influenza data – from vision to reality. *Eurosurveillance*, *22*(13), 2–4. https://doi.org/10.2807/1560-7917.ES.2017.22.13.30494

Sohrabi, C., Alsafi, Z., O'Neill, N., Khan, M., Kerwan, A., Al-Jabir, A., Iosifidis, C., & Agha, R. (2020). World Health Organization declares global emergency: A review of the 2019 novel coronavirus (COVID-19). In *International Journal of Surgery* , 76, 71–76. https://doi.org/10.1016/j.ijsu.2020.02.034

Sokhansanj, B. A., & Rosen, G. L. (2022). Predicting COVID-19 disease severity from SARS-CoV-2 spike protein sequence by mixed effects machine learning. In *Computers in Biology and Medicine* (Vol. 149). https://doi.org/10.1016/j.compbiomed.2022.105969

Suri, J. S., Puvvula, A., Biswas, M., Majhail, M., Saba, L., Faa, G., Singh, I. M., Oberleitner, R., Turk, M., Chadha, P. S., Johri, A. M., Sanches, J. M., Khanna, N. N., Viskovic, K., Mavrogeni, S., Laird, J. R., Pareek, G., Miner, M., Sobel, D. W., … Naidu, S. (2020). COVID-19 pathways for brain and heart injury in comorbidity patients: A role of medical imaging and artificial intelligence-based COVID severity classification: A review. *Computers in Biology and Medicine*, *124*(January), 103960. https://doi.org/10.1016/j.compbiomed.2020.103960

Tang, L., Wu, T., Chen, X., Wen, S., Zhou, W., Zhu, X., & Xiang, Y. (2024). How COVID-19 impacts telehealth: an empirical study of telehealth services, users and the use of metaverse. *Connection Science*, *36*(1), 2282942. https://doi.org/10.1080/09540091.2023.2282942

Tarek, Z., Shams, M. Y., Towfek, S. K., Alkahtani, H. K., Ibrahim, A., Abdelhamid, A. A., Eid, M. M., Khodadadi, N., Abualigah, L., Khafaga, D. S., & Elshewey, A. M. (2023). An Optimized Model Based on Deep Learning and Gated Recurrent Unit for COVID-19 Death Prediction. *Biomimetics*, *8*(7), 552. https://doi.org/10.3390/biomimetics8070552

Taspinar, Y. S., Cinar, I., & Koklu, M. (2022). Classification by a stacking model using CNN features for COVID-19 infection diagnosis. *Journal of X-Ray Science and Technology*, *30*(1), 73–88. https://doi.org/10.3233/XST-211031

Thara, T. D. K., Prema, P. S., & Xiong, F. (2019). Auto-detection of epileptic seizure events using deep neural network with different feature scaling techniques. In *Pattern Recognition Letters*, *128*, 544-550. https://doi.org/10.1016/j.patrec.2019.10.029

Thölke, P., Jose, Y., Ramos, M., & Abdelhedi, H. (2022). Class imbalance should not throw you off balance : Choosing the right classifiers and performance metrics for brain decoding with imbalanced data. *NeuroImage*, 277, 120253. https://doi.org/10.1016/j.neuroimage.2023.120253

Toche Tchio, G. M., Kenfack, J., Kassegne, D., Menga, F., & Ouro-Djobo, S. S. (2024). A Comprehensive Review of Supervised Learning Algorithms for the Diagnosis of Photovoltaic Systems, Proposing a New Approach Using an Ensemble Learning Algorithm. *Applied Sciences*, *14*(5), 2072. https://doi.org/10.3390/app14052072

Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lucic, M., & Dosovitskiy, A. (2021). MLP-Mixer: An all-MLP Architecture for Vision. *Advances in Neural Information Processing Systems*, 34, 24261–24272.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Voita, E., Talbot, D., Moiseev, F., Sennrich, R., & Titov, I. (2020). Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Italy,5797–5808. https://doi.org/10.18653/v1/p19-1580

Wang, R., Hozumi, Y., Yin, C., & Wei, G.-W. (2020). Mutations on COVID-19 diagnostic targets. *Genomics*, *112*(6), 5204–5213. https://doi.org/10.1016/j.ygeno.2020.09.028

