



# Düzce Üniversitesi Bilim ve Teknoloji Dergisi

Araştırma Makalesi

## Derin Öğrenme Tabanlı Trafik Yoğunluğu Tahmini: İstanbul İçin Bir Vaka Çalışması

 Anıl UTKU\*

Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi, Munzur Üniversitesi, Tunceli, TÜRKİYE

\* Sorumlu yazarın e-posta adresi: anilutku@munzur.edu.tr

DOI:10.29130/dubited.1139534

### ÖZ

Trafik yoğunluk tahmini, kullanıcıların daha iyi seyahat kararları verebilmeleri, trafik sıkışıklığının hafifletilmesi, zaman ve yakıt tasarrufu sağlanması ile trafik işlem verimliliğinin artırılması açısından önemlidir. Akıllı ulaşım sistemlerinin gelişmesi ve yaygınlaşmasıyla birlikte trafik yoğunluğunun tahmin edilmesi giderek daha fazla ilgi görmeye başladı. Trafik yoğunluk tahmini, büyük ölçüde geçmiş ve gerçek zamanlı trafik verilerine bağlıdır. Sensörler, kameralar, mobil cihazlar ve sosyal medya gibi kaynaklarından anlık olarak büyük miktarlarda trafik verileri elde edilmektedir. Giderek artan trafik verileri, trafik yönetimi sorununu çözebilmek amacıyla yapay zekâ teknolojilerinin kullanımını ön plana çıkarmaktadır. Bu çalışmada, trafik yoğunluk tahminine yönelik LSTM tabanlı bir tahmin modeli geliştirilmiştir. Geliştirilen tahmin modeli LR, RF, SVM, MLP, CNN ve Recurrent Neural Network (RNN) ile İstanbul'un trafik verileri kullanılarak test edilmiştir. Deneysel sonuçlar, geliştirilen LSTM tabanlı modelin karşılaştırılan modellere göre daha başarılı sonuçlara sahip olduğunu göstermiştir. Geliştirilen model, araç sayısı ve ortalama hız tahmininde yaklaşık olarak 0,9 R<sup>2</sup> değerine sahip olmuştur.

**Anahtar Kelimeler:** Trafik yoğunluk tahmini, Makine öğrenmesi, Derin öğrenme, LSTM

## Deep Learning Based Traffic Density Prediction: A Case Study for Istanbul

### ABSTRACT

Traffic density prediction is important for users to make better travel decisions, reduce traffic congestion, save time and fuel, and increase traffic processing efficiency. With the development and widespread use of intelligent transportation systems, traffic density prediction has received more and more attention. Traffic density prediction is heavily dependent on historical and real-time traffic data. Large amounts of traffic data are obtained instantly from sources such as sensors, cameras, mobile devices and social media. Increasing traffic data reveals the use of artificial intelligence technologies to solve the traffic management problem. In this study, an LSTM-based prediction model has been developed for traffic density prediction. The developed prediction model has been tested using traffic data of Istanbul with LR, RF, SVM, MLP, CNN and RNN. Experimental results showed that the developed LSTM-based model has more successful results than the compared models. The developed model had approximately 0.9 R<sup>2</sup> value in predicting the number of vehicles and average speed.

**Keywords:** Traffic density prediction, Machine learning, Deep learning, LSTM



Liu ve ark. tarafından yapılan çalışmada [9], trafik akış bilgilerinin uzamsal-zamansal bilgilerini çıkarabilmek amacıyla hibrit Conv-LSTM modeli geliştirilmiştir. Trafik akışının periyodiklik özelliğini çıkarabilmek için tahmin noktasının geçmiş trafik akışı verilerini analiz etmek amacıyla LSTM kullanılmıştır. Deneysel sonuçlar, önerilen modelin mevcut yaklaşımlara kıyasla daha iyi bir tahmin doğruluğu sağladığını göstermektedir.

Duan ve ark. tarafından yapılan çalışmada [10], kentsel trafik akışının tahmini için ağaçlı algoritma ile geliştirilmiş derin bir hibrit sinir ağı önerilmiştir. Önerilen modelde, kentsel trafik akışını tahmin etmek için zamansal bilgileri çıkarmak amacıyla LSTM kullanılmıştır. Mekânsal özellikleri çıkarmak amacıyla ise Convolutional Neural Network (CNN) kullanılmıştır. Önerilen model Ağaçlı (Greedy) algoritma kullanılarak kısa zaman tüketimi için eğitilmektedir. Xi'an şehrindeki taksilerin GPS verileri kullanılarak yapılan deneysel çalışmalar, önerilen CNN-LSTM modelinin mevcut yöntemlere kıyasla daha yüksek tahmin doğruluğu ve daha kısa zaman tüketimi sağladığını göstermiştir.

Lin ve ark. tarafından yapılan çalışmada [11], seyrek otomatik kodlayıcı ve LSTM tabanlı SpAE-LSTM modeli geliştirilmiştir. Geliştirilen model, trafik akışının zamansal ve mekânsal özelliklerini işlemektedir. Seyrek otomatik kodlayıcı, tam bağlantılı katmanlar aracılığıyla uzamsal-zamansal matris içindeki zamansal özellikleri çıkarmaktadır. Trafik akışının mekânsal-zamansal özelliklerini yakalamak ve tahminde bulunmak için LSTM kullanılmaktadır. Veriseti olarak Çin'in Chengdu şehrinin Qingyang Bölgesi'nin trafik verileri kullanılmıştır. Deneysel sonuçlar, önerilen modelin seyrek otomatik kodlayıcı ve LSTM'den daha başarılı sonuçlar verdiğini göstermiştir.

Mohammed ve Kianfar tarafından yapılan çalışmada [12], trafik hacmi, trafik hızı ve karayolu bölümlerinin doluluğu gibi koşulların tahminine yönelik Deep Neural Networks (DNN), Random Forest (RF), Gradient Boosting ve Generalized Linear Model'in (GLM) karşılaştırmalı bir analizi yapılmıştır. Veriseti olarak Amerika Birleşik Devletleri'ndeki St. Louis, Missouri'deki Interstate 64'ten elde edilen veriler kullanılmıştır. Deneysel sonuçlar, RF'ın karşılaştırılan modellerden daha başarılı sonuçlar verdiğini göstermiştir. Trafik tahmin sürecine trafik akış hızı, doluluk ve zamanın dâhil edilmesinin trafik tahmin hatasını azalttığı görülmüştür.

Zhang ve ark. tarafından yapılan çalışmada [13], CNN tabanlı kısa vadeli trafik akışı tahmin modeli önerilmiştir. Önerilen modelde, zaman gecikmeleri ve konumsal veri miktarları Spatio-Temporal Feature Selection Algorithm (STFSA) ile belirlenir ve seçilen uzay-zamansal trafik akış özellikleri gerçek verilerden çıkarılarak iki boyutlu hale dönüştürülür. CNN, bu özellikler kullanılarak eğitilir. Deneysel sonuçlar, önerilen modelin doğruluk açısından kNN, SARIMA, ANN ve Support Vector Machine (SVM)'den daha iyi performans elde ettiğini göstermiştir.

