

TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



CNN YÖNTEMİ KULLANILARAK TRAFİK AKIŞ HIZININ
TAHMİN EDİLMESİ

16011087 – Caner Kaya

BİLGİSAYAR PROJESİ

Danışman
Dr. Öğr. Üyesi Mehmet Amaç GÜVENSAN

Ocak, 2020

TEŞEKKÜR

Proje boyunca değerli fikirleriyle ve katkılarıyla yanımda olan danışman hocam Dr.Öğr.Üyesi Mehmet Amaç GÜVENSAN' a ve Dr.Öğr.Üyesi Hafıza İrem TÜRKMEN ÇİLİNGİR hocama, moral ve desteklerini hiçbir zaman esirgemeyen anneme ve arkadaşlarıma teşekkürlerimi sunarım.

Caner Kaya

İÇİNDEKİLER

KISALTMA LİSTESİ	v
ŞEKİL LİSTESİ	vi
TABLO LİSTESİ	vii
ÖZET	viii
ABSTRACT	ix
1 Giriş	1
2 Ön İnceleme	2
2.1 Mevcut Uygulamalar	2
2.1.1 Google Uygulamaları	2
2.1.2 Yandex Uygulamaları	3
2.1.3 İBB CepTrafik	3
2.2 Projenin Katkıları ve Yol Haritası	3
3 Fizibilite	5
3.1 Teknik Fizibilite	5
3.1.1 Yazılım Fizibilitesi	5
3.1.2 Donanım Fizibilitesi	6
3.2 Zaman Fizibilitesi	6
3.3 Yasal Fizibilite	7
3.4 Ekonomik Fizibilite	7
4 Sistem Analizi	9
4.1 Veri Önışleme	10
4.1.1 Verilerin Tanıtılması	10
4.1.2 Verilerin Ayıklanması	11
4.2 Öğrenme Modelleri	12
4.2.1 Regresyon Modeli	12
4.2.2 2D CNN Modeli	13

4.2.3	1D CNN Modeli	14
4.3	Hata Fonksiyonları	16
5	Sistem Tasarımı	18
5.1	1D CNN Modeli	18
5.1.1	Gün ve Zaman Bilgisinin Eklenmesi	18
5.1.2	Hava Durumu Bilgilerinin Eklenmesi	19
5.1.3	Eğitime Dahil Edilecek Ay Aralığının Belirlenmesi	21
5.2	Veri Tabanı Tasarımı	23
5.3	Harita Üzerinde Gösterim	23
5.3.1	KML Dönüşümü	24
5.3.2	Google API	24
6	Uygulama	25
7	DeneySEL Sonuçlar	27
8	Sonuç	30
	Referanslar	33
	Özgeçmiş	34

KISALTMA LİSTESİ

1D	1 Boyutlu
2D	2 Boyutlu
CNN	Convolutional Neural Network
GB	Giga Byte
KML	Keyhole Markup Language
MB	Mega Byte
mm	MiliMetre
TL	Türk Lirası
vSegDir	Virtual Segment Direction
vSegID	Virtual Segment Identification

ŞEKİL LİSTESİ

Şekil 3.1	Gant diyagramı zaman grafiği	7
Şekil 3.2	Başabaş Noktası Grafiği	8
Şekil 4.1	Kullanım Senaryosu Diyagramı	9
Şekil 4.2	UML Sınıf Diyagramı	10
Şekil 4.3	Veritabanı sütun isimleri	10
Şekil 4.4	Matris Formatına Haritalama	12
Şekil 4.5	Eğitim Örneği	13
Şekil 4.6	Test Örneği	13
Şekil 4.7	2D CNN modeli için verilerin matris formu getirilmesi	14
Şekil 4.8	2D CNN modeli	14
Şekil 4.9	1D CNN modeli için verilerin hazırlanması	15
Şekil 4.10	1D CNN modeli	16
Şekil 5.1	1D CNN modeli için verilerin hazırlanması	19
Şekil 5.2	Hava Durumu Verilerinin Doluluk Grafiği	20
Şekil 5.3	Rüzgarın Yönleri	21
Şekil 5.4	Son 1 Aylık Veriler Kullanılarak Eğitim	21
Şekil 5.5	Son 3 Aylık Veriler Kullanılarak Eğitim	22
Şekil 5.6	Son 6 aylık veriler kullanılarak eğitim	22
Şekil 5.7	Varlık İlişki Diyagramı	23
Şekil 5.8	KML Örneği	24
Şekil 6.1	Harita Üzerinde Segmentlerin Gösterilmesi	25
Şekil 6.2	Segmentler ve Hata Oranları	26
Şekil 6.3	Tahminler ve Gerçek Değerler	26
Şekil 6.4	Renk Ölçeği	26
Şekil 7.1	Son 1 Ay ile Eğitim Hata Yüzdeleri	27
Şekil 7.2	1 - 3 - 6 Ay Eğitim Modellerinin Hata Yüzdeleri	28
Şekil 7.3	3 Aylık Temel ve Yağışlı Modelin Hata Yüzdesi	28
Şekil 7.4	6 Aylık Temel ve Yağışlı Modelin Hata Yüzdesi	29
Şekil 7.5	Tüm Modellerin Ortalama Hata Yüzdesi	29

TABLO LİSTESİ

Tablo 3.1	Keras ile Tensorflow kütüphanesinin karşılaştırılması	6
Tablo 3.2	Donanım Özellikleri	6
Tablo 3.3	Maliyet ve Gelir Tablosu	7
Tablo 4.1	Günün dilimleri	11
Tablo 4.2	Eğitim Hiperparametreleri	14

CNN YÖNTEMİ KULLANILARAK TRAFİK AKIŞ HIZININ TAHMIN EDİLMESİ

Caner Kaya

Bilgisayar Mühendisliği Bölümü
Bilgisayar Projesi

Danışman: Dr. Öğr. Üyesi Mehmet Amaç GÜVENSAN

Trafik, nüfus yoğunluğunun fazla olduğu metropol şehirlerin en önemli sorunlarından biridir. Bu sorunla başa çıkmanın yolu trafik destek sistemlerinden geçmektedir. Bu sistemler; yollarda bulunan sabit sensörler, kullanıcıların telefonları aracılığıyla elde edilen gps bilgileri, hükümetin trafik kameraları gibi birçok veriyi kullanarak trafik akışını hesaplayabilmekte ve tahmin edebilmektedir. Trafik akış hızının tahmin edilmesi, en uygun rotanın hesaplanmasını sağlayarak zamandan ve enerjiden tasarruf sağlar.

Trafik akış hızının tahmin edilmesinin zorluğu, geleceğe yönelik tahmin yapılırken trafik akış hızının, geçmiş hız verilerinin yanı sıra farklı parametrelerden de etkilenmesidir. Bu parametreler tatiller, mitingler, derbi maçları, hava koşulları vs. olabilir.

Bu projede, Evrimsel Yapay Sinir Ağı modeli kullanılarak İstanbul'un trafik akışı üzerinde tahminler yapılmıştır. Hava durumu bilgisinden faydalanılarak tahminlerin başarısı arttırılmaya çalışılmıştır. Araştırma sırasında rüzgar bilgisi ile trafik akışı arasında bağlantı olduğu görülmüştür. Rüzgar bilgisinin eklenmesiyle ortalama hata oranı %1 azalmıştır. Proje sonucunda bir haftalık zaman aralığında trafik akış hızının %20 - %25 arasında hata oranları ile tahmin edilmesi sağlanmıştır.

Anahtar Kelimeler: trafik akış hızı tahmini, trafik yoğunluğu, trafik ve havadurumu, evrimsel sinir ağları

ABSTRACT

ESTIMATION OF TRAFFIC FLOW SPEED USING CNN METHOD

Caner Kaya

Department of Computer Engineering
Computer Project

Advisor: Assist. Prof. Mehmet Amaç GÜVENSAN

Traffic is one of the most important problems of metropolitan cities where population density is high. The way to deal with this problem is through traffic support systems. These systems are able to calculate and predict the traffic flow using a lot of data, such as fixed sensors on the roads, gps information obtained through users' phones, and government traffic cameras. Estimating the traffic flow speed saves time and energy by calculating the optimal route.

The challenge of predicting the traffic flow speed is that the traffic flow speed is influenced by different parameters as well as historical speed data when making predictions for the future. These parameters may be holidays, rallies, derby matches, weather conditions, etc

In this project, predictions were made about the traffic flow of Istanbul by using the Convolutional Neural Network model. The success of the estimations was tried to be increased by using weather information. During the research, it was seen that there is a connection between wind information and traffic flow. With the addition of wind information, the average error rate decreased by 1%. As a result of the project, it was provided to estimate the traffic flow rate over a period of one week with error rates range from 20% to 25%.

Keywords: estimation of traffic flow speed, traffic volume, traffic and weather, convolutional neural networks

1

Giriş

Şehirlerdeki nüfusun ve araç sayısının artmasıyla trafik, günümüzün popüler sorunlarından biri haline gelmiştir. Trafik yoğunluğu yüzünden insanlar, vakitlerini trafikte boşa geçirmekte ve istedikleri yere vaktinde ulaşamadıkları zaman çeşitli sorunlar yaşamaktadırlar. Güne trafikle başlamak insanların daha mutsuz bir gün geçirmesine sebep olduğu gibi trafik yüzünden acil durumlara müdahâleler de gecikebilmektedir. Trafikte harcanan süre aynı zamanda tüketilen enerji miktarını da olumsuz etkilemektedir. Bu sorunları azaltmak adına trafik yoğunluğuna karşı uygun zamanlamanın yapılması ve en uygun güzergahın belirlenmesi için trafik akış hızını tahmin etmek gereklidir.

Çeşitli sensörler ve gps verileri vs. kullanılarak trafik akış hızını anlık olarak tahmin etmek mümkündür. Gelecek bir zaman için bir tahmin yapılması gerektiğinde anlık veriler tek başına yeterli değildir. Geçmişe dayalı verilerden bir model üretilmek istendiğinde çeşitli istatistiksel yaklaşımlar ve farklı yapay zeka algoritmaları kullanılmaktadır. Bu projede, Makina Öğrenmesi ve Derin Öğrenme modelleri kullanılarak yeni çözüm yöntemleri geliştirilmeye çalışılmıştır. Proje kitabında bu yöntemlerden Polinomsal Regresyon (Polynomial Regression), Evrişimsel Yapay Sinir Ağı (Convolutional Neural Network) nın 1 boyutlu ve 2 boyutlu (1D ve 2D) çeşitleri analiz edilmiştir. Elde edilen sonuçlar ışığında 1D CNN üzerine yazılım tasarımı yapılmıştır. Uygulama kısmında yapılan farklı denemelerin çıktısını görüntülemek için bir arayüz tasarımı yapılmıştır.

2 Ön İnceleme

Trafik problemi yeni bir problem olmadığı için elbette hali hazırda geliştirilen çözümler mevcuttur. Trafik probleminin önemi ve yaygınlığı düşünüldüğünde bu çözümler sektörün önemli aktörleri tarafından sunulmuş oldukça iyi hizmetler içermektedir. Ancak bu hizmetlerin kendisini sürekli güncellemesinden de anlayabiliriz ki hala bir takım eksiklikler ve iyileştirmeler söz konusudur. Bu bölümde bir yerden bir yere varış süresini hesaplamaya yarayan mevcut uygulamaların incelenmesi yapılmış ve ardından yapılabilecek iyileştirmelerden bahsedilmiştir.

2.1 Mevcut Uygulamalar

Bu alanda hizmet vermekte olan uygulamalardan ülkemizde ve İstanbul’ da en çok kullanılan Google, Yandex ve IBB nin hizmet alanları ve sistem sınırları aşağıdaki başlıklarda ele alınmıştır.