DSÖ (2023, 26 Mayıs). *Novel-Coronavirus-2019 Www.Who.Int*. https://www.who.int/es/emergencies/diseases/novel-coronavirus-2019

Wu, C. rong, Yin, W. chao, Jiang, Y., & Xu, H. E. (2022). Structure genomics of SARS-CoV-2 and its Omicron variant: drug design templates for COVID-19. *Acta Pharmacologica Sinica*, *43*(12), 3021-3033. https://doi.org/10.1038/s41401-021-00851-w

Wu, F., Zhao, S., Yu, B., Chen, Y. M., Wang, W., Song, Z. G., Hu, Y., Tao, Z. W., Tian, J. H., Pei, Y. Y., Yuan, M. L., Zhang, Y. L., Dai, F. H., Liu, Y., Wang, Q. M., Zheng, J. J., Xu, L., Holmes, E. C., & Zhang, Y. Z. (2020). A new coronavirus associated with human respiratory disease in China. *Nature*, *579*(7798), 265–269. https://doi.org/10.1038/s41586-020-2008-3

Yin, R., Luo, Z., Zhuang, P., Zeng, M., Li, M., Lin, Z., & Kwoh, C. K. (2023). ViPal: A framework for virulence prediction of influenza viruses with prior viral knowledge using genomic sequences. *Journal of Biomedical Informatics*, *142*, 142, 104388. https://doi.org/10.1016/j.jbi.2023.104388

Yin, R., Luusua, E., Dabrowski, J., Zhang, Y., & Kwoh, C. K. (2020). Tempel: Time-series mutation prediction of influenza A viruses via attention-based recurrent neural networks. *Bioinformatics*, *36*(9), 2697–2704. https://doi.org/10.1093/bioinformatics/btaa050

Yin, R., Thwin, N. N., Zhuang, P., Lin, Z., & Kwoh, C. K. (2022). IAV-CNN: A 2D Convolutional Neural Network Model to Predict Antigenic Variants of Influenza A Virus. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *19*(6), 3497–3506. https://doi.org/10.1109/TCBB.2021.3108971

Yin, R., Ye, B., & Bian, J. (2023). CLCAP: Contrastive learning improves antigenicity prediction for influenza A virus using convolutional neural networks. *Methods*, *220*(July), 21–28. https://doi.org/10.1016/j.ymeth.2023.10.010

Zainol Rashid, Z., Othman, S. N., Abdul Samat, M. N., Ali, U. K., & Wong, K. K. (2020). Diagnostic performance of COVID-19 serology assays. *Malaysian Journal of Pathology*, *42*(1), 13–21.

Zaremba, W., Sutskever, I., & Vinyals, O. (2014). Recurrent Neural Network Regularization. *ArXiv:1409.2329, 2013*, 1–8. http://arxiv.org/abs/1409.2329

Zhang, J., Xiao, T., Cai, Y., & Chen, B. (2021). Structure of SARS-CoV-2 spike protein. *Current Opinion in Virology*, *50*, 173–182. https://doi.org/10.1016/j.coviro.2021.08.010

Zhou, B., Zhou, H., Zhang, X., Xu, X., Chai, Y., Zheng, Z., Kot, A. C., & Zhou, Z. (2023a). TEMPO: A transformer-based mutation prediction framework for SARS-CoV-2 evolution. *Computers in Biology and Medicine*, *152*, 106264. https://doi.org/10.1016/j.compbiomed.2022.106264

Zhou, B., Zhou, H., Zhang, X., Xu, X., Chai, Y., Zheng, Z., Kot, A. C., & Zhou, Z. (2023b, 10 Ekim). *SARS-CoV-2 S Protein Dataset*. https://github.com/facebookresearch/mlqe/tree/main/data

EKLER

EK A. Önerilen TfrAdmCov modeli için ek tablolar

Tablo A.1. Agglomerative kümeleme algoritmasının parametreleri ve bu parametrelerin değerleri (Burukanlı ve Yumuşak, 2024c).