Bratsas ve ark. tarafından yapılan çalışmada [14], Yunanistan'ın Selanik karayolu ağundan toplanan araştırma verilerini kullanarak, LR, RF, SVM ve Multilayer Perceptron (MLP) modellerinin karşılaştırmalı bir analizi sunulmuştur. Deneysel çalışmalar, üç tür senaryoda kümelenmiş çoklu testler ile gerçekleştirilmiştir. İlk senaryo, algoritmaları rastgele seçilen farklı yollarda rastgele seçilen belirli tarihlerde test etmektedir. İkinci senaryo, 15 dakikalık aralıklarla rastgele seçilen yollarda algoritmaları test etmektedir. Üçüncü senaryo, algoritmaları tüm gün boyunca rastgele yollarda test etmektedir. Deneysel sonuçlar, MLP'nin karşılaştırılan modellere göre daha başarılı sonuçlar verdiğini göstermiştir.

Taş ve Müngen tarafından yapılan çalışmada [15], İstanbul Büyükşehir Belediyesi tarafından sunulan 1 Ocak 2020 - 1 Ocak 2021 tarihleri arasındaki veriler kullanılarak bölgesel trafik yoğunluğunun tahmin edilmesi amaçlanmıştır. Çalışmada bölgesel hava koşulları ve trafik yoğunluk verileri kullanılarak SVM, MLP ve Genel Regresyon Sinir Ağı (General Regression Neural Network-GRNN) modellerinin sonuçları karşılaştırılmıştır. MAE, MAPE ve  $R^2$  metriklerine göre deneysel sonuçlar elde edilmiştir. Deneysel sonuçlar önerilen sinir ağı modelinin %90 başarı oranı ile trafik yoğunluk tahmininde başarılı olduğunu göstermiştir.

Bu çalışmanın literatüre olan temel katkıları şu şekilde özetlenebilir:

- Trafik yoğunluk tahmine yönelik bu verisetinin kullanıldığı ilk çalışmadır.
- LSTM tabanlı bir derin öğrenme modeli geliştirilerek Linear Regression (LR), RF, SVM, MLP, CNN ve Recurrent Neural Network (RNN) gibi popüler makine öğrenmesi ve derin öğrenme modelleriyle karşılaştırmalı bir analiz yapılmıştır.
- Bu çalışma vasıtasıyla Türkçe literatüre katkıda bulunabilmek amaçlanmıştır.

## **II. TAHMİN YÖNTEMLERİ**

Bu çalışmada LR, RF, SVM, MLP, CNN, RNN ve geliştirilen LSTM tabanlı modelin karşılaştırmalı bir analizi sunulmuştur.

LR, gelecekteki olayların sonucunu tahmin etmek için bağımsız bir değişken ile bağımlı bir değişken arasında doğrusal ilişki sağlayan bir algoritmadır [16]. LR, gözlemlenen verilere doğrusal bir denklem uydurarak iki değişken arasındaki ilişkiyi modellemeye çalışır. Genel model, bir bağımlı değişken olan  $y$  ile bir veya daha fazla bağımsız değişken olan  $x$  arasında doğrusal bir ilişki olduğunu varsayar. Tahmine dayalı analiz için veri bilimi ve makine öğreniminde kullanılan istatistiksel bir yöntemdir. Bağımsız değişken, değişmeden kalan yorumlayıcı veya açıklayıcı değişkendir. Ancak bağımlı değişken, bağımsız değişkendenki değişimlere göre farklı değerler almaktadır. Regresyon modeli, bağımlı değişkenin değerini tahmin etmektedir [17].

RF, sınıflandırma ve regresyon problemlerinde yaygın olarak kullanılan bir denetimli öğrenme algoritmasıdır. Farklı örnekler üzerinde karar ağaçları oluşturur ve regresyon problemlerinde ortalamayı, sınıflandırma problemlerinde ise çoğunluk oyunu alır. Karmaşık bir problemi çözmek ve modelin performansını iyileştirmek için birden fazla sınıflandırıcıyı birleştirme süreci olan topluluk öğrenme kavramına dayanmaktadır [18]. RF'nin en önemli özelliklerinden biri, regresyon problemlerinde olduğu gibi sürekli değişkenleri ve sınıflandırma durumunda olduğu gibi kategorik değişkenleri içeren verisetlerini işleyebilmesidir. RF, verisetinin çeşitli alt kümelerinde bir dizi karar ağacı içeren ve bu verisetinin tahmin doğruluğunu iyileştirmek için ortalamayı alan bir sınıflandırıcıdır. RF, tek bir karar ağacına güvenmek yerine, her ağaçtan ve tahminlerin çoğunluk oylarına dayalı olarak tahmini alır ve nihai çıktıyı tahmin eder [19].

SVM, sınıflandırma ve regresyon analizi için kullanılan bir öğrenme algoritmasıdır. SVM'in, verileri sınıflandırmak için kullandığı düz çizgiye hiperdüzlem adı verilir. SVM'in amacı, veri noktalarını belirgin bir şekilde sınıflandıran  $n$ -boyutlu bir uzayda bir hiperdüzlem bulmaktır [20]. Hiperdüzlemin her iki tarafında, hiperdüzleme en yakın olan veri noktalarına destek vektörleri adı verilir. Bunlar, hiperdüzlemin konumunu ve yönünü etkiler ve bu sayede SVM'in oluşturulmasına yardımcı olur [21].

MLP, doğrusal ve doğrusal olmayan veriler arasındaki ilişkiyi öğrenen bir sinir ağıdır. Girdiler ve çıktılar arasındaki eşlemenin doğrusal olmadığı bir yapıdadır. MLP, giriş ve çıkış katmanlarına ve birçok nöronun dâhil edildiği bir veya daha fazla gizli katmana sahiptir [22]. MLP'de her katman, bir sonrakini hesaplamalarının sonucuyla, verilerin dâhili temsiliyle besler. Bu süreç, gizli katmanlardan çıktı katmanına kadar devam eder. MLP, maliyet fonksiyonunu en aza indirmek amacıyla ağıdaki ağırlıkların yinelemeli olarak ayarlanmasına izin veren bir öğrenme mekanizması olan geriye yayılımı kullanmaktadır [23].

CNN, temel olarak görüntü verileri gibi iki boyutlu dizileri işleyebilen bir derin öğrenme modelidir. CNN'in, regresyon problemlerindeki sürekli verileri tahmin edebilmesi için ağına sonuna bir regresyon katmanı eklenmektedir. CNN, her biri bir görüntünün farklı özelliklerini algılamayı öğrenen birçok katmana sahip olabilir [24]. Her eğitim görüntüsüne farklı çözünürlüklerde filtreler uygulanır ve her bir görüntünün çıktısı bir sonraki katmana giriş olarak kullanılır. CNN bir giriş katmanı, bir çıkış katmanı ve bunların arasındaki birçok gizli katmandan oluşur. Bu katmanlar, verilere özgü özellikleri öğrenmek

amacıyla verileri deęiřtiren iřlemleri yerine getirir. En yaygın kullanılan katmanlar konvülyasyon, Rectified Linear Unit (ReLU) ve pooling katmanlarıdır. Evriřim, giriř görüntülerini, her biri görüntülerden belirli özellikleri etkinleřtiren bir dizi evriřimli filtreden geçirir. ReLU, negatif deęerleri sıfıra eřleyerek ve pozitif deęerleri koruyarak daha hızlı ve daha etkili eęitim saęlar. Havuzlama, doęrusal olmayan alt örnekleme geręekleřtirerek çıktıyı basitleřtirir ve aęın öęrenmesi gereken parametre sayısını azaltır [25].