2.1.1 Google Uygulamaları

Google 2009 yılında trafik yoğunluğunu tahmin etmek için devlet ve özel şirketlerden sağladığı verilerin yanı sıra google haritalar uygulamasını kullanan kişilerin GPS bilgisini kullanmaya başlamıştır. Bu GPS bilgileriyle kullanıcıların hareket hızlarını ölçmekte ve anlık trafik yoğunluğunu daha tutarlı hesaplayabilmektedir. 2013 yılında Waze uygulamasını satın alarak anlık tahminleme yaparken insan faktörünü ön plana çıkarmayı amaçlamıştır. Bu sistemin tutarlılığı kullanıcı sayısı ve kullanıcıların verdiği geri dönütlerle doğru orantılıdır[1]. Türkiye’de yeterli kullanıcı kitlesine sahip olmadığı için bu uygulama ülkemizde verimli çalışmamaktadır. Google haritalar anlık trafik akışında başarılı olmasına rağmen gelecek bir zaman için tahmin yapılırken 1 hafta veya 2 hafta sonrası farketmeksizin haftanın aynı gününe aynı sonuçları göstermektedir. Buradan çıkarılabilecek sonuç Google tahminleme yaparken haftanın günlük karakteristiği, mevsimsel karakteristik gibi sabit veya fazla değişken olmayan ağırlıklar kullanmaktadır. Bu durum sürekli değişen mevsim normallerine karşı

gittikçe güvenilirliğini kaybeden bir sistem demektir. Örneğin İstanbul için 2019 Yılı Aralık ayı yağışı 95.3 mm olarak gerçekleşmiştir, normali 76.8 mm ve geçen yıl yağış ortalaması 126.7 mm'dir. Yağışlarda normale göre %24 artış meydana gelirken, geçen yıl aralık ayı yağışlarına göre %25 azalma gerçekleşmiştir[2]. Sonuç olarak Google uygulaması yağmurlu günlerde stabil çalışmadığı için trafiğin akışını bilen bir sürücü uygulamayı kullanmak yerine kendi tecrübeleriyle çok daha iyi bir tahminde bulunabilmektedir. Bu geliştirilmesi gereken eksikliklerden biridir.

2.1.2 Yandex Uygulamaları

Yandex 2011 yılından bu yana Türkiye'ye birçok servisi ile beraber hizmet vermektedir. Yandex Haritalar da bunlardan biridir. Trafik verilerini elde edilmesinde Google ile benzer yöntemler kullanmaktadır. Trafik akışının problemli olduğu Rusya, Ukrayna ve Türkiye' nin büyük şehirlerinde yoğunluk tespitine ağırlık verdiğini belirten Yandex, kullanıcılardan aldığı anlık gps verilerine göre trafik akış hızını 0 - 10 arasında ölçeklemektedir. Bu bilgiyi aynı zamanda kullanıcılarıyla anlık paylaşmakta ve gelecek zamanlı tahminlemeler için depolamaktadır[3]. Kullanıcı yorumları dikkate alındığında Yandex' in Google'a kıyasla trafiğin daha az yoğun olduğu bölgelerde daha tutarlı çalıştığı görülmektedir. Google için bahsedilen sorunlar Yandex uygulamaları için de geçerlidir.

2.1.3 İBB CepTrafik

İstanbul Büyükşehir Belediyesi Mühendisleri tarafından geliştirilen 2007 yılından bu yana hizmet sunan bir uygulamadır. Uygulama ana arterler üzerine kurulan Bluetooth sensörlerden toplanan verileri kullanarak gideceğiniz yer için en kısa süre ve en kısa mesafeye göre tahmini seyahat süresini ve alternatifleri göstermektedir[4]. Yandex ve Google uygulamalarına ekstra olarak içerdiği kent genelindeki trafik kameralarını canlı olarak izleme özelliği kullanıcılar tarafından tercih edilme sebeplerinden biridir. Gelecek bir zamandaki trafik durumu ve varış süresi tahmin edilmek istendiğinde yalnızca 1 saat sonrası için sonuçlar gösterilmektedir. Diğer uygulamalar ve kullanıcı ihtiyaçları göz önünde bulundurulduğunda bu süre oldukça dar bir aralıktır.

2.2 Projenin Katkıları ve Yol Haritası

Geçmiş hız verilerini kullanarak basit matematiksel ve istatistiksel formüller yardımıyla trafik akış hızının hesaplanması mümkündür ancak trafik akışını etkileyen farklı durumlar (hava durumu, özel günler, festivaller, gösteriler...) karşısında sağlıklı sonuçlar üretmek oldukça zordur. Google ve Yandex uygulamalarında da bahsedilen

bu özel günlerde yaşanan tahmin problemini aşabilmek adına Hız verilerinin yanı sıra havadurumu bilgileri, gün ve ay bilgileri gibi farklı parametreleride göz önünde bulunduran yeni parametrelerin kolayca eklenip çıkarılabileceği esnek bir sistem tasarlanması planlanmıştır. Trafik akışının tahmin edileceği zaman aralığı konusunda insanların ihtiyaç duyabileceği aralık düşünülmüş ve İBB Cep Trafikten daha geniş bir aralıkta, 1 hafta sonrasına kadar tahmin yapabilen bir sistem uygun görülmüştür.

Bu bölümde projenin genel hatlarını belirlemek adına yapılan fizibilite çalışmasından bahsedilmiştir. Yazılım Fizibilitesi bölümünde kullanılan yazılım dili ve kütüphanelerinin seçilme kriterlerinden, Donanım Fizibilitesi bölümünde sistemin tasarlanması sırasında ve tasarlandıktan sonra ihtiyaç duyulacak donanım kaynaklarından, Zaman Fizibilitesi bölümünde projenin geliştirilmesi için gereken zaman aralıklarından, Yasal Fizibilitede projenin yapılması sırasında alınan izinler ve lisans sınırlamalarına uygunluğundan, Ekonomik fizibilitede sistemin gerçekleşmesi sırasında ve sistemin devamlılığı için yapılan harcamalar ve gelir modelinden bahsedilmiştir.

3.1 Teknik Fizibilite

Sistemin teknik olanakları Yazılım Fizibilitesi ve Donanım Fizibilitesi olmak üzere 2 alt başlıkta incelenmiştir.

3.1.1 Yazılım Fizibilitesi

Proje geliştirilirken kullanılacak programlama dili belirlenirken kullanım kolaylığı, kütüphane kolaylıkları, örnek çeşitliliği gibi artılarından dolayı python programlama dilinin kullanılmasına karar verilmiştir. Derin öğrenme modellerinin gerçekleştirilmesi sırasında kullanılacak kütüphanenin belirlenmesi için Tablo 3.1 üzerinde Keras ve Tensorflow karşılaştırması yapılmıştır. Projenin geliştirilmesi için ayrılan süre kısıtlı olduğu için daha üst seviye bir ortam sunan Tensorflow kütüphanesinin üzerinde çalışan Keras API' yı kullanılmak üzere seçilmiştir.

Tablo 3.1 Keras ile Tensorflow kütüphanesinin karşılaştırılması

Özellik	Keras	Tensorflow
Soyutlama seviyesi	Yüksek	Alt seviyeli işlemlere inilebilir ancak yüksek seviyeli sınıfları da mevcut
Öğrenme Zorluğu	Çok basit	Daha zor, çok fazla detay var
Karmaşık işlem desteği	Basit katman ve hücreler üzerine işlemleri hedef alır	Karmaşık işlemlerin yapılmasına yardımcı olabilecek sınıf ve fonksiyonlar mevcut
Uygunluğu	Hızlı prototipleme	Gelişmiş işlemler

Kullanılan modellerin eğitilmesi sırasında Google Colab bulut servisinden faydalanılmıştır.

Harita modülleri için Google Maps API kullanılmıştır.

3.1.2 Donanım Fizibilitesi

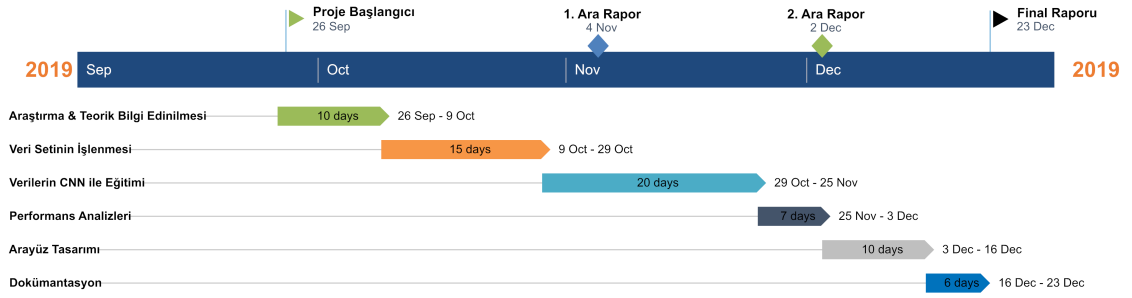
Projenin geliştirilmesinde kullanılan kişisel bilgisayarın teknik özellikleri Tablo 3.2 de görüldüğü gibidir.

Tablo 3.2 Donanım Özellikleri

RAM	12GB
Depolama	1 TB HDD
İşlemci	AMD A8 7410 APU 2.20 GHz
Ekran Kartı	AMD Radeon R5 M430
İşletim Sistemi	Windows 10 Edu. 64 bit

3.2 Zaman Fizibilitesi

Projenin güz döneminde toplam 66 günde gerçekleştirilmesi planlanmıştır. Tek kişilik bir proje olduğu için görev dağılımı yapılmamıştır. Şekil 3.1 de preojede yapılan işlerin zaman grafiği gant diyagramından yararlanılarak gösterilmiştir.



Şekil 3.1 Gant diyagramı zaman grafiği

3.3 Yasal Fizibilite

Trafik hızı verileri İstanbul Büyükşehir Belediyesi Ulaşım Yönetim Merkezi'nden , havadurumu verileri Meteoroloji Genel Müdürlüğü'nden araştırma faaliyetlerinde kullanılmak üzere toplanmış izinleri alınmış verilerdir.

Projede kullanılan kütüphanelerin herhangi bir kısıtlama olmaksızın ücretsiz olarak kullanımına izin verilmiştir.

TensorFlow kütüphanesi Apache License 2.0 [5] ile lisanslıdır.

Keras kütüphanesi The MIT License [6] ile lisanslıdır.

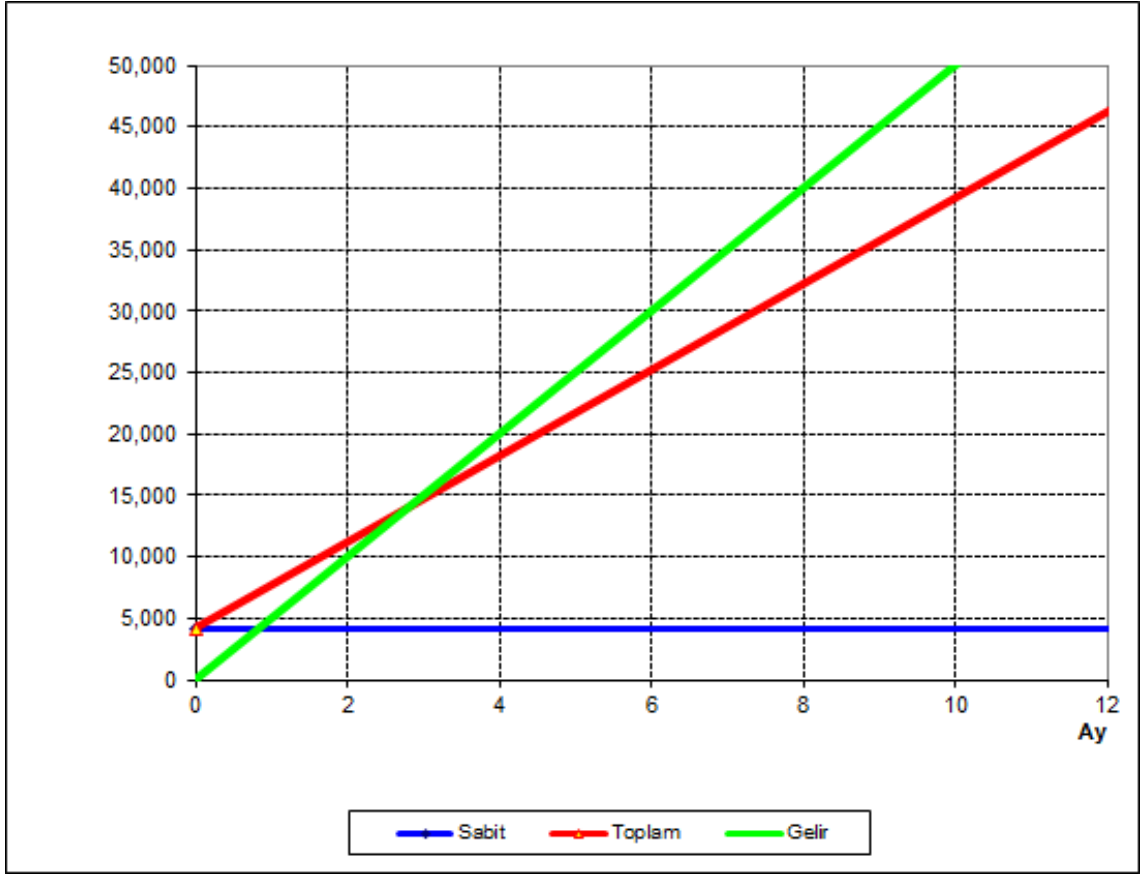
3.4 Ekonomik Fizibilite

Sayfa 7 Tablo 3.3 maliyet ve gelirlerin listesini içermektedir.