Parametre adı	**Değeri**
n_clusters	2
affinity	euclidean
memory	None
connectivity	None
compute_full_tree	auto
linkage	ward
distance_threshold	None
compute_distances	False

Tablo A.2. SVM modelinin rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024c).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
C =1.0, kernel ='rbf', probability = True	C = [1.0, 2.0,3.0,4.0,5.0] kernel =['linear', 'poly', 'rbf', 'sigmoid', 'precomputed'] probability =[True, False]	Holdout	Stratified 10-kat çapraz doğrulama
		C= 3.0,	K=1 için C= 4.0, kernel= 'rbf', probability= True,

Tablo A.2. (Devamı) SVM modelinin rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024c).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
		kernel= 'rbf', probability= True	K=2 için C= 4.0, kernel= 'rbf', probability= True, K=3 için C= 5.0, kernel= 'rbf', probability= True, K=4 için C= 5.0, kernel= 'rbf', probability= True, K=5 için C= 4.0, kernel= 'rbf', probability= True, K=6 için C= 5.0, kernel= 'rbf', probability= True, K=7 için C= 4.0, kernel= 'rbf', probability= True, K=8 için C= 4.0, kernel= 'rbf', probability= True, K=9 için C= 5.0, kernel= 'rbf', probability= True, K=10 için C= 5.0, kernel= 'rbf', probability= True

Tablo A. 3. KNN modelinin rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024c).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
n_neighbors =5, weights ='uniform', algorithm ='auto'	n_neighbors =[3,5,7,9,11] weights = ['uniform', 'distance'] algorithm = ['auto', 'ball_tree', 'kd_tree', 'brute']	n_neighbors = 3, weights = 'distance', algorithm = 'auto'	K=1 için n_neighbors = 3, weights = 'distance', algorithm = 'auto', K=2 için n_neighbors = 3, weights = 'distance', algorithm = 'auto', K=3 için n_neighbors = 5, weights = 'distance', algorithm = 'auto', K=4 için n_neighbors = 3, weights = 'distance', algorithm = 'auto', K=5 için n_neighbors = 3, weights = 'distance', algorithm = 'auto', K=6 için n_neighbors = 3, weights = 'distance', algorithm = 'auto', K=7 için n_neighbors = 3, weights = 'distance', algorithm = 'auto', K=8 için n_neighbors = 5, weights = 'distance', algorithm = 'auto', K=9 için n_neighbors = 5, weights = 'distance', algorithm = 'auto', K=10 için n_neighbors =3, weights = 'distance', algorithm = 'auto'

Tablo A. 4. XGBoost modelinin rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024c).

Test edilen parametreler in varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
booster = None, learning_rate = None, n_estimators =100	booster=[‘gbtree’, ‘gblinear’,’dart’,None] learning_rate=[0.001,0.01,0.1, 1,None] n_estimators=[50,100,150,200 ,250]	booster = 'gbtree', learning_rate = 0.1, n_estimators =250	K=1 için booster = 'gbtree' , learning_rate = 1, n_estimators = 100, K=2 için booster = 'gbtree', learning_rate = 1, n_estimators = 50, K=3 için booster = 'gbtree', learning_rate = 1, n_estimators = 50, K=4 için booster = 'gbtree', learning_rate = 1, n_estimators =100, K=5 için booster = 'gbtree', learning_rate = 0.1, n_estimators = 150, K=6 için booster = 'gbtree', learning_rate = 1, n_estimators =250, K=7 için booster = 'gbtree', learning_rate = None, n_estimators = 250, K=8 için booster = 'gbtree', learning_rate = None, n_estimators =150, K=9 için booster = 'gbtree', learning_rate = None, n_estimators =50, K=10 için booster = 'gbtree', learning_rate = None, n_estimators =250

Tablo A.5. LR modelinin rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024c).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
C =1.0, solver ='lbfgs', max_iter =100	C = np.linspace(1, 10, num=5) solver = ['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'] max_iter=[100, 1000, 10000, 100000, 1000000]	C= 10.0, solver = 'lbfgs', max_iter = 100	K=1 için C= 10.0, solver ='lbfgs', max_iter =1000, K=2 için C= 10.0, solver ='saga', max_iter =10000, K=3 için C= 7.75, solver ='newton-cg', max_iter =100, K=4 için C= 7.75, solver ='liblinear', max_iter =100, K=5 için C= 7.75, solver ='newton-cg', max_iter =100, K=6 için C= 7.75, solver ='saga', max_iter =10000, K=7 için C= 10.0, solver ='newton-cg', max_iter =100, K=8 için C= 10.0, solver ='newton-cg', max_iter =100, K=9 için C= 10.0, solver ='sag', max_iter =10000, K=10 için C=7.75, solver ='newton-cg', max_iter =100