RNN, sıralı verileri veya zaman serisi verilerini iřleyen bir derin öęrenme modelidir. Dil çevirisi, doęal dil iřleme ve konuřma tanıma gibi sıralı veya zamansal problemler için yaygın olarak kullanılır. Mevcut girdi ve çıktıyı etkilemek için önceki girdilerden bilgi aldıkları için hafıza bloklarını kullanırlar [26]. RNN, bir önceki adımın çıktısının mevcut adıma girdi olarak beslendięi bir sinir aęıdır. RNN'in ayırt edici özellięi, zaman adımları üzerinden bilgi gönderme yetenekleridir. RNN', zaman adımları arasındaki baęlantılar için ek bir parametre matrisine sahiptir. RNN, her bir zaman adımındaki tahminlerin mevcut girdiye ve önceki zaman adımlarından gelen bilgilere dayandıęı çıktı üretmek için eęitilir. RNN, yönlendirilmiş döngüleri olan bir grafik oluşturur. İleri beslemeli sinir aęlarından farklı olarak, bilgi aędaki mevcut döngülerle katmandan katmana hareket eder [27]. Bu sayede her durum önceki durumlarından etkilenir. Döngülerin eklenmesi sayesinde RNN, geęmiř hesaplamaların saklanmasına izin veren bir hafıza yeteneęine sahiptir.

LSTM, bilginin kalıcı olmasını saęlayan geliřmiř bir RNN'dir. RNN'nin karřılařtıęı kaybolan gradyan problemine çözüm olarak geliřtirilmiřtir. RNN'in eksiklięi, kaybolan gradyan problemi nedeniyle uzun vadeli baęımlılıkları hatırlayamamalarıdır [28]. LSTM, uzun vadeli baęımlılık sorunlarına çözüm sunmak amacıyla geliřtirilmiřtir. LSTM temel olarak girdi kapısı, unutma kapısı ve çıktı kapısı birimlerinden oluřur. Girdi kapısı, hücreye gelen girdiden yeni bilgilerin öęrenmeye çalıřıldıęı birimdir. Unutma kapısı, önceki zaman damgasından gelen bilgilerin hatırlanıp hatırlanmayacaęına veya alakasız ve unutulabilir olup olmadıęına karar verir. Çıktı kapısı ise hücredeki güncellenmiř bilgiyi mevcut zaman damgasından bir sonraki zaman damgasına iletir [29].

### **III. GELİřTİRİLEN DERİN ÖęRENME TABANLI TAHMİN MODELİ**

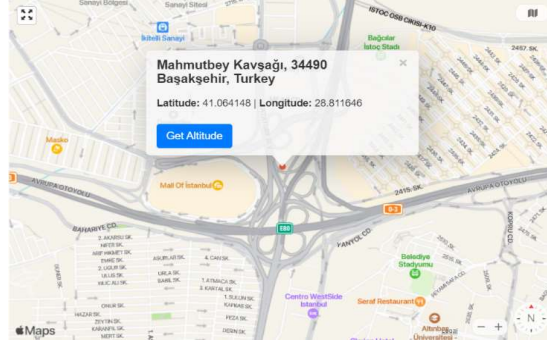
Bu çalıřmada, İstanbul için trafikteki ortalama hız ve araç yoğunluęunun tahminine yönelik LSTM tabanlı bir derin öęrenme modeli geliřtirilmiřtir. Geliřtirilen model ile İstanbul'daki önemli kavřak noktalarından biri olan Bařakřehir Mahmutbey kavřaęındaki ortalama hızın ve trafik yoğunluęunun tahmin edilmesi amaçlanmıřtır. Geliřtirilen LSTM tabanlı model LR, RF, SVM, MLP, CNN ve RNN ile uygulamalı olarak karřılařtırılmıřtır.

#### **A. VERİSETİ**

Bu çalıřmada İstanbul Büyükşehir Belediyesi tarafından sunulan, Ocak 2021-Mayıs 2021 tarihleri arasındaki saatlik trafik yoğunluęu veriseti kullanılmıřtır [30]. Kullanılan veriseti Date\_Time, Longitude, Latitude, Geohash, Minimum\_Speed, Maximum\_Speed, Average\_Speed ve Number\_of\_Vehicles özniteliklerinden oluřmaktadır. Date\_Time Yıl/Ay/Gün ve saat cinsinden zaman bilgisini ifade etmektedir. Longitude boylam deęerini, Latitude ise enlem deęerini ifade etmektedir. Geohash enlem ve boylam deęerlerine göre elde edilen coęrafi konum kodunu ifade etmektedir. Minimum\_Speed, Maximum\_Speed ve Average\_Speed öznitelikleri sırasıyla minimum, maksimum ve ortalama hız deęerlerini ifade etmektedir. Number\_of\_Vehicles ise belirli bir konumdan geęen saatlik araç sayısını ifade etmektedir.

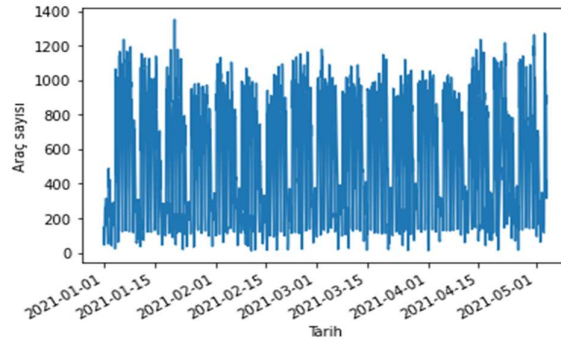
Verisetinde 2321 farklı noktaya ait trafik yoğunluk bilgileri bulunmaktadır. Trafik yoğunluęu ve hız tahmini yapılacak noktayı belirlemek için en yüksek araç sayısına sahip olan konum seçilmiřtir. Enlem ve boylam deęerlerine göre bu konumun Bařakřehir Mahmutbey kavřaęı olduęu görölmüř ve bu konum deneysel çalıřmalarda kullanılmak üzere seçilmiřtir. Verisetinde, Bařakřehir Mahmutbey Kavřaęı'na

ait Ocak 2021-Mayıs 2021 tarihleri arasında bir saat aralıklarla kaydedilen 2900 satır araç verisi bulunmaktadır. Başakşehir Mahmutbey Kavşağı, TEM otoyolunun İstanbul'a batıdan giriş ve çıkışında önemli bir rol oynayan kilit bir kavşaktır. Bu kavşak, Başakşehir, İkitelli Organize Sanayi Bölgesi, İstoç, Bağcılar, Halkalı, Basın ekspres yolu, TEM otoyolu ve Kuzey Marmara Otoyolunu birbirine bağlamaktadır. Bu sebeple İstanbul'daki en yüksek trafik yoğunluğuna sahip kavşak durumundadır. Şekil 2'de 18 Haziran 2022 saat 15:44 için Google Maps üzerinden alınan Başakşehir Mahmutbey kavşağının bilgileri görülmektedir.