Tablo 3.3 Maliyet ve Gelir Tablosu

Sabit Maliyetler	
Donanım	3500 TL
Alan adı	700 TL (10 Yıl)
Aylık Maliyetler	
İş gücü maaliyeti	3000 TL
Sunucu Giderleri	500 TL
Aylık Gelirler	
Reklam ve kullanıcı gelirleri	5000TL

Bu hesaba göre projenin 3. aydan sonra kâra geçebileceği görülmüştür.



Şekil 3.2 Başabaş Noktası Grafiği

4 Sistem Analizi

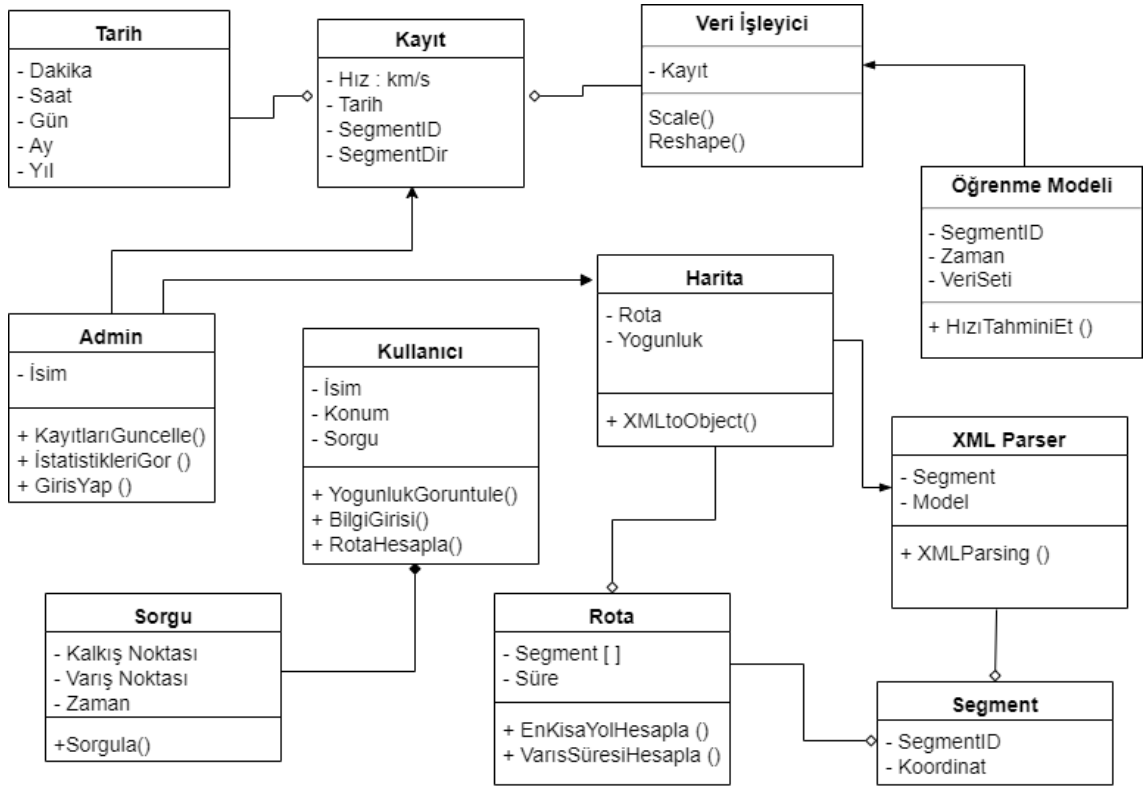
Bu bölümde bir ulaşım uygulaması sistemine ait kullanım senaryosu, gereksinim duyulan alt modüller, sistemin ihtiyaç duyduğu veriler ve alternatif tahmin modelleri analiz edilmiştir.

Sistemin içerdiği aktörler, ve kullanım senaryolarına ilişkin diyagram Şekil 4.1 de görülmektedir.

İhtiyaç duyulabilecek nesneler ve ilişkileri genel hatlarıyla Şekil 4.2 deki kavramsal sınıf diyagramında gösterilmiş olup sistemin tüm ayrıntılarına girilmemiştir.



Şekil 4.1 Kullanım Senaryosu Diyagramı



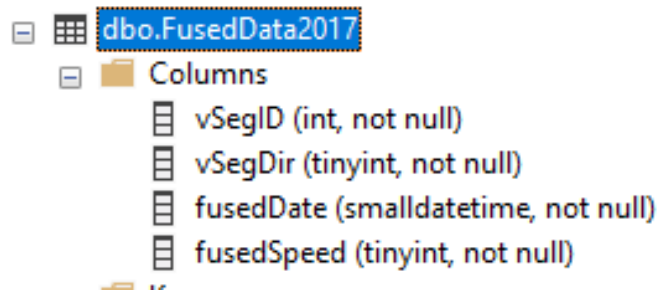
Şekil 4.2 UML Sınıf Diyagramı

4.1 Veri Önışleme

Projenin en önemli kısımlarından biri veri önışlemedir. Bu bölümde ham verilerin tahmin modellerine uygun hale getirilmesi sürecinde gerçekleşen aşamalardan bahsedilmiştir.

4.1.1 Verilerin Tanıtılması

Kullanılan veriler İstanbul Büyükşehir Belediyesi'nden alınan 2017 yılına ait hız verileridir. Bu veriler İstanbul'un genelinde bulunan sensörlerin ölçtüğü araç hızlarından oluşmaktadır. Veri tabanı tablosu Şekil 4.3 te görülen başlıklardan oluşmaktadır.



Şekil 4.3 Veritabanı sütun isimleri

vSegID (Virtual Segment ID) : İstanbul yol ağının ortalama 1000 metre uzunluğa sahip sanal parçalara ayrılması sonucu segmente verilen ID numarasını içerir.

vSegDir (Virtual Segment Direction) : Trafik akış yönüdür, 0 ya da 1 ile gösterilir.

fusedDate : Datetime türünde tarih ve zaman bilgisini içerir. Dakikalık hassasiyete sahiptir.

fusedSpeed : Km cinsinden anlık ölçülen hız ortalamasını içerir.

4.1.2 Verilerin Ayıklanması

Veriler 7096 adet segmente ait bilgiler içermektedir. Ancak her segment yeterli veri yoğunluğu sahip değildir. Her dakikada 1 ölçüm içermesi gereken verilerden %80 doluluğun altında bulunan ve ana trafik akış güzergahının dışında kalan segmentler elenmiştir. Geriye kalan 236 adet segmente ait hız verileri .csv formatında dışarıya aktarılmıştır. Dosyaların toplam boyutu 4.5GB olmuştur. Yapılan hesaplamalar bu ana arterde bulunan segmentler üzerinden gerçekleştirilmiştir.

Bu işlemin ardından yaklaşık 100 milyon verinin satır satır tutulmasının sebep olduğu veri tekrarını önlemek ve verilere erişim hızını arttırmak için bazı işlemler yapılmıştır. Arka arkaya gelen dakikalara ait hız değişimi küçük ölçekte olduğu için her 5 dakikalık hızların ortalaması alınıp günün bir dilimini belirtecek şekilde düzenlenmiştir. Bunun sonucunda 1440 dakika içeren bir gün Tablo 4.1 de görüldüğü gibi 288 (1440/5) dilime bölünmüştür.

Tablo 4.1 Günün dilimleri

Dilim	Saat Aralığı
1	00:00 - 00:05
2	00:05 - 00:10
:	
145	12:00 – 12:05
:	
287	23:50 – 23:55
288	23:55 – 24:00

Bu dilimleme mantığından yararlanılarak her segment için ayrı bir matris yapısı oluşturulmuştur ve her matris segmentIDsiyle aynı isimde olan dosyalara kaydedilmiştir. Şekil 4.4 te bu haritalama işleminin 30 numaralı segment üzerinde gerçekleştirilmesinin bir örneği görülmektedir. Yapılan işlemlerin sonucunda hız bilgilerini içeren toplam dosya boyutu 300 MB a düşmüştür. SegmentIDsi ve zamanı verilen bir hız bilgisine erişilme karmaşıklığı $O(1)$ e indirgenmiştir.

vSegID	vSegDir	fusedDate	fusedSpeed
30	0	2017-01-01 00:00:00	90
30	0	2017-01-01 00:03:00	98
30	0	2017-01-01 00:05:00	98
30	0	2017-01-01 01:08:00	100
30	0	2017-01-01 01:09:00	100
30	0	2017-01-01 01:10:00	100
30	0	2017-01-01 01:11:00	104
30	0	2017-01-01 01:12:00	104
30	0	2017-01-01 01:13:00	105
30	0	2017-01-01 01:15:00	92
30	0	2017-01-01 01:16:00	92
30	0	2017-01-01 01:17:00	68
30	0	2017-01-01 01:18:00	88
30	0	2017-01-01 01:19:00	89
30	0	2017-01-01 01:20:00	89



segmentID	ay	gün	yön	dilim		hız
segment30	[0]	[0]	[0]	[0]	=	94
segment30	[0]	[0]	[0]	[1]	=	98
segment30	[0]	[0]	[0]	[2]	=	0
segment30	[0]	[0]	[0]	[13]	=	100
segment30	[0]	[0]	[0]	[14]	=	103.25
segment30	[0]	[0]	[0]	[15]	=	85.8
segment30	[0]	[0]	[0]	[16]	=	86.8
segment30	[0]	[0]	[0]	[17]	=	88
segment30	[0]	[0]	[0]	[18]	=	96.4

Şekil 4.4 Matris Formatına Haritalama

4.2 Öğrenme Modelleri

Sistemin tahmin modülünde kullanılabilecek olası modeller bu bölümde ele alınmıştır. Sırasıyla Regresyon Modeli, 1D CNN Modeli ve 2D CNN Modeli için analizler yapılmıştır.

4.2.1 Regresyon Modeli

Regresyon, değişkenler arasındaki ilişkiyi modellemenin basit yollarından birisidir. Gün içerisinde zamana göre trafik akış hızının değişiminde doğrusal bir ilişki söz konusu olmadığı için Polynomial Regression (Polinomsal Regresyon) formu kullanılmıştır. Polinomun derecesi için 3 ile 7 arasında farklı parametreler denenmiştir.