EK B. Önerilen StackGridCov modeli için tablolar

Tablo B. 1. SVM algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
gamma = 'scale', class_weight = None, probability = False	gamma =['scale', 'auto', 10.0,20.0,30.0] class_weight =['balanced', None] probability =[True, False]	gamma = 'auto', class_weight = 'balanced', probability = True	K=1 için gamma = 'auto', class_weight = None, probability=False, K=2 için gamma = 'auto', class_weight = 'balanced', probability= True, K=3 için gamma = 'auto', class_weight = None, probability= False, K=4 için gamma = 'scale', class_weight = 'balanced', probability= True, K=5 için gamma = 'auto', class_weight = 'balanced', probability= False, K=6 için gamma = 'auto', class_weight = None, probability= True, K=7 için gamma = 'auto', class_weight = None, probability= False, K=8 için gamma = 'auto', class_weight = 'balanced', probability= True,

Tablo B.1. (Devamı) SVM algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
			K=9 için gamma = 'scale', class_weight = 'balanced', probability= True, K=10 için gamma = 'auto', class_weight = 'balanced', probability= False

Tablo B.2. RF algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
max_leaf_nodes= None min_samples_leaf=1 n_estimators= 100	max_leaf_nodes= [5,10,15,20,None] min_samples_leaf= [1,2,3,4,5] n_estimators= [50,100,200,300,400]	max_leaf_nodes= 10 min_samples_leaf= 5 n_estimators= 400	K=1 için max_leaf_nodes= 15, min_samples_leaf= 4, n_estimators= 100, K=2 için max_leaf_nodes= 20, min_samples_leaf= 5, n_estimators= 50, K=3 için max_leaf_nodes= 20, min_samples_leaf= 3, n_estimators= 100,

Tablo B.2. (Devamı) RF algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
			K=4 için max_leaf_nodes= 20, min_samples_leaf= 1, n_estimators= 200,K=5 için max_leaf_nodes= 10, min_samples_leaf= 2, n_estimators= 400, K=6 için max_leaf_nodes= 15, min_samples_leaf= 4, n_estimators= 400, K=7 için max_leaf_nodes= 20, min_samples_leaf= 1, n_estimators= 400, K=8 için max_leaf_nodes= 20, min_samples_leaf= 5, n_estimators= 400,K=9 için max_leaf_nodes= 10, min_samples_leaf= 2, n_estimators= 300, K=10 için max_leaf_nodes= 15, min_samples_leaf= 2, n_estimators= 300

Tablo B.3. XGBoost algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
max_depth= None n_estimators= 100 tree_method= None	max_depth= [3,5,6,9,None] n_estimators= [50,100,150,200,250] tree_method= ["auto", "exact", "approx", "hist", "gpu_hist",None]	max_depth= 3 n_estimators= 50 tree_method= 'hist'	K=1 için max_depth= 3, n_estimators= 50, tree_method= 'auto', K=2 için max_depth= 3, n_estimators= 50, tree_method= 'hist', K=3 için max_depth= 3, n_estimators= 50, tree_method= 'hist', K=4 için max_depth= 3, n_estimators= 100, tree_method= 'approx', K=5 için max_depth= 3, n_estimators= 50, tree_method= 'approx', K=6 için max_depth= 3, n_estimators= 50, tree_method= 'hist', K=7 için max_depth= 3, n_estimators= 50, tree_method= 'auto', K=8 için max_depth= 3, n_estimators= 50, tree_method= 'hist', K=9 için max_depth= 3, n_estimators= 50, tree_method= 'approx',

Tablo B.3. (Devamı) XGBoost algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
			K=10 için max_depth= 3, n_estimators= 50, tree_method= 'hist'