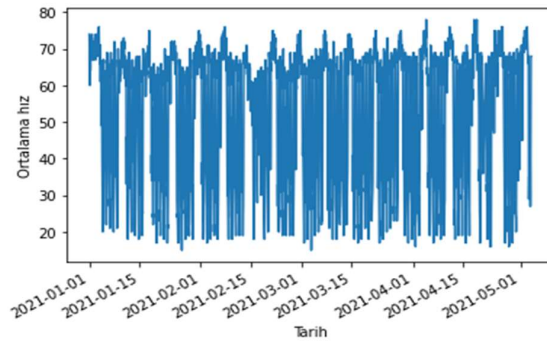
Şekil 2. Başakşehir Mahmutbey kavşağı.

Şekil 3'te Başakşehir Mahmutbey kavşağından saat başına geçen araç sayıları görülmektedir.



Şekil 3. Kavşak noktasından saat başına geçen araç sayıları.

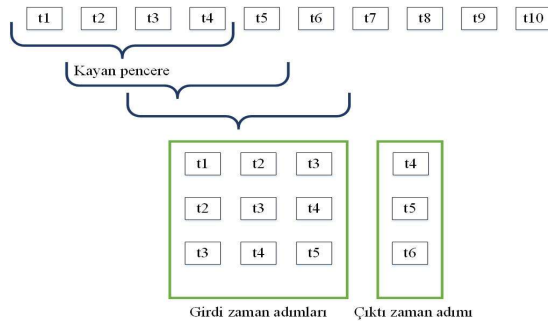
Şekil 4'te ise Başakşehir Mahmutbey kavşağından saat başına geçen araçların ortalama hızları görülmektedir.



Şekil 4. Kavşak noktasından saat başına geçen araçların ortalama hızları.

## B. VERİ ÖN-İŞLEME

Bu çalışmada, kullanılan veriseti zaman serisi verilerinden oluşmaktadır. Zaman serisi verileri, zaman sırasına göre indekslenen veri noktası dizileridir. Bu veri noktaları tipik olarak aynı kaynaktan bir zaman aralığında yapılan ardışık ölçümlerden oluşur ve zaman içindeki değişimi izlemek için kullanılır. Zaman serisi verilerinin makine öğrenmesi modellerinin işleyebileceği denetimli öğrenme problemi olarak yapılandırılması gereklidir. Zaman serisi verileri kayan pencere yöntemi kullanılarak denetimli öğrenme problemi yapısına dönüştürülebilir. Kayan pencere yöntemi, gelecek gözlem değerlerinin, geçmiş gözlem değerleri kullanılarak tahmin edilmesi temeline dayanmaktadır. Kayan pencerenin büyüklüğü, pencere içerisine yerleştirilecek zaman adımlarının sayısına göre belirlenmektedir. Bu çalışmada, yapılan deneysel çalışmaların sonucuna göre kayan pencere boyutu 3 olarak seçilmiştir. Şekil 5'te görüldüğü gibi  $t_3$ ,  $t_2$  ve  $t_1$  zaman adımlarındaki veriler kayan pencerenin girdisi,  $t$  zaman adımındaki veri ise kayan pencerenin çıktısı olacak şekilde zaman serisi verileri denetimli öğrenme problemine dönüştürülmüştür.



Şekil 5. Kayan pencere yöntemi.

Veriler denetimli öğrenme formatına dönüştürüldükten sonra Scikit Learn kütüphanesinden MinMaxScaler metodu kullanılarak 0-1 aralığında normalize edilmiştir. Normalizasyon adımından sonra veriler eğitim, test ve doğrulama kümelerine ayrılmıştır. Yapılan deneysel çalışmalar sonucunda en yüksek tahmin doğruluğuna %80 eğitim ve %20 test kombinasyonunda erişildiği için bu değerler deneysel çalışmalarda kullanılmak üzere seçilmiştir. Eğitim verilerinin %10'u doğrulama için ayrılmıştır. Doğrulama verileri uygulanan modellerin parametrelerinin optimizasyonu için kullanılmıştır. Uygulanan modellerin en iyi tahmin sonuçlarını elde edebilmesi için Scikit Learn kütüphanesinden GridSearchCV metodu kullanılarak model parametreleri optimize edilmiştir. Grid SearchCV, belirtilen tüm hiperparametrelerin ve değerlerinin farklı bir kombinasyonunu kullanır ve her bir kombinasyon için performansı hesaplayarak hiperparametreler için en iyi değeri seçer.

## C. DEĞERLENDİRME METRİKLERİ

Regresyon problemlerinde yapılan tahminler ile gerçek değerler arasındaki hata oranını belirleyebilmek için temel olarak Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) ve R Squared ( $R^2$ ) metrikleri kullanılmaktadır. MSE, gözlenen ve tahmin edilen değerler arasındaki ortalama kare farkını değerlendirir. Bir modelde hata olmadığında, MSE sifıra eşittir. Model hatası arttıkça değeri de artar. MSE,  $y$  gerçek değerleri, tahmin edilen değerleri ve  $n$  ise örnek sayısını ifade etmek üzere Eş.1 kullanılarak hesaplanır.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y - \hat{y})^2 \quad (1)$$

RMSE, tahmin hataların standart sapmasını hesaplamaktadır. RMSE, tahmin hatalarının regresyon çizgisi veri noktalarından ne kadar uzakta olduğunu bir ölçüsüdür RMSE, MSE'nin karekökü alınarak Eş.2'de görüldüğü gibi hesaplanır.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y - \hat{y})^2} \quad (2)$$

MAE, gözlenen değerler ile tahmin edilen değerler arasındaki farkların mutlak değerlerinin ortalamasını ifade eder. Tahmin hataların ortalamasını hesaplamaktadır. MAE, Eş.3 kullanılarak hesaplanmaktadır.