Eğitim verisini elde etmek için aşağıdaki adımlar uygulanmıştır.

$$\{V_n | n \text{ hafta önce, günün aynı dilimindeki hız, } V_n \neq 0\} \quad (4.1)$$

$$V_a = \frac{\sum_{n=1}^k V_n}{k}, \text{ Ortalama hız formülü} \quad (4.2)$$

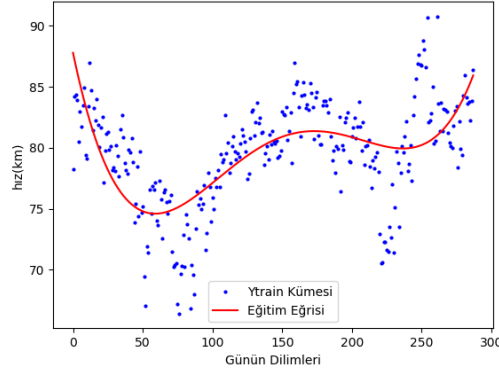
Eşitlik 4.1 ve 4.2 kullanılarak segmentin o tarihten önceki son k haftasına ait hızlarının ortalaması alınmıştır. Hızın '0' olması eksik verinin olduğunu ifade etmektedir. Bu gibi durumlarda '0' ortalamaya dahil edilmemiştir.

Örneğin : $k=3$, $V = \{80, 0, 90\}$ bilgileri için $V_a = \frac{80+90}{2} = 85$ olarak hesaplanmıştır.

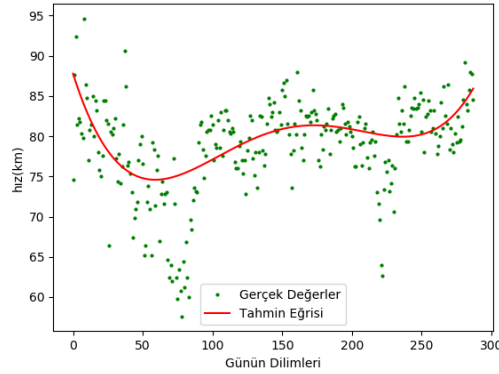
Bu formül tablo 4.1 gösterilen günün tüm dilimleri için uygulandıktan sonra eğitimde kullanılacak veri kümesi elde edilmiştir.

$$Y_{train} = V_{a_1}, V_{a_1}, V_{a_2}, V_{a_3} \dots V_{a_{287}}, V_{a_{288}} \quad (4.3)$$

Eşitlik 4.3 deki Y_{train} noktalarından geçen en uygun polinom hesaplanmış ve gerçek değerler üzerine tahmin yapılmıştır. Bu işleminin grafik üzerinde bir örneği SegmentID:32 Direction:1 ve 30/06/2017 tarihi için şekil 4.5 ve 4.6 de görülmektedir.



Şekil 4.5 Eğitim Örneği



Şekil 4.6 Test Örneği

4.2.2 2D CNN Modeli

Convolutional Neural Networklerin 2 boyutlu konvolusyon matrisi içeren formuna 2D CNN denir. Bu bölümde 2D CNN uygulanarak verilerin eğitilmesi gerçekleştirilmiştir.

$$X_{sc} = \frac{X - X_{min}}{X_{max} - X_{min}}, \text{ Min-Max Scaling} \quad (4.4)$$

Bunun için hız verilerine önce Eşitlik 4.4 de görülen normalizasyon uygulanıp ardından veriler Şekil 4.7 de görülen Matris[k][d] formatına getirilmiştir.

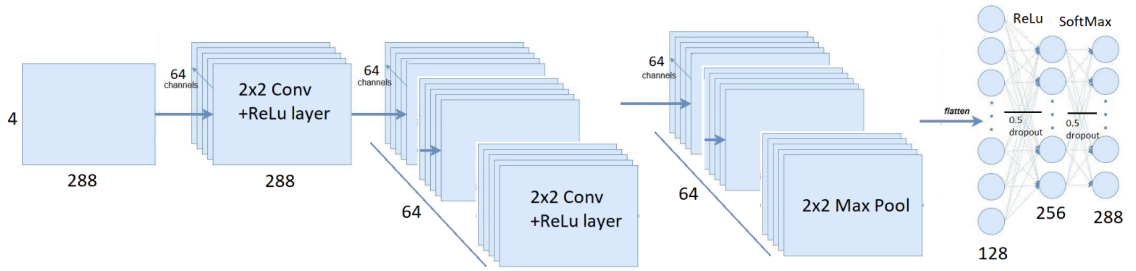
d : günün dilim sayısı

k : kaç hafta öncesinin verileri olduğunu belirtmektedir

V_k : k hafta öncesi için hız verisi

$$\begin{matrix} V_{k_1} & V_{k_2} & V_{k_3} & \dots & V_{k_{287}} & V_{k_{288}} \\ V_{k_1} & & & \ddots & & V_{k_{288}} \\ V_{k_1} & & & & & V_{k_{288}} \\ V_{k_1} & V_{k_2} & V_{k_3} & \dots & V_{k_{287}} & V_{k_{288}} \end{matrix}$$

Şekil 4.7 2D CNN modeli için verilerin matris formu getirilmesi



Şekil 4.8 2D CNN modeli

Eğitim için Tablo 4.2 de tanımlanan parametreler kullanılmıştır.

Modelin katmanları ve diğer parametreleri Şekil 4.8 de gösterilmiştir.

Tablo 4.2 Eğitim Hiperparametreleri

Parametre	Değer
Loss Funntion	mean absolute error
Optimizer	adam
Metrics	accuracy
Epoch	30
Batch size	14

4.2.3 1D CNN Modeli

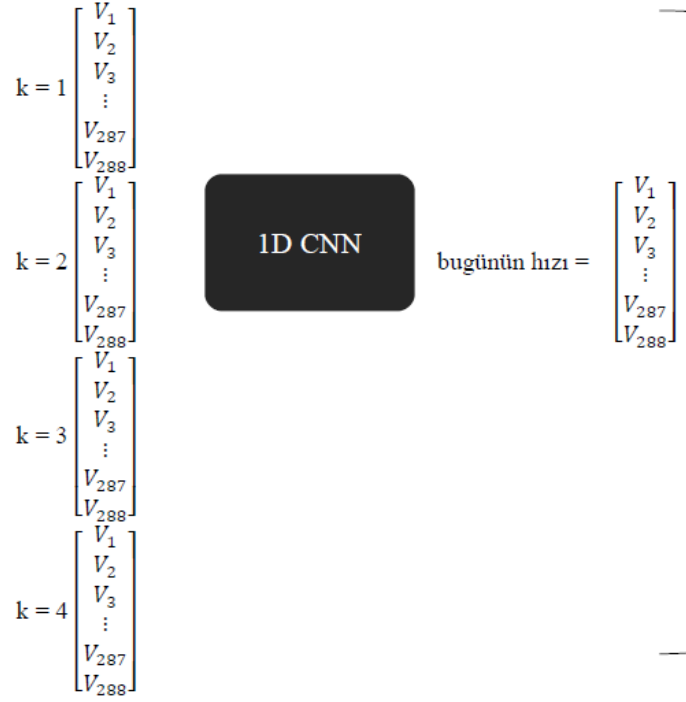
Öncelikle hız verileri Eşitlik 4.4 deki normalizasyon formülü kullanılarak 0-1 aralığına indirgenmiştir. Ardından 4.9 de görülen formata getirilmiştir.

Eğitilmek istenen günün hızı çıkış değeri olarak sisteme verilmiş giriş değerleri olarak da 1,2,3 ve 4 hafta öncesi aynı günün hız verileri verilerek sistem eğitilmiştir.

V_n : günün n . dilimi için hız verisi

k : kaç hafta öncesinin verileri olduğunu belirtmektedir

Modelin girdi kümesi oluşturulurken her bir gün için 4 hafta öncesine bakılarak $[1152 \times 1]$ uzunluğunda dizi elde edilmiştir.

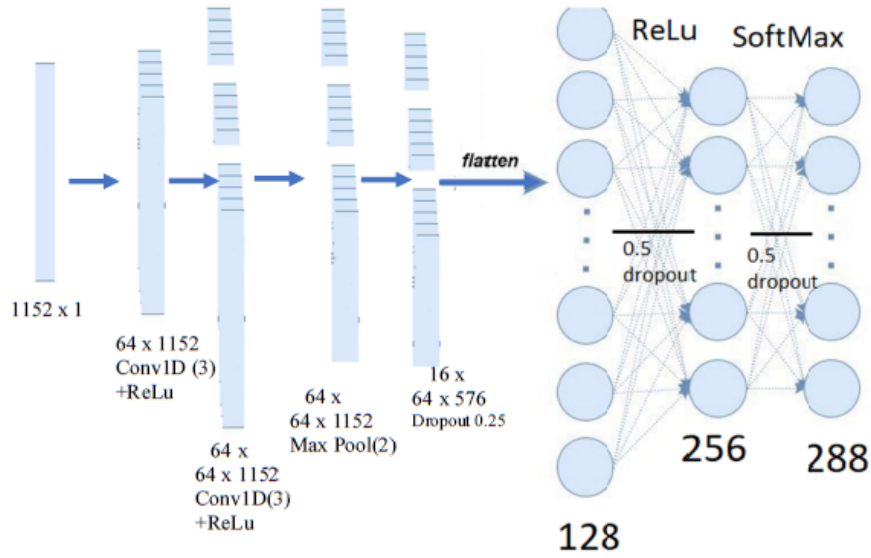


Şekil 4.9 1D CNN modeli için verilerin hazırlanması

Bu işlem Ocak ayı hariç tüm günler için uygulandığında $[334 \times 1152 \times 1]$ boyutlarında matris yapısı elde edilmiştir. Ocak ayı öncesinde veri olmadığı için 4 hafta öncesine bakılamamış doğal olarak da modele dahil edilmemiştir. Çıktı kümesi de $[334 \times 288 \times 1]$ formatında Ocak ayı hariç günün her dilimi için tahmin edilen hız verilerinden oluşmaktadır. Sistem Tasarımı bölümünde bütün bir yılı eğitmek yerine aylık modeller ele alınmıştır.

Eğitim için Tablo 4.2 de 2D CNN için tanımlanan hiperparametlerin aynısı kullanılmıştır.

Modelin ağ yapısı katmanları ve parametreleriyle şekil 5.1 da gösterilmiştir.



Şekil 4.10 1D CNN modeli

4.3 Hata Fonksiyonları

Bu bölümde analiz edilen 3 farklı modelin karşılaştırılmasında kullanılan hata fonksiyonları Eşitlik 4.5, 4.6 ve 4.7 de görülmektedir.

$$\text{Mean Absolute Error } MAE = \frac{1}{m} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (4.5)$$

$$\text{Mean Absolute Percentage Error } MAPE = \frac{100\%}{n} \sum_{t=1}^n \left| \frac{e_t}{y_t} \right| \quad (4.6)$$

$$\text{Mean Squared Error } MSE = \frac{1}{n} \sum_{t=1}^n e_t^2 \quad (4.7)$$

Polinomsal Regresyon modeli uygulanması kolay olmasına rağmen ürettiği çıktılar oldukça iyidir. Trafik akış hızının düzenli olduğu zamanlarda %5 - %10 arası hata oranlarıyla tahminleme yapabilmektedir. Ancak özellikle yaz aylarında ve bazı günlerde %30 hata ortalamasını aşan durumlar da gözlemlenmiştir.

2D CNN modelinin tahmin sonuçları genellikle diğer iki modelin biraz gerisindedir. Ancak daha iyi sonuç verdiği segmentler de vardır. Bu segmentler analiz edildiğinde 2 boyutlu CNN modelinin 1 boyutlu CNN modeline kıyasla eksik verilere olan toleransının daha fazla olduğu tespit edilmiştir.

1D CNN modeli Regresyon modeline yakın sonuçlar vermiştir ancak biraz gerisinde kalmıştır. Bunun sonucunda 1D CNN modelinin üzerine Regresyondan daha iyi sonuçlar alınacak şekilde geliştirme yapılmasına karar verilmiştir.

