

Uçak Motoru Sağlığı için Uzun-Kısa Süreli Bellek Yöntemi ile Öngörücü Bakım

Araştırma Makalesi/Research Article

 Merve Ayyüce KIZRAK¹,  Bülent BOLAT²

¹Elektrik-Elektronik Mühendisliği, Bahçeşehir Üniversitesi, İstanbul, Türkiye

²Elektronik ve Haberleşme Mühendisliği, Yıldız Teknik Üniversitesi, İstanbul, Türkiye

merveayyuce.kizrak@eng.bau.edu.tr, bbolat@yildiz.edu.tr

(Geliş/Received:12.12.2018; Kabul/Accepted:28.03.2019)

DOI: 10.17671/gazibtd.495730

Özet—Otomotiv, uçak ve fabrika otomasyonu gibi özellikle maliyetli motor bakımı gerektiren uygulamalarda öngörücü bakım önemli bir yer almaktadır. Hem iş güvenliği hem de araçlardan sağlanacak verim bakımından motorların bakım periyotlarını önceden kestirmek ve buna göre iş yönetim stratejisi geliştirmek önemlidir. Öngörücü bakım için motorlardan alınan sensör verileri motorun yıpranma süresini ve seviyesini belirlemede kullanılmaktadır. Çalışmada Uzun-Kısa Süreli Bellek (LSTM) yapısı kullanılarak, uçak motorlarının kalan yaşam ömürlerinin tahmin edilmesi amaçlanmıştır. NASA tarafından sunulmuş olan bir veri kümesi üzerinde LSTM yapısı test edilmiştir ve elde edilen sonuçlar farklı yöntemlerle kıyaslanmıştır. Yapılan uygulamaların sonucunda en yüksek sınıflandırma başarımı %98,876; en düşük ortalama mutlak hata ise 1,343 olarak LSTM ile elde edilmiştir.

Anahtar Kelimeler— öngörücü bakım, derin öğrenme, uzun-kısa süreli bellek, yapay sinir ağları, makine öğrenmesi, uçak motor sağlığı.

Predictive Maintenance of Aircraft Motor Health with Long-Short Term Memory Method

Abstract— Predictive maintenance is an important part of applications that require costly engine maintenance, such as automotive, aircraft and factory automation. It is important to anticipate the maintenance periods of the engines and develop a business management strategy accordingly in terms of both work safety and efficiency. For predictive maintenance, the sensor data from the motors were used to determine the wear time and level of the engine. In this study, a solution based on deep learning is proposed as an alternative to traditional regression and classification methods. The NASA Turbofan Engine Corruption Simulation data set was studied by using Long-Short Term Memory (LSTM), one of the deep learning models and known to make successful predictions on time-dependent data such as time series. During the simulations, the highest classification performance and the lowest mean absolute error were obtained by LSTM as 98,876% 1.343 respectively.

Keywords— predictive maintenance, deep learning, long-short term memory, artificial neural networks, machine learning, aircraft health system.

1. GİRİŞ (INTRODUCTION)

Bir uçak motorunun bakımı karmaşık, zaman alıcı ve her şeyden önce pahalı bir işlemdir. Doğrudan motor bakım maliyetleri bir uçağın toplam bakım maliyetinin yaklaşık %30'unu oluşturmaktadır [1]. Uçak motoru bakımı, tarihsel olarak çoğunlukla sabit zaman aralıklarıyla gerçekleştirilmekteydi. Ancak havacılık sektörünün gelişmesiyle birlikte bu yaklaşımın doğru olmadığı görülmüştür. Günümüzde motor bakımı sadece gerek duyulduğunda gerçekleştirilmekte ve sabit zamanlı bakımların sayısı azaltılmaya çalışılmaktadır.

Modern uçak motorlarında motorun durumunu değerlendirmek amacıyla çok sayıda sensör bulundurulmaktadır. Bu sensörlerden gelen veriler değerlendirilerek motorun yararlı ömrü (Remaining Useful Life-RUL) tahmin edilmeye çalışılır. Yararlı ömür tahminindeki amaç, motorda yaşanabilecek olası hasar ve arızaların herhangi bir kaza gerçekleşmeden önce öngörülmesi ve önleyici bakım faaliyetlerinin gerçekleştirilmesi olarak özetlenebilir [1]. Tahmine dayalı bu önleyici bakım faaliyetlerine öngörücü bakım adı da verilir.

Uçak motorlarının bakım faaliyetleri iki temel gruba ayrılabilir. Metal yorulması, mikroskobik bozulmalar gibi etkiler nedeniyle kullanım süresi kısıtlanmış parçaların değiştirildiği bakım faaliyetleri sabit zamanlı (SZ) faaliyetler adı verilir. Bir SZ bakım gerçekleştirilirken ilgili parça durumuna bakılmaksızın değiştirilir. Koşula bağlı (KB) faaliyetler ise önleyici birincil bakım süreçleridir. Bir cihazın veya parçanın hizmete devam edip edemeyeceğini belirlemek için belirli bir fiziksel standarda karşı düzenli olarak denetlenmesini veya kontrol edilmesini gerektirir. Bu denetleme ve gerekliyse onarım sürecinin ne zaman gerçekleştirilmesi gerektiğinin doğru belirlenmesi bakım masraflarının azaltılmasında önemli bir rol oynamaktadır [1, 2].