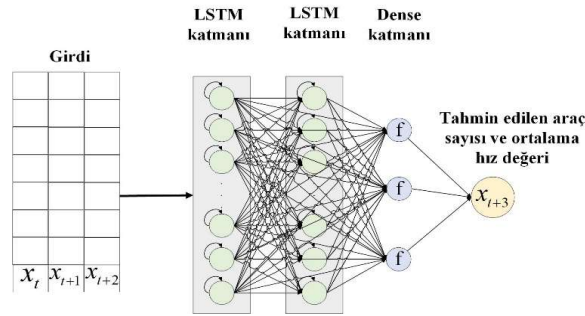
$$MAE = \frac{1}{n} \sum_{i=1}^n |y - \hat{y}| \quad (3)$$

$R^2$ , bir regresyon modelindeki değişkenler tarafından açıklanan bir bağımlı değişken için varyansın oranını temsil eden istatistiksel bir ölçüdür.  $R^2$ , doğrusal regresyon modelleri için bir uyum iyiliği ölçüsüdür.  $R^2$ , tahmin edilen  $y$  değerlerini, ise  $y$  değerlerinin ortalamasını ifade etmek üzere Eş.4 kullanılarak hesaplanmaktadır.

$$R^2 = \frac{\sum (y - \hat{y})^2}{(y - \bar{y})^2} \quad (4)$$

#### D. GELİŞTİRİLEN LSTM TABANLI DERİN ÖĞRENME MODELİ

Bu çalışmada geliştirilen LSTM tabanlı derin öğrenme modeli, kavşaktan geçen saatlik araç sayısı ve araçların ortalama hız değerlerini girdi olarak almakta, çıktı olarak araç sayısı ve ortalama hız tahminleri üretmektedir. Geliştirilen LSTM tabanlı modelin mimarisi Şekil 6'da görülmektedir.



Şekil 6. Geliştirilen LSTM tabanlı modelin mimarisi.

Bu çalışmada, geliştirilen modelin parametrelerinin optimize edilmesi amaçlanmıştır. Denetimli öğrenme problemi yapısına dönüştürülen zaman serisi verileri LSTM'e girdi olarak sunulmuştur. Parametre analizi çalışmalarıyla katman sayısı, nöron sayısı, eğitim sayısı ve yığın boyutu gibi parametreler ile en yüksek tahmin doğruluğuna ulaşmak amaçlanmıştır. Optimize edici olarak Adam kullanılmıştır. Geliştirilen çok katmanlı LSTM mimarisinde parametrelerin optimizasyonu için GridSearchCV kütüphanesi kullanılmıştır. Katman sayısının ve gizli katmanlarda bulunan nöron sayılarının belirlenmesi için deneysel çalışmalar yapılmıştır. 2 gizli katman ve 128 nörondan oluşan bir ağ yapısı oluşturulmuştur. En yüksek tahmin doğruluğuna sahip parametreler belirlendikten sonra model oluşturulmuştur.



## **IV. DENEYSEL SONUÇLAR**

Bu çalışmada, İstanbul'daki en yoğun kavşak noktalarından biri olan Başakşehir Mahmutbey kavşağından saatlik olarak geçen araç sayısı ve geçen araçların ortalama hızlarının tahminine yönelik LR, RF, SVM, MLP, CNN ve RNN'in geliştirilen LSTM tabanlı derin öğrenme modeliyle karşılaştırmalı bir analizi sunulmuştur. Uygulanan her bir algoritma ve model için MSE, RMSE, MAE ve  $R^2$  metriklerine göre elde edilen sonuçlar karşılaştırmalı olarak analiz edilmiştir. Karşılaştırılan modellerin en iyi sonuçları elde edebilmeleri için GridSearchCV kullanılarak modellerin parametreleri analiz edilmiştir. RF için bootstrap=True, max\_depth=70, max\_features=auto, min\_samples\_leaf=4, min\_samples\_split=10 ve n\_estimators=400 olarak seçilmiştir. SVM için C=10, gamma=1e-07, epsilon=0.1 ve kernel=linear olarak seçilmiştir. MLP için input\_size=24, nodes=100, epochs=100 ve batch\_size=1 olarak seçilmiştir. CNN için input\_size=12, filters=64, kernel\_size=5, epochs=100, batch\_size=1 ve activation=relu olarak seçilmiştir. RNN için input\_size=24, nodes=64, epochs=50, batch\_size=16 ve optimizer=adam olarak seçilmiştir. LSTM için input\_size=24, nodes=128, epochs=100, batch\_size=32 ve optimizer=adam olarak seçilmiştir.

Doğrulama verileri kullanılarak parametre optimizasyonu yoluyla en uygun parametreler belirlendikten sonra modeller oluşturulmuştur. Zaman serisi modelleme problemlerinde, tahmin edilen tarih ilerledikçe yapılan tahminler hataya meyilli olma eğilimindedir. Bu nedenle, modeli gelecekteki tahminler için gerçek verilerle yeniden eğitmek ve eşzamanlı olarak doğrulamak gerçekçi bir yaklaşım olacaktır. Bu amaçla ileri yürüyüş doğrulaması (Walk forward validation) tekniği kullanılarak tahmin doğruluğunun artırılması hedeflenmiştir. Zaman serisinin başlangıcından itibaren, bir modeli eğitmek için penceredeki minimum örnek sayısı kullanılır. Model, bir sonraki zaman adımı için bir tahminde bulunur. Tahmin edilen değer, gerçek değere ile karşılaştırılır. İleri yürüyüş doğrulamasında kayan pencere, gerçek değeri içerecek şekilde genişletilir ve bu süreç tekrarlanır.

Tablo 1'de kavşaktan saatlik olarak geçen araç sayısı tahmine yönelik karşılaştırmalı deneysel sonuçlar görülmektedir.

*Tablo 1. Araç sayısı tahminine yönelik deneysel sonuçlar.*

<b>Modeller</b>	<b>MSE</b>	<b>RMSE</b>	<b>MAE</b>	<b>R<sup>2</sup></b>
LR	18081,293	134,466	94,807	0,824
RF	13009,035	114,057	82,962	0,873
SVM	18795,840	137,097	90,415	0,817
MLP	12697,998	112,685	82,680	0,876
CNN	13093,633	114,427	82,772	0,872
RNN	12554,149	112,045	80,907	0,878
LSTM	<b>10545,670</b>	<b>102,692</b>	<b>71,116</b>	<b>0,897</b>