5

Sistem Tasarımı

Bu bölümde daha önce analizi yapılan modellerden 1D CNN modelinin detaylı tasarımı yapılmış ve Veri modeli gösterilmiştir. Harita Üzerinde Gösterim başlığında da CNN modelinin çıktılarının coğrafi konum kullanarak nasıl görselleştirildiğinden bahsedilmiştir. Harita üzerinde bulunan segment noktalarının akış hızları birbirinden bağımsız tahminlenmiş ve yorumlanmıştır. Sistem Analizinde bulunan rotalama modülünün içerdiği güzergah belirleme işlemleri proje kapsamı dışında tutulmuştur.

5.1 1D CNN Modeli

Bu bölümde daha önce ortaya koyulan Regresyon, 2D CNN ve 1D CNN yaklaşımlarından 1D CNN modeli geliştirilerek gün zaman ve havadurumu bilgileri dahil edilmiştir. Katman tasarımı daha önce sistem analizi kısmında gösterildiği gibi aynen kalmıştır. Modelin girdi kümesi değiştirilerek denemeler yapılmıştır.

5.1.1 Gün ve Zaman Bilgisinin Eklenmesi

Trafik yoğunluğu aynı lokasyon için haftanın günlerine bağlı olarak değişkenlik gösterebilmektedir. Örneğin hafta içi günleri ile haftasonu günleri arasında oldukça ciddi farklar gözlenmektedir. Bu yüzden eğitim modeline girecek günlerin haftanın hangi gününe ait olduğu bilgisi modele eklenmiştir. Bu bilgi eklenirken haftanın her gününü temsil etmek üzere bir sütun eklenmiş ve sadece hızın ait olduğu gün sütununun 1 diğer sütunların 0 olduğu One Hot Encoding yapısı kullanılmıştır.

Hız bilgileri ardışıl verilmesine rağmen temsil ettikleri günün dilimlerinin model tarafından bilinmesi amacıyla her hızın ait olduğu zaman dilimini içeren bir sütun eklenmiştir.

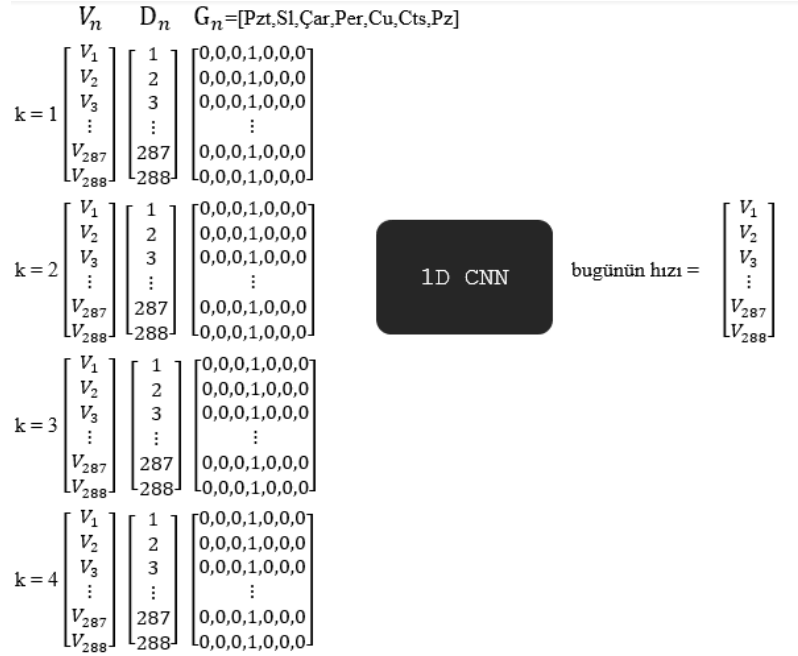
V_n : hız verisi

D_n : zaman dilimi bilgisi

G_n : hızın ait olduğu gün bilgisi

k : kaç hafta öncesinin verileri olduğunu belirtmektedir

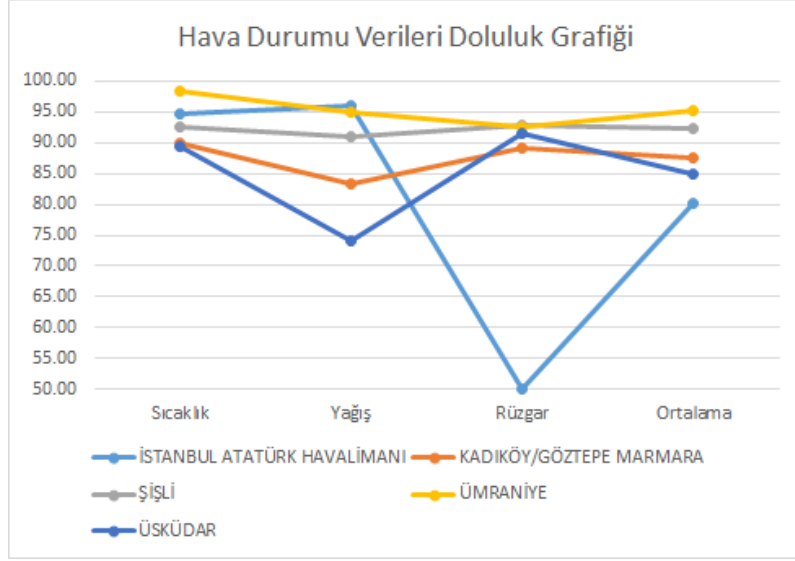
Modelin girdi kümesi oluşturulurken her bir gün için 4 hafta öncesine bakılarak $[1152 \times 9]$ uzunluğunda matris elde edilmiştir.



Şekil 5.1 1D CNN modeli için verilerin hazırlanması

5.1.2 Hava Durumu Bilgilerinin Eklenmesi

Hava durumu bilgileri trafik akışını etkileyen faktörlerden biridir. Bu bilgiler elde edilirken hız verileriyle aynı yıl olan 2017 yılı hava durumu bilgisi incelenmiştir. Şekil 5.2 de hava durumu verilerinin doluluk oranları görülmektedir. Veri yoğunluğu ve konumunun merkezियeti göz önünde bulundurulduğunda İstanbul genelini temsil etmek üzere Şişli konumu seçilmiştir. Bu konuma ait hava durumunun İstanbul geneliyle aynı olduğu varsayımı üzerinden ilerlenmiştir.



Şekil 5.2 Hava Durumu Verilerinin Doluluk Grafiği

Sırasıyla sıcaklık, yağış ve rüzgar bilgileri modele ayrı ayrı dahil edilip sonuçlar değerlendirilmiştir.

5.1.2.1 Sıcaklık Bilgisi

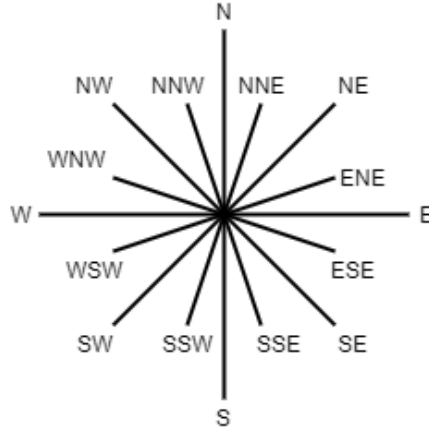
Sıcaklık değerleri °C cinsinden saat başı bir ölçüm hassasiyetindedir. Bu değerler için herhangi bir kategorizasyon yapılmamıştır. Değerlerin normalizasyonu sağlandıktan sonra eğitim veri setinin ilgili gününe ait sıcaklık bilgisi yeni bir sütun olarak eklenmiştir.

5.1.2.2 Yağış Bilgisi

Yağış değerleri, m^2 ye düşen yağışın yüksekliğini mm (milimetre) cinsinden saatlik kaydedilmesiyle elde edilmiştir. Yağış değerleri için de herhangi bir kategorizasyon yapılmamıştır. Değerlerin normali alınıp eğitim veri setinin ilgili gününe ait yağış bilgisi yeni bir sütun olarak eklenmiştir.

5.1.2.3 Rüzgar Bilgisi

Rüzgar bilgisi Şekil 5.3 de görülen 16 yönü ve esme hızını km/s cinsinden içermektedir. Yönün tayin edilmesinde regular expression kullanılarak bu yönler 0-15 arasında kategorilere ayrılmıştır. Daha sonra One Hot Encoding Uygulanıp rüzgar hızlarının normalize değerleriyle birlikte eğitim modeline dahil edilmiştir. Rüzgar verileriyle birlikte modele 16 sütunluk yeni bilgi eklenmiş olmuştur.



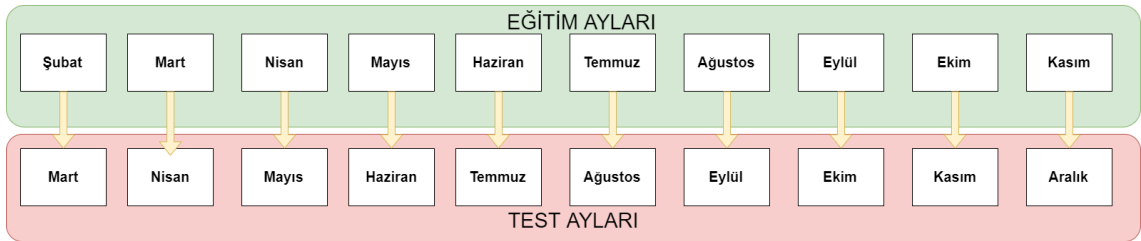
Şekil 5.3 Rüzgarın Yönleri

5.1.3 Eğitime Dahil Edilecek Ay Aralığının Belirlenmesi

Bundan önceki bölümlerde eğitim verilerinin nasıl kullanılacağına üzerinde durulmuştur. Eğitimin diğer önemli parametresi de verilerin ne kadarının kullanılacağıdır. Bu bölümde belirli mantıksal çıkarımlar yapabilmek adına 3 farklı eğitim aralığı seçilmiştir.

5.1.3.1 Son 1 Aylık Veriler Kullanılarak Eğitim

Bu eğitim modeli gerçekleştirilirken eğitim ayı olarak Şubat ayından, tahmin ayı olarak Mart ayından başlanmıştır. Ocak ayından eğitime başlanamamaktadır çünkü her eğitim günü için girdi olarak son 4 haftanın aynı günlerine ihtiyaç olduğundan bu veriler bir önceki yılın Aralık ayına sarmaktadır. Görüldüğü üzere tahmin aralığını etkileyen bu sınırlama veri setinden kaynaklıdır. 2016 yılı verileri de olsaydı Ocak ayından başlanabilirdi. Şekil 5.4 de modellerin başlangıç ve bitiş sınırları ifade edilmiştir. Eğitime için her segment için toplamda 10 adet model kullanılmıştır.