Tablo B.4. YSA algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
hidden_layer_sizes = (100,) max_iter= 200 solver= 'adam'	hidden_layer_sizes= [(50,),(100,),(150,),(200,),(250,)] max_iter= [100,200,300,400,500] solver= ['lbfgs', 'sgd', 'adam']	hidden_layer_sizes = (200,) max_iter= 200 solver= 'sgd'	K=1 için hidden_layer_sizes = (250,), max_iter= 100, solver= 'sgd', K=2 için hidden_layer_sizes = (150,), max_iter= 200, solver= 'sgd', K=3 için hidden_layer_sizes = (200,), max_iter= 200, solver= 'sgd',

Tablo B.4. (Devamı) YSA algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
			K=4 için hidden_layer_sizes= (100,), max_iter= 500, solver= 'sgd', K=5 için hidden_layer_sizes= (100,), max_iter= 200, solver= 'sgd', K=6 için hidden_layer_sizes= (250,), max_iter= 100, solver= 'sgd', K=7 için hidden_layer_sizes= (250,), max_iter= 200, solver= 'sgd',K=8 için hidden_layer_sizes= (150,), max_iter= 200, solver= 'sgd', K=9 için hidden_layer_sizes= (100,), max_iter= 200, solver= 'sgd', K=10 için hidden_layer_sizes= (200,), max_iter= 100, solver= 'sgd'

Tablo B.5. DT algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
max_leaf_nodes= None min_impurity_decrease= 0.0 min_samples_split= 2	max_leaf_nodes= [5,10,15,20,None] min_impurity_decrease= [0.0,0.1,0.2,0.3,0.4] min_samples_split= [1,2,3,4,5]	max_leaf_nodes= 15 min_impurity_decrease= 0.0 min_samples_split= 2	K=1 için max_leaf_nodes= 5, min_impurity_decrease= 0.0, min_samples_split= 2, K=2 için max_leaf_nodes= 10, min_impurity_decrease= 0.0, min_samples_split= 2, K=3 için max_leaf_nodes= 20, min_impurity_decrease= 0.0, min_samples_split= 2, K=4 için max_leaf_nodes= 20, min_impurity_decrease= 0.0, min_samples_split= 2, K=5 için max_leaf_nodes= 15, min_impurity_decrease= 0.0, min_samples_split= 2,

Tablo B.5. (Devamı) DT algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
			K=6 için max_leaf_nodes=5, min_impurity_decrease= 0.0, min_samples_split= 2, K=7 için max_leaf_nodes= 20, min_impurity_decrease= 0.0, min_samples_split= 2, K=8 için max_leaf_nodes= 20, min_impurity_decrease= 0.0, min_samples_split= 5,K=8 için max_leaf_nodes= 20, min_impurity_decrease= 0.0, min_samples_split= 5, K=9 için max_leaf_nodes=10, min_impurity_decrease= 0.0, min_samples_split= 2, K=10 için max_leaf_nodes= 20, min_impurity_decrease= 0.0, min_samples_split= 2

Tablo B.6. GB algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
Loss= 'log_loss' min_samples_leaf= 1 min_samples_split=2	Loss= ['log_loss', 'deviance', 'exponential'] min_samples_leaf= [0,1,2,3,4] min_samples_split= [1,2,3,4,5]	Loss= 'deviance' min_samples_leaf= 4 min_samples_split= 2	K=1 için Loss= 'exponential', min_samples_leaf= 1, min_samples_split= 4, K=2 için Loss= 'exponential', min_samples_leaf= 1, min_samples_split=4, K=3 için Loss= 'deviance', min_samples_leaf= 4, min_samples_split=2, K=4 için Loss= 'deviance', min_samples_leaf=4, min_samples_split=4, K=5 için Loss='exponential', min_samples_leaf=2, min_samples_split=4, K=6 için Loss= 'deviance', min_samples_leaf=4, min_samples_split=4, K=7 için Loss= 'deviance', min_samples_leaf=3, min_samples_split=2,

Tablo B.6. (Devamı) GB algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
			K=8 için Loss= 'deviance', min_samples_leaf=2, min_samples_split=4, K=9 için Loss= 'deviance', min_samples_leaf=1, min_samples_split=4, K=10 için Loss= 'exponential', min_samples_leaf=4, min_samples_split=4