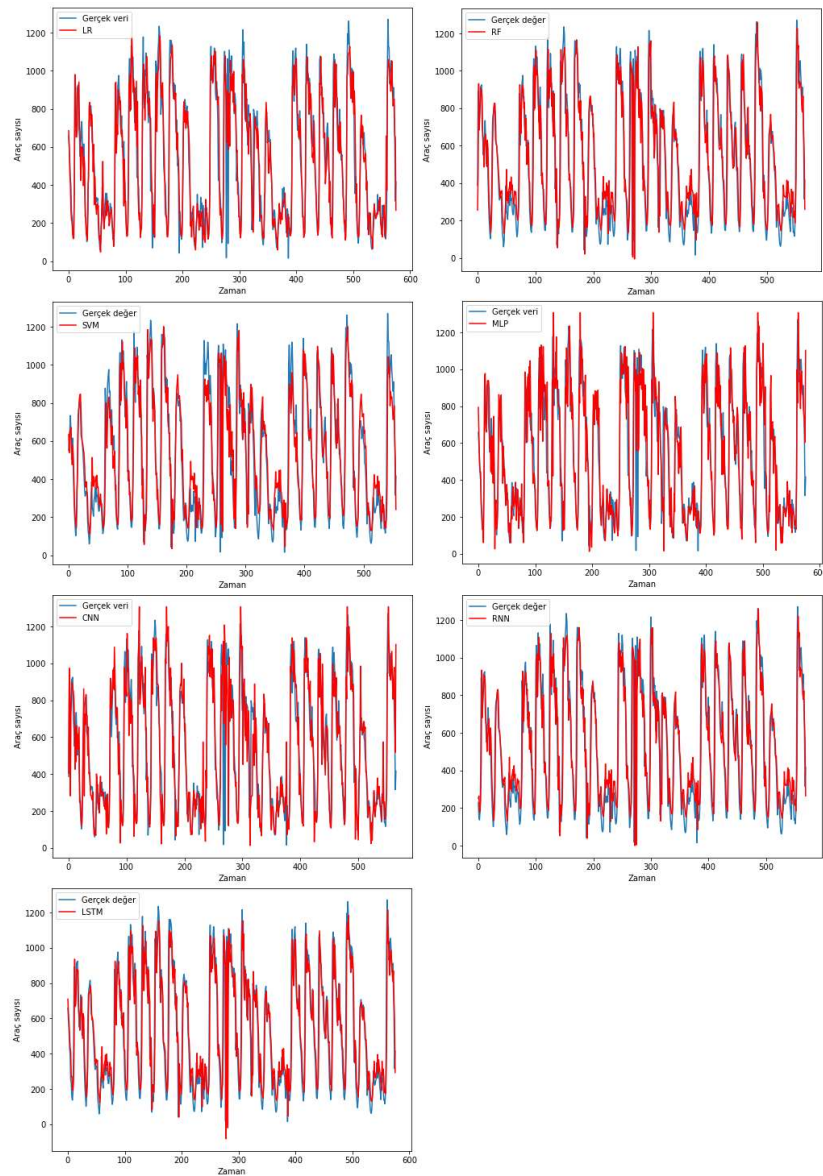
Kavşaktan geçen araç sayısı tahminine yönelik deneysel sonuçlar LSTM'in karşılaştırılan modellerden daha başarılı olduğunu göstermiştir. LSTM'in ardından RNN, MLP, RF, CNN, LR ve SVM sırasıyla başarılı olmuştur. LSTM, kavşaktan geçen araç sayısı tahmininde 71,116 MAE ve 0,897  $R^2$  değerine sahip olmuştur.

Tablo 2'de kavşaktan geçen araçların ortalama hızlarının tahmine yönelik karşılaştırmalı deneysel sonuçlar görülmektedir.

**Tablo 2.** Araçların ortalama hızlarının tahminine yönelik deneysel sonuçlar.

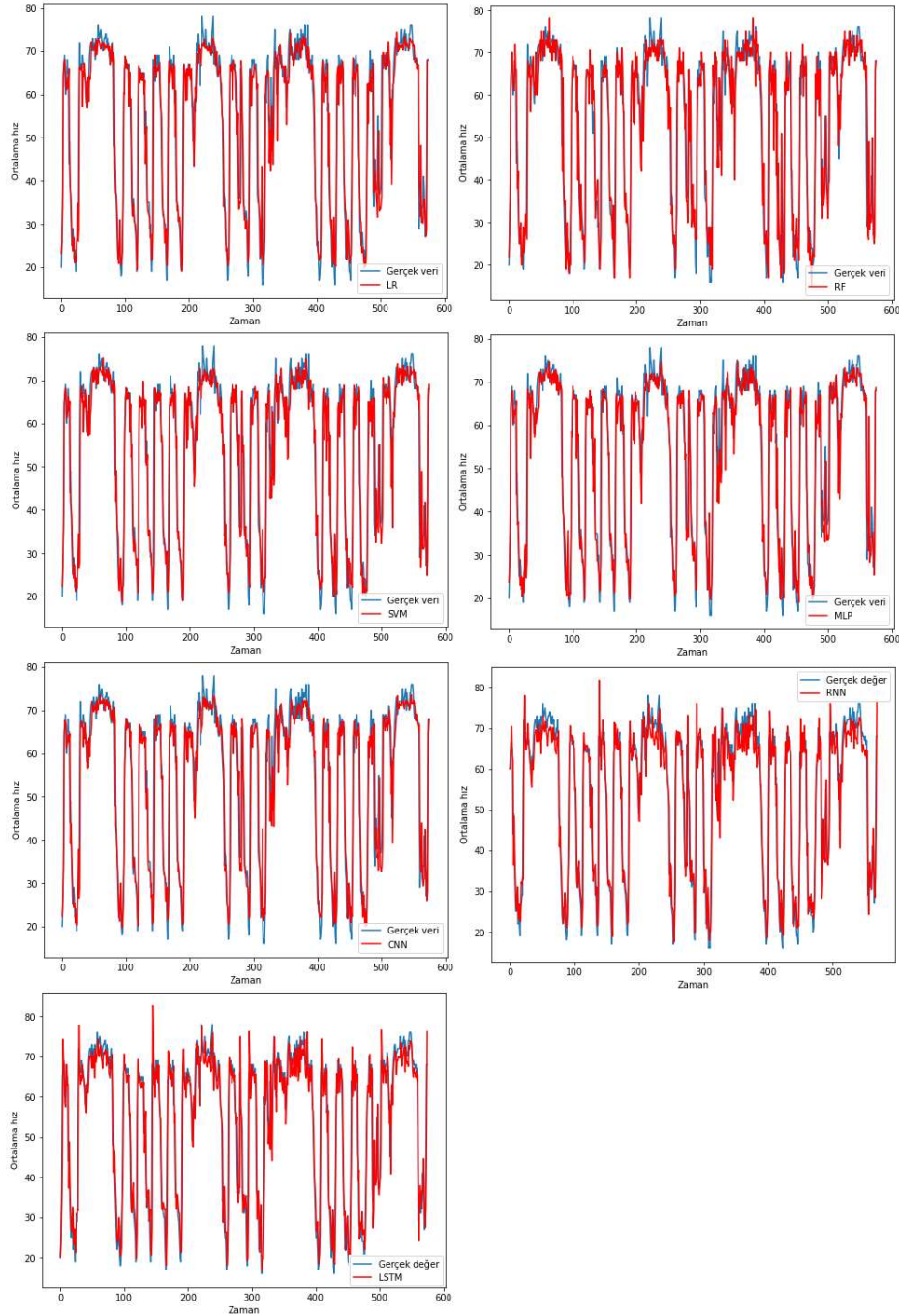
Modeller	MSE	RMSE	MAE	R <sup>2</sup>
LR	46,980	6,854	4,743	0,858
RF	44,543	6,674	4,362	0,866
SVM	47,446	6,888	4,501	0,857
MLP	42,469	6,517	4,423	0,872
CNN	44,806	6,694	4,511	0,865
RNN	44,310	6,657	4,321	0,866
LSTM	<b>38,857</b>	<b>6,234</b>	<b>3,989</b>	<b>0,883</b>

Kavşaktan geçen araçların ortalama hızlarının tahminine yönelik deneysel sonuçlar LSTM'in karşılaştırılan modellerden daha başarılı olduğunu göstermiştir. LSTM'in ardından MLP, MLP, RNN, RF, LR ve SVM sırasıyla başarılı olmuştur. Şekil 7'de araç sayısı tahmini için uygulanan modellerin test veriseti üzerindeki tahmin grafikleri görülmektedir.