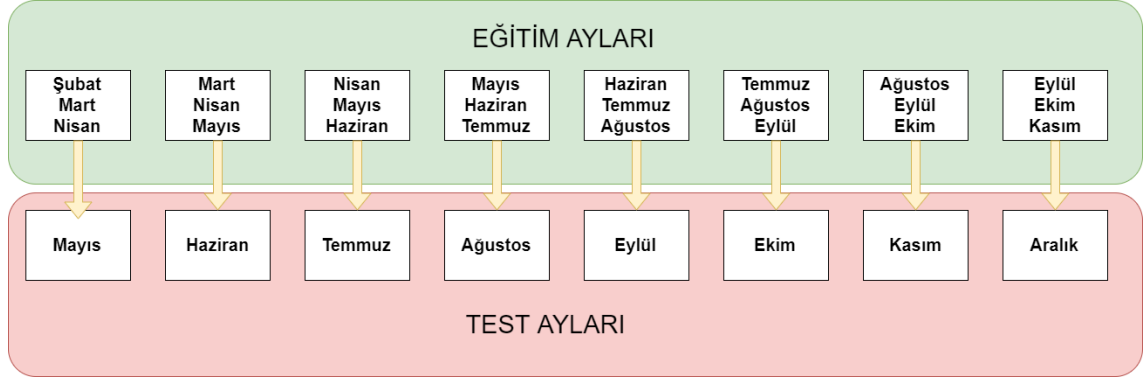


Şekil 5.4 Son 1 Aylık Veriler Kullanılarak Eğitim

5.1.3.2 Son 3 Aylık Veriler Kullanılarak Eğitim

Bu eğitim modeli gerçekleştirilirken eğitim ayı olarak Şubat ayından, tahmin ayı olarak Mayıs ayından başlanmıştır. Şekil 5.5 de bu durum daha iyi görülmektedir. 3

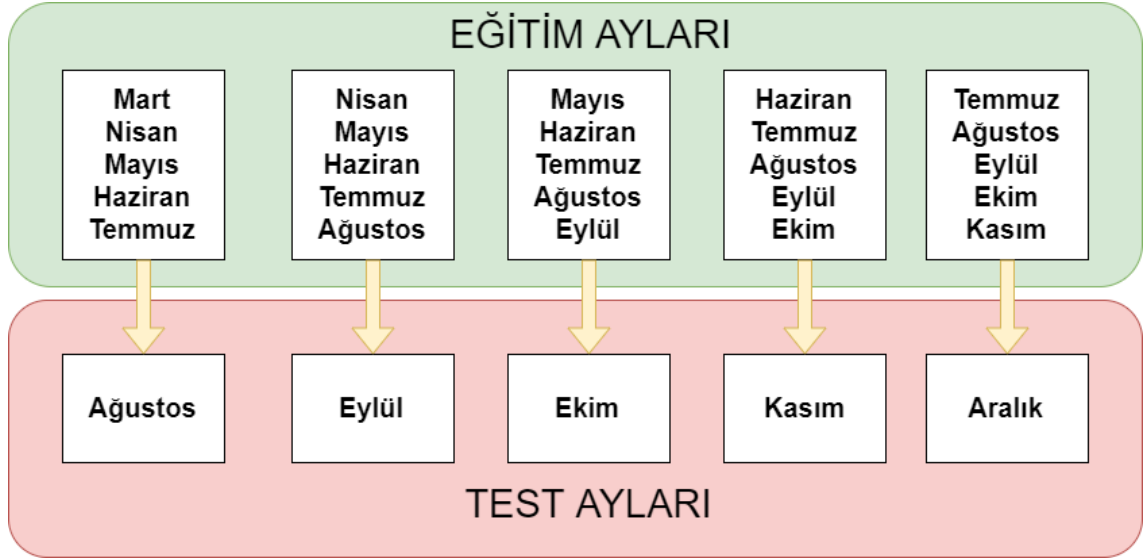
aylık eğitim mevsim geçişlerinden etkilenen bir aralıktır etkileri ilerleyen bölümlerde tartışılmıştır.



Şekil 5.5 Son 3 Aylık Veriler Kullanılarak Eğitim

5.1.3.3 Son 6 Aylık Veriler Kullanılarak Eğitim

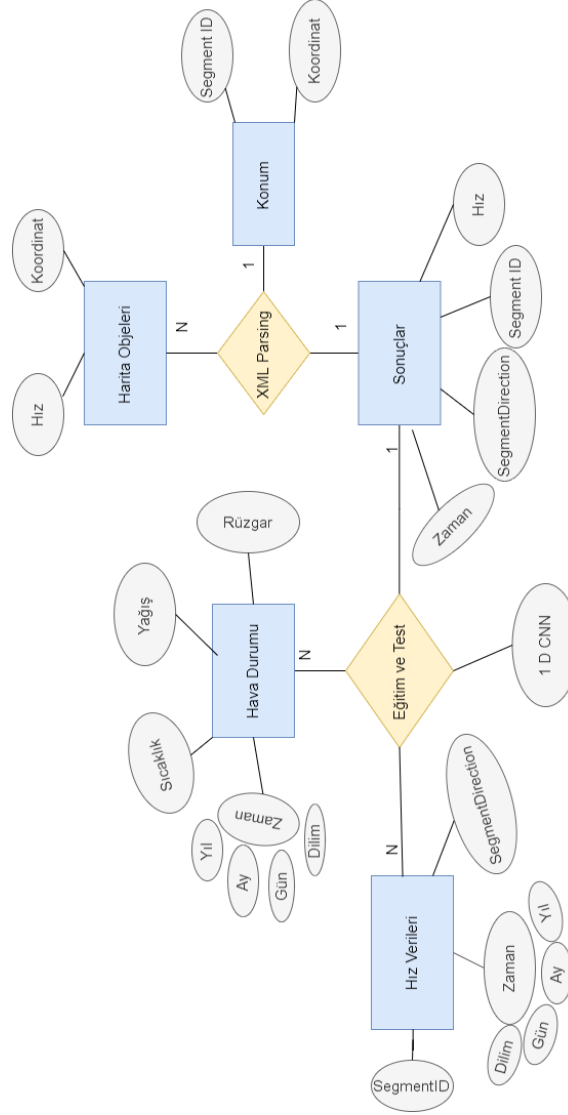
Bu eğitim modelinin tasarlanma amacı yağış verilerinin etkisini daha iyi görebilmektir. 1 ay eğitim modelinde yağışlı gün sayısının az olması durumunda yağış bilgisi öğrenilememektedir. 6 aylık modelin dezavantajı ise 6 ay mevsim geçişlerini ve trafikte meydana gelen bir çok değişimi kapsamaktadır. Bu durumlar Deneysel Sonuçlar kısmında detaylı incelenmiştir. Son aylık veriler kullanılarak oluşturulan eğitim modelinin aralığı Şekil 5.6 de görülmektedir.



Şekil 5.6 Son 6 aylık veriler kullanılarak eğitim

5.2 Veri Tabanı Tasarımı

Şekil 5.7 de görülen diyagramda sisteme giren ve çıktı olarak kaydedilen verilerin arasındaki ilişki görselleştirilmiştir.



Şekil 5.7 Varlık İlişki Diyagramı

5.3 Harita Üzerinde Gösterim

Eğitim bölümünde üretilen çıktılar segmentlere ait hız sonuçları ve hata oranlarıdır. Bu segmentlerin fiziki dünyadaki karşılıklarının harita üzerinde gösterilmesi sonuçların yorumlanması açısından son derece önemlidir. Bu aşamadan sonra modellerin başarısını bölgesel olarak da yorumlamak mümkün hale gelmiştir.

5.3.1 KML Dönüşümü

Modelin ürettiği sonuçlara koordinat bilgisinin eklenmesiyle segmentlerin harita üzerinde işaretlenmesini sağlayan KML formatında dosya üretilmiştir. KML dosyası Google Earth, Haritalar ya da KML kodlamayı uygulayan başka coğrafi yazılımlar için bir dizi özellikler (yer işaretleri, görüntüler, çokgenler, 3D modelleri, metinsel açıklamaları, vs.) belirtir.[7]

Harita üzerinde okulumuzun Elektrik Elektronik Fakültesini gösteren örnek bir kml dosyası örneği şekil 5.8 de verilmiştir.