Tablo B.7. ET algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
Criterion= "gini" max_depth= None n_estimators=100	Criterion= ["gini", "entropy", "log_loss"] max_depth= [5,10,15,20,None] n_estimators= [50,100,150,200,250]	Criterion= 'gini' max_depth=5 n_estimators=200	K=1 için Criterion= 'gini', max_depth=5, n_estimators= 100,

Tablo B.7. (Devamı) ET algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
			K=2 için Criterion= 'gini', max_depth=5, n_estimators= 100, K=3 için Criterion= 'gini', max_depth=5, n_estimators=200, K=4 için Criterion= 'entropy', max_depth=5, n_estimators=200, For K=5 için Criterion= 'entropy', max_depth=5, n_estimators= 250, K=6 için Criterion= 'entropy', max_depth=5, n_estimators=50, K=7 için Criterion= 'entropy', max_depth=10, n_estimators=50, K=8 için Criterion= 'entropy', max_depth=10, n_estimators=150, K=9 için Criterion= 'entropy', max_depth=5, n_estimators=250, K=10 için Criterion= 'entropy', max_depth=5, n_estimators=250

Tablo B.8. Önerilen StackGridCov algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
	final_estimator=[LogisticRegression(), AdaBoostClassifier()] stack_method= ['auto', 'predict_proba', 'decision_function', 'predict'] cv =[None, 10]	Holdout	Stratified 10-kat çapraz doğrulama
	StackGridCov sınıflandırıcıları		K=1 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10,
final_estimator = LogisticRegression() stack_method= 'auto cv =None	estimators =[SVC(), RandomForestClassifier(), XGBClassifier(), MLPClassifier(), DecisionTreeClassifier(), GradientBoostingClassifier(), ExtraTreesClassifier()]	final_estimator = AdaBoostClassifier() stack_method='auto cv =10	K=2 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10, K=3 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10, K=4 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10, K=5 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10, K=6 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10, K=7 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10,

Tablo B.8. (Devamı) Önerilen StackGridCov algoritmasının rastgele seçilen 3 özelliği için GridSearchCV algoritması kullanılarak elde edilen en iyi değerler (Burukanlı ve Yumuşak, 2024b).

Test edilen parametrelerin varsayılan değerleri	GridSearchCV algoritması ile test edilecek parametreler ve bu parametrelerin değerleri	GridSearchCV kullanılarak elde edilen en iyi değerler	
		Holdout	Stratified 10-kat çapraz doğrulama
	StackGridCov sınıflandırıcıları		K=8 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10, K=9 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10, K=10 için final_estimator= AdaBoostClassifier(),stack_method='auto, cv =10,

ÖZGEÇMİŞ

Ad-Soyad : Mehmet BURUKANLI

ÖĞRENİM DURUMU:

- **Lisans :** 2013, Harran Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü
- **Yüksek Lisans :** 2020, Bitlis Eren Üniversitesi, Elektrik-Elektronik Mühendisliği Anabilim Dalı, Elektrik-Elektronik Mühendisliği Programı

MESLEKİ DENEYİM:

- 2014-2016 yılları arasında Muş Alparslan Üniversitesi'nde araştırma görevlisi olarak çalıştı.
- 2016 yılından beri Bitlis Eren Üniversitesi'nde öğretim görevlisi olarak çalışmaya devam etmektedir.

TEZDEN TÜRETİLEN ESERLER:

Burukanlı, M., & Yumuşak, N. (2024a). TfrAdmCov: A Robust Transformer Encoder based Model with Adam Optimizer Algorithm for COVID-19 Mutation Prediction. *Connection Science*, 36(1), 2365334.

Burukanlı, M., & Yumuşak, N. (2024b). StackGridCov: A Robust Stacking Ensemble Learning Based Model Integrated with GridSearchCV Hyperparameter Tuning Technique for Mutation Prediction of COVID-19 Virus. *Neural Computing and Applications*, (in Review Process – 13.11.2023).

Burukanlı, M., & Yumuşak, N. (2024c). COVID-19 Virus Mutation Prediction with LSTM and Attention Mechanisms. *The Computer Journal,* bxae058.

DİĞER ESERLER:

Burukanlı, M., & Yumuşak, N. (2022). Mutation detection and analysis for COVID 19 virus isolated in Turkey. *IV. International Ankara Multidisciplinary Studies Congress*, Turkey, 887–891.