**Şekil 7.** Araç sayısı tahmini için modellerin tahmin grafikleri

Şekil 8'de ortalama hız tahmini için uygulanan modellerin test veriseti üzerindeki tahmin grafikleri görülmektedir.



Şekil 8. Ortalama hız tahmini için modellerin tahmin grafikleri.

Şekil 7 ve Şekil 8'de görüldüğü gibi geliştirilen model, araç sayısı ve ortalama hız değerlerinde görülen dalgalanmaları diğer modellere göre daha başarılı bir şekilde tahmin etmiştir.

Tahmin hatalarının ortalamasını ifade eden MAE metriği, araç sayısı tahmininde yaklaşık 71 araçlık hata değeriyle tahminlerin yapıldığını göstermektedir. Araçların ortalama hızlarının tahminine yönelik

deneysel sonuçlar ise LSTM'in yaklaşık 4 MAE değerine sahip olduğunu göstermiştir. Bu değerler, Başakşehir Mahmutbey Kavşağı gibi yoğun bir nokta için oldukça başarılıdır.  $R^2$  değeri ise modelin verisetine sağladığı uyumun ölçüsüdür.  $R^2$  değeri araç sayısı ve ortalama hız tahmininde yaklaşık olarak 0,9 değerine sahip olmuştur. Bu sonuçlar, geliştirilen modelin verisetine oldukça iyi uyum sağladığını göstermektedir.

## **V. SONUÇLAR**

Dünyanın birçok kentinde yaşanan trafik sıkışıklığı problemi, kentsel yaşam kalitesi üzerinde bir tehdit oluşturmaktadır. Trafik sıkışıklığı yolculuk sürelerinde, yakıt tüketiminde, işletme maliyetlerinde ve çevre kirliliğinde artışa neden olmaktadır. Günümüzde giderek artan araç sayılarıyla birlikte, sürdürülebilir mobilite ve trafik durumu tahmini, kentsel alanlardaki trafik sıkışıklığını azaltmada kilit bir rol oynamaktadır. Trafik akışı tahmini, akıllı ulaşım sisteminin önemli bir parçasıdır. Trafik akışı tahmini, gelecekte belirli bir zaman aralığında belirli bir bölgedeki trafik akışının doğru bir şekilde tahmin edilmesini ifade etmektedir.

Gelişen teknolojilerle birlikte, şehirlerdeki çeşitli noktalara yerleştirilen sensörler vasıtasıyla trafik gözlemleri yapılmaktadır. Anlık olarak elde edilen araç verileri kullanılarak araç sayısı ve hız gibi trafik planlamasına yönelik bilgiler elde edilmektedir. Bilgi ve iletişim teknolojilerinin trafik planlaması alanında kullanımı, trafik verilerini akıllı ulaşım sistemi uygulamaları için önemli bir gerçek zamanlı veri kaynağı haline getirmektedir.

Bu çalışmada, İstanbul'daki önemli kavşak noktalarından biri olan Başakşehir Mahmutbey kavşağından saatlik olarak geçen araç sayısı ve geçen araçların ortalama hızlarının tahminine yönelik LR, RF, SVM, MLP, CNN ve RNN'in geliştirilen LSTM tabanlı derin öğrenme modeliyle karşılaştırmalı bir analizi sunulmuştur. Uygulanan her bir algoritma ve model için MSE, RMSE, MAE ve  $R^2$  metriklerine göre elde edilen sonuçlar karşılaştırmalı olarak analiz edilmiştir. Deneysel sonuçlar, geliştirilen modelin karşılaştırılan modellerden daha yüksek bir tahmin doğruluğuna sahip olduğunu göstermiştir.

Kavşaktan geçen araç sayısı tahminine yönelik deneysel sonuçlar geliştirilen LSTM tabanlı modelin 10545,670 MSE, 102,692 RMSE, 71,116 MAE ve 0,897  $R^2$  değerine sahip olduğunu göstermiştir. Kavşaktan geçen araçların ortalama hızlarının tahminine yönelik deneysel sonuçlar geliştirilen LSTM tabanlı modelin 38,857 MSE, 6,234 RMSE, 3,989 MAE ve 0,883  $R^2$  değerine sahip olduğunu göstermiştir.

LSTM'in karşılaştırılan modellerden daha başarılı olması LSTM'in geri beslemeli yapısı ile açıklanabilir. Bu yapı geçmiş özelliklerin hatırlanmasını ve ağa tekrar girdi olarak sunulmasını sağlayarak daha başarılı bir tahmin performansına sahip olunmasını sağlamıştır. RNN'in MLP ve CNN'den daha başarılı olması RNN'in tekrarlı yapısı ile açıklanabilir. RNN, önceden belirlenmiş herhangi bir boyut sınırı olmaksızın bir dizi girdi alacak şekilde tasarlanmıştır. RNN, dizideki önceki değerini veya durumun belleğini koruyarak dizileri işlemektedir. Bu sayede, RNN'de mevcut adımın çıkışı bir sonraki adımın girişi olmaktadır. Model her aşamada hem mevcut girdiyi hem de önceki tüm çıktıları dikkate almaktadır. RF'in SVM'den daha başarılı olmasının nedeni veri setinin yapısı ile yorumlanabilir. RF, özellikler çeşitli ölçeklerde olduğunda avantajlıdır. Bu durum, RF'nin verileri olduğu gibi kullanmasına izin verir. SVM ise farklı noktalar arasındaki marjı maksimize eder ve noktalar arasındaki mesafeyi hesaplar. SVM'in LR ile benzer sonuçlara sahip olması, verisetindeki öznitelliklerin yapısı ile açıklanabilir. LR, verisetinin boyutu küçük olduğunda ve birçok özelliğe sahip olduğunda SVM'den daha iyi performans gösterebilir. Kategorik veriler için SVM, LR'den daha başarılıdır.

Trafik akışının hacmini ve yoğunluğunu tahmin etmek, araç hareketlerini yönetmek, trafik sıkışıklığını azaltmak ile en az zaman ve yakıt tüketen rotayı oluşturmak amacıyla önemlidir. Trafik tahmini problemi ulusal ve yerel yönetimler ile ulaşım şirketleri açısından da oldukça önemlidir. Bu sebeple anlık olarak üretilen veri miktarlarının artışıyla elde edilen farklı çeşitlilikteki trafik verileri, yapay zekâ yöntemleri kullanılarak analiz edilmektedir. Bu amaç doğrultusunda gerçekleştirilen bu çalışmada elde

edilen sonuçlar, gerçek dünya uygulamalarına adapte edilerek trafik yönetimi konusunda etkili planlamalar yapılabilir. Gelecek çalışmalarda, veriseti genişletilerek İstanbul'daki trafik yoğunluğu yüksek diğer noktalar için de analizler yapılması planlanmaktadır. Hibrit bir derin öğrenme modeli geliştirilerek araç sayısı ve ortalama hız tahmininde daha başarılı sonuçların elde edilmesi amaçlanmaktadır.