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
  <Document>
    <Placemark>
      <name>YTÜ EEF</name>
      <description><![CDATA[<h1>YTÜ Elektrik Elektronik Fakültesi</h1>]]></description>
      <Point>
        <coordinates>28.8908835,41.0282786</coordinates>
      </Point>
    </Placemark>
  </Document>
</kml>
```

Şekil 5.8 KML Örneği

Bu işlemin otomatizasyonu python string işlemleri ve xquery kullanılarak yapılmıştır.

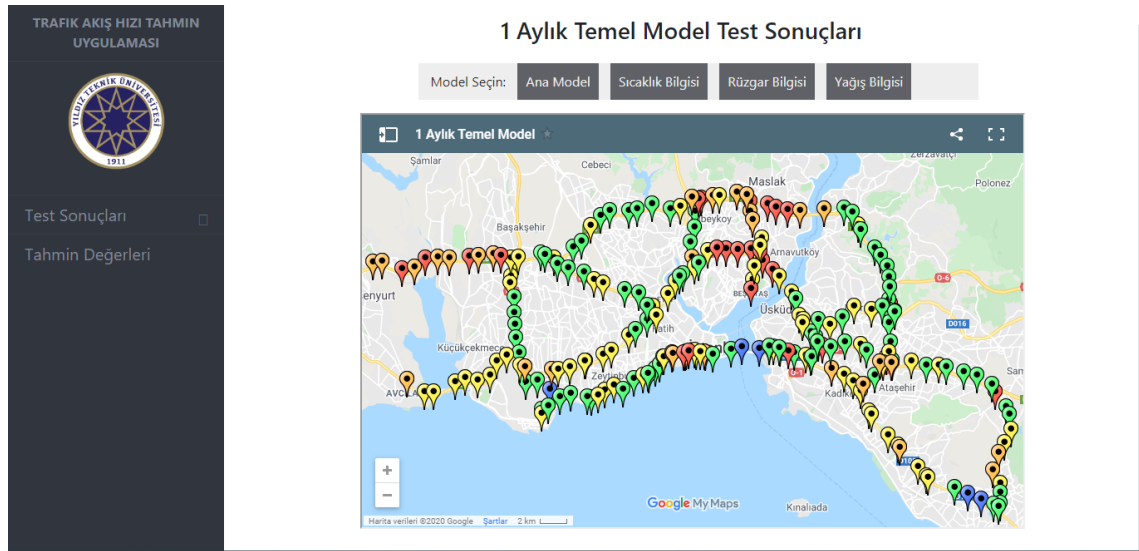
5.3.2 Google API

Üretilen KML dosyasını yorumlayacak harita moturu için Google servisleri kullanılmıştır. KML dosyası Google sunucularına yüklendikten sonra üretilen anahtarla websitesi veya mobil uygulamaya kolayca entegre olmaktadır.

6 Uygulama

Bu bölümde uygulamaya ait ekran görüntülerine yer verilmiştir.

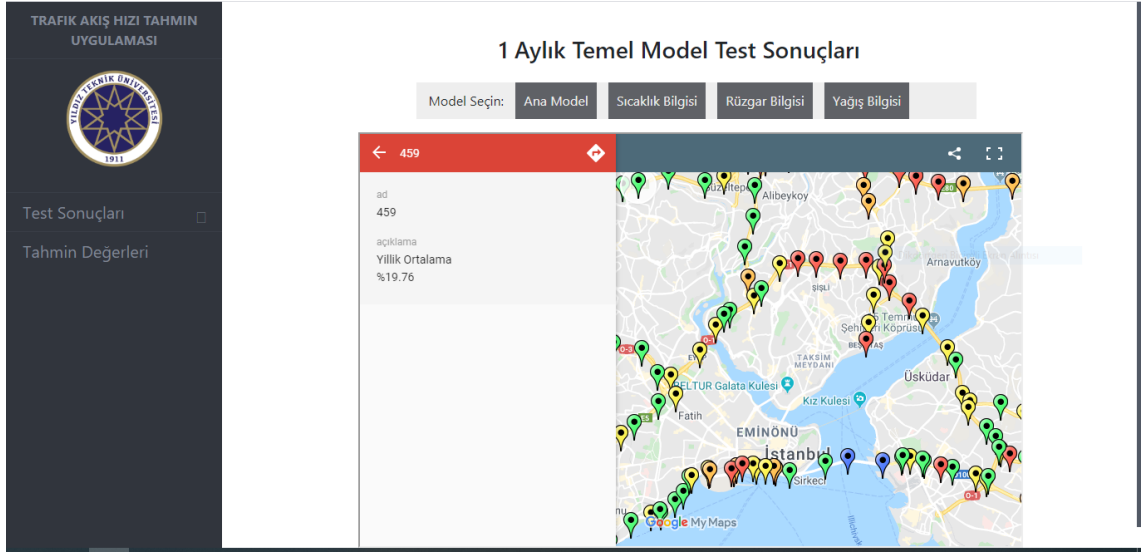
Şekil 6.1 de 1 aylık temel modelin yıllık ortalama hata oranları (MAPE) harita üzerinde gösterilmiştir. Renkler hata oranlarını temsil etmektedir. Renk ölçeği Şekil 6.4 te ifade edildiği gibi yeşilden kırmızıya doğru hata miktarı artacak şekilde belirlenmiştir.



Şekil 6.1 Harita Üzerinde Segmentlerin Gösterilmesi

Şekil 6.2 de harita üzerinde görüldüğü gibi segmentlerin üzerine tıklandığında hata oranlarının yıllık ortalaması ekrana yansıtılmaktadır.

Verilen segmente ve tarihe ait bir saat aralığında hız bilgilerinin gerçek ve tahmin değerlerinin karşılaştırıldığı kısım Şekil 6.3 te görülmektedir.



Şekil 6.2 Segmentler ve Hata Oranları

Trafik Akış Hızı Tahmin Modülü

Görüntülenen Değerler
Segment ID: 30
Yön: A-B

Tarih	Saat	Yön	Segment ID	K Değeri	Tahmin Et
saat	gerçek	tahmin			
11:0-11:5	89.2	82.46266			
11:5-11:10	88.0	83.10021			
11:10-11:15	0.0	83.25026			
11:15-11:20	0.0	85.93411			
11:20-11:25	0.0	85.21956			
11:25-11:30	0.0	85.26345			
11:30-11:35	91.0	84.715			
11:35-11:40	91.0	86.53482			

Şekil 6.3 Tahminler ve Gerçek Değerler

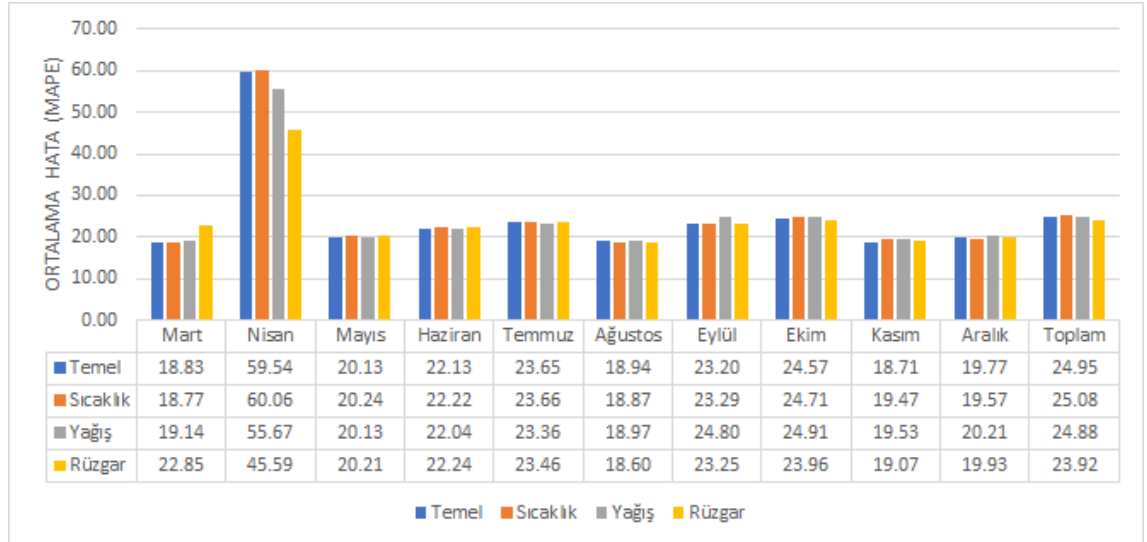
Renk Sınıfı	Hata Oranı (MAPE %)
Mavi	0 -- 10
Yeşil	10 -- 20
Sarı	20 -- 30
Turuncu	30 -- 40
Kırmızı	> 40

Şekil 6.4 Renk Ölçeği

7 Deneysel Sonuçlar

Bu bölümde Sistem Tasarımı bölümünde ortaya koyulan modellerin sonuçları incelenmiştir.

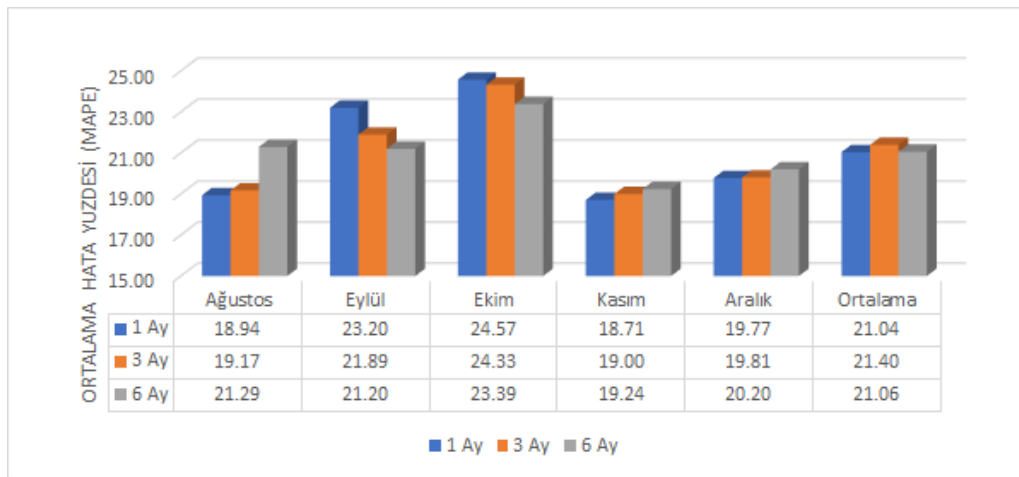
Şekil 7.1 de her ayın kendinden önceki 1 ay ile eğitilmesi sonucuna havadurumunun etkileri görülmektedir. Burada ilk dikkat çeken sonuç Nisan ayındaki anormal hata oranıdır. Ortalamadan neredeyse 3 kat daha fazla hata olmasının sebebi Şubat ayı verileridir. Veri setinde şubat ayına ait hız verilerinde ciddi bir eksiklik bulunmaktadır. Birçok segment için bu ayda neredeyse hiç veri yoktur. Son 1 ay verileriyle yapılan eğitim modelinde Mart ayının verileri eğitilerek bu modelle Nisan ayı test edilmiştir. Mart ayı verileriyle eğitim gerçekleştirirken de 4 haftaya kadar öncesine bakıldığı için Şubat ayının verilerinde kalıtım alınmaktadır. Bu durum modelin düzgün eğitilememesine sebebiyet vermiştir. Bu sorun diğer grafiklere de yansımaktadır.



Şekil 7.1 Son 1 Ay ile Eğitim Hata Yüzdeleri

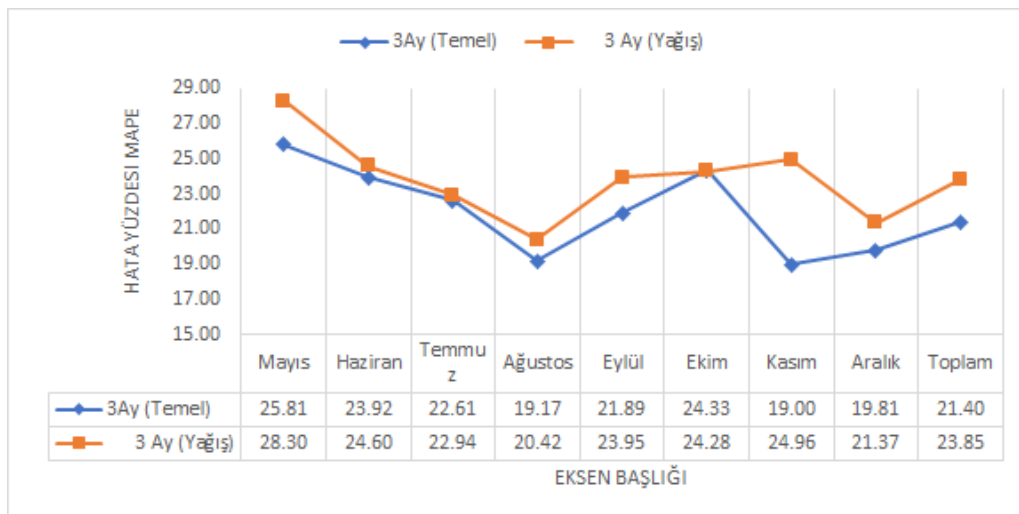
Şekil 7.1'i havadurumu verilerine göre yorumlamak gerekirse yağış verisi trafiğe yüksek oranda etki ettiği halde sonuçlara beklenen etkiyi göstermemiştir. Bunun sebebi 1 ay içerisinde denk gelecek yağmurlu gün sayısının yeterince yüksek olmaması

durumunda model yağış bilgisini öğrenememektedir. Sıcaklık bilgisinin de sonuçlara olumlu katkısı gözlenmemiştir bunun sebebi de 1 ay içerisinde günlük sıcaklık farkının benzer olmasından kaynaklı olabilir. Rüzgar bilgisi ise ortalamada ve birçok ayda az da olsa iyileştirme sağlamıştır. Bölgesel havadurumu verilerinin kullanılması durumunda rüzgar bilgisi çok daha yararlı olabilir. Şekil 7.2de Son 1,3 ve 6 aylık veriler kullanılarak yapılan modeller karşılaştırılmıştır. 5 aylık ortalamada 3 model de birbirine oldukça yakındır. Ağustos ayında diğer 2 modelden gerisinde kalan 6 aylık model Eylül ve Ekim aylarında daha iyi sonuçlar çıkarmıştır. Buradan çıkarılabilecek bir başka sonuç da Şekil 7.1de 1 aylık (temel) modelin hatası %24.95 iken Nisan ayı hesaba katılmadığında %21.04 e kadar düştüğü görülmektedir. Diğer bir deyişle modelin ortalama %79 başarıya ulaştığını söyleyebiliriz.



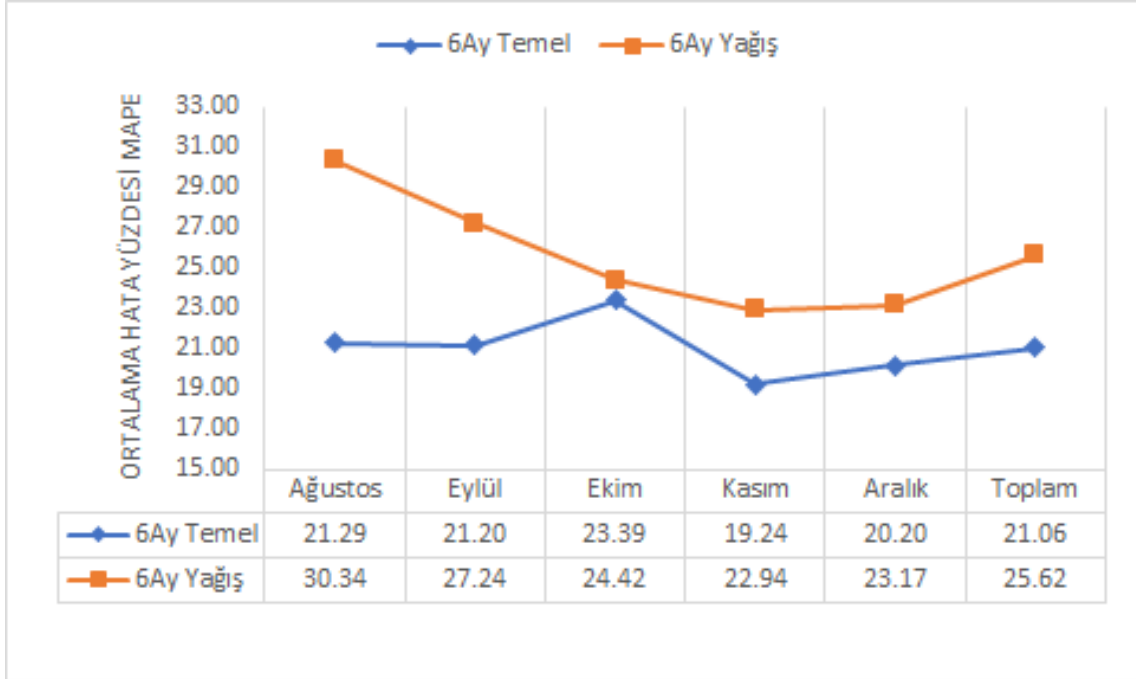
Şekil 7.2 1 - 3 - 6 Ay Eğitim Modellerinin Hata Yüzdeleri

Şekil 7.3de 3 aylık modele yağış bilgisi eklenmiştir. Veri aralığı arttıkça yağış bilgisinin daha işe yarar olması düşünülse de grafikte görüldüğü gibi pratikte sonuçlar daha kötüye gitmiştir. Ekim ayı hariç tüm aylarda yağışlı model daha başarısız olmuştur.



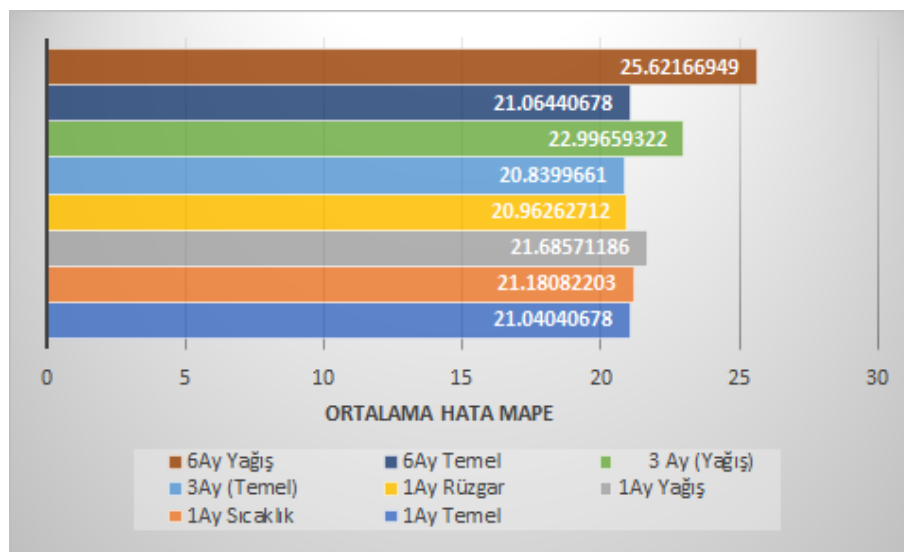
Şekil 7.3 3 Aylık Temel ve Yağışlı Modelin Hata Yüzdesi

Şekil 7.4de 6 aylık modele yağış bilgisi eklenmiştir ancak 3 aylık modelde olduğu gibi hata oranlarında sürekli artış gözlemlenmiştir. Buradan çıkarılacak sonuç yağış bilgisini modele parametre olarak verilmesi uygun değildir. Yağışlı günler tespit edilip bunlar için ayrı bir model oluşturmak gerekmektedir.



Şekil 7.4 6 Aylık Temel ve Yağışlı Modelin Hata Yüzdesi

Şekil 7.5 de tüm modellerin Ağustos ayından Aralık ayına kadar olan 5 aylık ortalaması alınarak karşılaştırılmıştır. 1 yıllık ortalamanın kullanılmamasının sebebi yukarıda bahsedilen Nisan sonuçlarının yanıltıcı olmasındandır.



Şekil 7.5 Tüm Modellerin Ortalama Hata Yüzdesi

8 Sonuç

Bu projede yardımcı trafik uygulamalarının trafik akış hızı modülü üzerinde durulmuştur. Kullanılacak modelin belirlenmesi sürecinde Polinomsal Regresyon, 1D CNN ve 2D CNN algoritmaları denenmiş ve 1D CNN modeli geliştirilmeye karar verilmiştir. Başlangıçta Regresyon modelinin gerisinde olan 1D cNN modeli, gün ve zaman bilgilerinin eklenmesiyle birçok segmentte Regresyondan daha iyi sonuç üretir hale gelmiştir. Projenin devamında 1D CNN üzerinden ilerlenmiştir ve başarı kriteri olarak kullanılan hata fonksiyonu MAPE (Mean Absolute Percentage Error) dir.