## **V. KAYNAKLAR**

- [1] M. Shahidehpour, Z. Li, and M. Ganji, "Smart cities for a sustainable urbanization: Illuminating the need for establishing smart urban infrastructures," *IEEE Electrification Magazine*, vol. 6, no. 2, pp. 16-33, 2018.
- [2] A. Sumalee, and H. W. Ho, "Smarter and more connected: Future intelligent transportation system," *Iatss Research*, vol. 42, no. 2, pp. 67-71, 2018.
- [3] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, and B. Yin, "Deep learning on traffic prediction: Methods, analysis and future directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4927 – 4943, 2021.
- [4] C. Benevolo, R. P. Dameri, and B. D'auria, "Smart mobility in smart city. In Empowering organizations," *Empowering Organizations*, 2016, pp. 13-28.
- [5] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson, "Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones," *In Proceedings of the 7th ACM conference on embedded networked sensor systems*, 2009, pp. 85-98.
- [6] E. Ozus, S. S.Turk, and V. Dokmeci, "Urban restructuring of Istanbul," *European Planning Studies*, vol. 19, no. 2, pp. 331-356, 2011.
- [7] NTV HABER. (2022, 25 Haziran). İstanbul trafiğinde yoğunluk [Çevrimiçi]. Erişim: [https://www.ntv.com.tr/turkiye/istanbul-trafiginde-yogunluk,p7VKP\\_LQsU21SJcUNIOYUA](https://www.ntv.com.tr/turkiye/istanbul-trafiginde-yogunluk,p7VKP_LQsU21SJcUNIOYUA)
- [8] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," *In 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 2016, pp. 324-328.
- [9] Y. Liu, H. Zheng, X. Feng, and Z. Chen, "Short-term traffic flow prediction with Conv-LSTM," *In 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)*, 2017, pp. 1-6.
- [10] Z. Duan, Y. Yang, K. Zhang, Y. Ni, S. Bajgain, "Improved deep hybrid networks for urban traffic flow prediction using trajectory data," *Ieee Access*, vol. 6, pp. 31820-31827, 2018.
- [11] F. Lin, Y. Xu, Y. Yang, and H. Ma, "A spatial-temporal hybrid model for short-term traffic prediction," *Mathematical Problems in Engineering*, vol. 19, 2019.
- [12] O. Mohammed, and J. Kianfar, "A machine learning approach to short-term traffic flow prediction: A case study of interstate 64 in Missouri," *In 2018 IEEE International Smart Cities Conference (ISC2)*, 2018, pp. 1-7.
- [13] W. Zhang, Y. Yu, Y. Qi, F. Shu, and Y. Wang, "Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning," *Transportmetrica A: Transport Science*, vol. 15, no. 2, pp. 1688-1711, 2019.

- [14] C. Bratsas, K. Koupidis, J. M. Salanova, K. Giannakopoulos, A. Kaloudis, and G. Aifadopoulou, "A comparison of machine learning methods for the prediction of traffic speed in urban places," *Sustainability*, vol. 12, no. 1, 2019.
- [15] Taş, İ. Ç., and Müngen, A. A. "Yapay Sinir Ağları Ve Destek Vektör Makineleri Yöntemleri İle Bölgesel Trafik Yoğunluk Tahmini," *Adıyaman Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 8, no. 15, pp. 378-390, 2021.
- [16] R. Zhu, X. Hu, J. Hou, and X. Li, "Application of machine learning techniques for predicting the consequences of construction accidents in China," *Process Safety and Environmental Protection*, vol. 145, pp. 293-302, 2021.
- [17] N. Fumo, and M. R. Biswas, "Regression analysis for prediction of residential energy consumption," *Renewable and sustainable energy reviews*, vol. 47, pp. 332-343, 2015.
- [18] J. K. Jaiswal, and R. Samikannu, "Application of random forest algorithm on feature subset selection and classification and regression," *In 2017 world congress on computing and communication technologies (WCCCT)*, 2017, pp. 65-68.
- [19] Q. Zou, K. Qu, Y. Luo, D. Yin, Y. Ju, and H. Tang, "Predicting diabetes mellitus with machine learning techniques," *Frontiers in genetics*, vol. 9, 2018.
- [20] Z. K. Şentürk, and N. Çekiç, "A machine learning based early diagnosis system for mesothelioma disease," *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, c. 8, s. 2, ss. 1604-1611, 2020.
- [21] L. Piyathilaka, and S. Kodagoda, "Affordance-map: Mapping human context in 3d scenes using cost-sensitive svm and virtual human models," *In 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2015, pp. 2035-2040.
- [22] D. H. Lee, Y. T. Kim, and S. R. Lee, "Shallow landslide susceptibility models based on artificial neural networks considering the factor selection method and various non-linear activation functions," *Remote Sensing*, vol. 12, no. 7, 2020.
- [23] A. İ. Taş, P. Gülüm, and G. Tulum, "Finansal Piyasalarda Hisse Fiyatlarının Derin Öğrenme ve Yapay Sinir Ağı Yöntemleri ile Tahmin Edilmesi; S&P 500 Endeksi Örneği," *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, c. 9, s. 3, ss. 446-460, 2021.
- [24] X. Wan, H. Song, L. Luo, Z. Li, G. Sheng, and X. Jiang, "Pattern recognition of partial discharge image based on one-dimensional convolutional neural network," *In 2018 Condition Monitoring and Diagnosis (CMD)*, 2018, pp. 1-4.
- [25] M. Volpi, and D. Tuia, "Dense semantic labeling of subdecimeter resolution images with convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 2, pp. 881-893, 2016.
- [26] Y. Tian, and L. Pan, "Predicting short-term traffic flow by long short-term memory recurrent neural network," *In 2015 IEEE international conference on smart city/SocialCom/SustainCom (SmartCity)*, 2015, pp. 153-158.
- [27] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3639-3655, 2017.

- [28] P. Bahad, P. Saxena, and R. Kamal, "Fake news detection using bi-directional LSTM-recurrent neural network," *Procedia Computer Science*, vol. 165, pp. 74-82, 2019.
- [29] S. Patil, V. M. Mudaliar, P. Kamat, S. Gite, "LSTM based Ensemble Network to enhance the learning of long-term dependencies in chatbot," *International Journal for Simulation and Multidisciplinary Design Optimization*, vol. 11, no. 25, 2020.
- [30] İstanbul Büyükşehir Belediyesi. (2022, 25 Mayıs). Saatlik Trafik Yoğunluk Veri Seti [Çevrimiçi]. Erişim: <https://data.ibb.gov.tr/dataset/hourly-traffic-density-data-set>