1D CNN modeli ile eğitim yapılırken son kaç ayın verisi kullanılacağı konusunda son 1,3 ve 6 aylık verileri kullanan modeller denenmiştir. 6 aylık model kendinden önceki 6 ayın verisine ihtiyaç duymaktadır. Kullanılan veri seti 2017 yılının verilerini içerdiği ve bir önceki yılın verileri mevcut olmadığı için eğitime ancak Temmuz ayından başlanabilmektedir. Kıyaslama yapılırken de tutarlılık sağlanması için bu 3 modelin Temmuzdan sonra gelen aylarının ortalamasına bakılmıştır. Bunun sonucunda en iyi sonuç %21 hata oranıyla 1 aylık modelde görülürken en kötü sonuç %25.6 hata ortalamasıyla 6 aylık modelde görülmüştür.

Kullanılan veri setinde eksik verilerin bulunması üretilen sonuçları fazlasıyla etkilemiştir. Örneğin son 1 aylık veriyi kullanarak yapılan tahmin modelinde Nisan ayı sonuçlarında hata oranı %59 a kadar çıkmıştır. Bunun sebebi Nisan ayı üzerine tahmin yapılırken Mart ayının verileriyle eğitim yapılmakta Mart ayı verileriyle eğitim yapılırken de 4 hafta öncesine kadar veriler yani Şubat ayının verileri kullanılmaktadır. Şubat ayı eksik verilerin anormal düzeyde bulunduğu bir aydır. Birçok segment için neredeyse hiç veri yoktur ve modeller bu ayın verilerini kullandığı zaman eğitilememekte doğal olarak tutarlı tahmin yapamamaktadır.

Eksik verilerin tahmin sonuçlarını negatif etkilediği görüldüğü gibi veri doluluğu fazla olan segmentlerin sonuçlarının da % 5 yakın hata oranlarıyla diğer bir deyişle %95 e varan başarılarla tahmin edildiği görülmüştür. Ancak bazı segmentlere ait veriler dolu olduğu halde sonuçların %20 leri aştığı dikkat çekmiştir. Bu

segmentler incelendiğinde trafik akış hızının düşük olduğu görülmüştür. Kullanılan hata fonksiyonunun MAPE olmasından dolayı nicelik olarak tahmin ile gerçek değer arasında fazla fark olmamasına rağmen yüksek hata oranlarıyla karşılaşılmaktadır. Örn trafik akış hızının 50 km/s olduğu bir lokasyon 60 km/s hızında tahmin edildiği vakit %20 hata oranı hesaplanmaktadır.

Havanın durumu trafiğin seyrini beklenmedik şekilde değiştirebilmektedir. Bu projede hava durumu verilerinden yararlanılarak tahminlemede başarıyı arttırmak hedeflenmiştir. İşlemi basitleştirmek adına İstanbul genelini temsilen Şişli konumuna ait hava durumu verileri kullanılmıştır. Sıcaklık verileri modele eklendiğinde istenilen iyileşme görülememiştir.

Yağış konusu üzerinde özellikle durulmuştur çünkü trafik üzerinde yadsınamaz bir etkisi vardır. Yağmurlu günlerde tahmin yapmak mevcut uygulamaların da yaşadığı en büyük sıkıntılardan biridir. Yağış bilgisi 1 aylık modele eklendiğinde beklenenin aksine neredeyse başarıyı neredeyse hiç etkilememiştir. Bunun sebebi olarak 1 aylık verinin içerisinde yağmurlu gün sayısının az olabileceği veya hiç olmaması durumunda model yağmur bilgisini öğrenemediği için sonuçları iyileştirememektedir. 3 ve 6 aylık modellere yağmur bilgisi eklenerek işlem tekrarlandığında iki durumda da modelin hata oranı yaklaşık %4 artmıştır. Bu bilgilerin ışığında yağmur bilgisinin modele parametre olarak verilmesinin iyi olmadığı bunun yerine yağmurlu günler için ayrı model oluşturulabileceği kanısına varılmıştır.

Rüzgar verileri modele eklendiğinde bir çok ay için iyileşme görülmüştür. Eksik verilerden dolayı iyi tahmin yapılamayan Nisan ayı hata oranlarını %15 düşürmüş 1 yıllık ortalamada temel modelden %1 daha az hatalı sonuçlar üretmiştir. Hava durumu verilerinin genelleştirilmesinden dolayı bu fark oldukça önemlidir. Çünkü rüzgarın yönü ve hızı İstanbul'un bölgesine göre değişiklik gösterebilmektedir. Bölgesel hava durumu verileri kullanıldığı takdirde sonuçların daha da iyileşmesi olasıdır. Buradan çıkarılabilecek sonuç rüzgarla trafik akış hızı arasında bir ilişki olduğudur.

Bu projede ele alınan modeller sonucunda tahminleme kısmı yaklaşık %20 - %25 hata ile gerçekleştirilmiştir. Bir başka deyişle modellerin başarısı %75 - %80 arasındadır. Proje sonunda bu tahminleri daha iyiye götürebilmek adına yukarıda da bahsedildiği gibi birçok çıkarım yapılmıştır. Bunlardan bir diğeri de her segment için ayrı eğitim yapmak yerine birbirleriyle yol bağlantısı olan segmentlerin verileriyle tek bir eğitim yapmaktır. Modele giren veri boyutunu arttırdığı için derin öğrenme modelinin performansını da arttırması yüksek olasıdır.

Bu proje sonucunda geliştirilen uygulamada seçilen segment için girilen zamana ait trafik akış hızı 1D cNN yöntemi kullanılarak tahmin edilebilmektedir. Bugünden

1 hafta sonrasına kadar herhangi bir anda tahmin sonuçlarının görüntülenmesi mümkündür. İnsanlar bu sayede önemli seyahatlerini planlamadan önce trafik durumunu kontrol ederek istedikleri yere vaktinde ulaşabilirler ve trafiğin daha az yoğun olduğu güzergahlara kullanarak trafikte geçirdikleri süreyi azaltabilirler.

Referanslar

- [1] D. Barth. (Aug. 9). The bright side of sitting in traffic: Crowdsourcing road congestion data, [Online]. Available: <https://googleblog.blogspot.com/2009/08/bright-side-of-sitting-in-traffic.html?m=1> (visited on 12/19/2019).
- [2] M. G. Müdürlüğü. (Dec. 19). 2019 yılı aralık ayı alansal yağış raporu, [Online]. Available: <https://www.mgm.gov.tr/veridegerlendirme/yagis-raporu.aspx>.
- [3] Y. Company. (Jan. 18). 2019 yılı aralık ayı alansal yağış raporu, [Online]. Available: https://yandex.com/company/technologies/traffic_jams_technology/.
- [4] U. Y. Merkezi. (). İbb ceptrafik, [Online]. Available: <https://uym.ibb.gov.tr/hizmetler/ibb-cep-trafik> (visited on 01/07/2020).
- [5] A. S. Foundation. (2004). Apache license, [Online]. Available: <https://www.apache.org/licenses/LICENSE-2.0>.
- [6] keras-team. (Mar. 18). Copyright, [Online]. Available: <https://github.com/keras-team/keras/blob/master/LICENSE>.
- [7] Vikipedi. (). Kml yapısı, [Online]. Available: <https://tr.wikipedia.org/wiki/KML> (visited on 01/08/2020).

BİRİNCİ ÜYE

İsim-Soyisim: Caner Kaya

Doğum Tarihi ve Yeri: 01.10.1998, İzmir

E-mail: canerkaya718@gmail.com

Telefon: 0537 745 35 38

Staj Tecrübeleri: Yıldız Teknik Üniversitesi - Akıllı Sistemler Laboratuvarı

Proje Sistem Bilgileri

Sistem ve Yazılım: Windows İşletim Sistemi, Python

Gerekli RAM: 8GB

Gerekli Disk: 128GB