Literatür incelendiğinde birçok farklı yöntem ile bakım öngörülmesi yapılmaya çalışıldığı görülmektedir. Xu vd 100 farklı motordan 21 adet sensör yardımıyla alınan veriler ile motorun sağlık durumunu takip etmeye çalışmışlardır [3]. Elde edilen verilerin incelenmesinden sonra 7 sensör haricindekiler motorun durumuna dair anlamlı bilgi içermediği gerekçesi ile elimine edilmiştir. Kalan veriler Dempster-Shafer Regresyonu (DSR), Destek Vektör Makinesi (SVM) ve Tekrarlı Sinir Ağından (RNN) oluşan bir bileşik sistem ile kestirilmiştir. Bileşik kestiricinin ortalama karesel hatası (MSE) 3.49 olarak ölçülmüştür. Malhotra vd [4] NASA Turbofan Engine Corruption Simulation (C-MAPSS) veri kümesinden rastgele seçtikleri 80 adet motorun verilerini Uzun-Kısa Süreli Bellek Kodlayıcı/Kod Çözücü (LSTM-ED) yardımı ile eğitimsiz olarak işlemişlerdir. Prognostics and Health Management 2008 (PHM08) Konferansı sonuç bildirisinde konferans bünyesinde yapılan RUL yarışmasının en başarılı üç yöntemi Benzerlik Temelli Yaklaşım (SBA), RNN, Kalman Süzgeç ve Çok Katmanlı Almaca (MLP) dayalı karma bir yöntem olarak listelenmiştir [5]. Zhang

[6] ve arkadaşları, C-MAPSS veri setinden seçtikleri bir alt kümenin üzerinde doğrusal olmayan uyarlanabilir kestiriciler yardımı ile motorun sağlıklı ya da arızalı olup olmama durumunu kestirmişlerdir. Babu vd [7] C-MAPSS ve PHM08 veri setlerini Destek Vektör Regresyon (SVR), MLP, İlgili Vektör Regresyon (RVR) ve Konvolüsyonel Sinir Ağı (CNN) ile işlemişlerdir. Bu çalışmada CNN'in Xu vd. 100 farklı motordan 21 adet sensör yardımıyla alınan verileri kullanıp bir motorun sağlık durumunu takip etmeye çalışmıştır. Kullandıkları sensörlerden 14 tanesinin motorun sağlık durumuna dair kullanışlı bir bilgi sunmadığı belirlenmiştir. Kalan 7 sensöre ait veriler ile değerlendirme yapılmıştır. Derin İnanç Ağları (DBN) kullanılan bir başka çalışmada [8] C-MAPSS veri setinden seçilen iki alt küme değerlendirilmiştir. Bu çalışmadan en yüksek sınıflandırma başarımı topluluk oylamasına dayalı bileşik sınıflandırıcı ile %90'ın biraz üzerinde bulunmuştur.

Mevcut literatür incelendiğinde konu ile ilgili yakın tarihli çalışmaların çok büyük bir kısmının veri-güdümlü olduğu, model tabanlı çalışmaların ise güncelliklerini yitirdiği görülmüştür. Veri-güdümlü çalışmalar ise iki ana grupta toplanabilir. İlk gruptaki çalışmalarda yararlı ömür süresi tespit edilmeye çalışılmaktadır. Bu tür çalışmalarda motordan alınan geçmiş veriler bir zaman serisi gibi ele alınmakta ve problem bir regresyon problemine dönüştürülmektedir [3, 4, 7]. İkinci gruptaki çalışmalarda ise motorun durumu sağlam, az hasarlı, çok hasarlı gibi kategorilere ayrılarak bir sınıflandırma problemine indirgenmektedir [6, 8, 9]. Bu çalışmada motorun durumu RUL değerine bağlı olarak sağlam ve bakım görmeli şeklinde iki kategoriye ayrılmıştır.

Bu çalışmada bir derin öğrenme modeli olan uzun-kısa süreli bellek (Long-Short Term Memory-LSTM) ile uçak sağlık sistemlerinde öngörücü bakım değerlendirmesi yapılması amaçlanmıştır. Derin öğrenmenin başarısı temel regresyon ve yapay sinir ağı modelleri ile karşılaştırılarak değerlendirilmiştir. Veri kümesi olarak NASA tarafından ücretsiz olarak dağıtılan C-MAPSS veri kümesi kullanılmıştır. Çalışmanın İkinci bölümde veri kümesi ve çalışma ortamı hakkında bilgi verilmiştir. Üçüncü bölümde uzun kısa vadeli bellek üzerinde durulmuştur. Yapılan uygulama ve benzetimlere ait detaylar dördüncü bölümde sunulmuştur. Elde edilen sonuçlar beşinci bölümde özetlenirken altıncı bölümde çalışmanın kısa bir değerlendirmesi yapılmıştır.

2. VERİ KÜMESİ ve ÇALIŞMA ORTAMI (DATASET AND WORKSPACE)

Literatürde uçak sağlık sistemlerinde öngörücü bakım konusu için yaygın olarak kullanılan veri kümesi NASA'nın sunduğu NASA Turbofan Engine Corruption Simulation (C-MAPSS) veri kümesidir [10]. Veri kümesi, C-MAPSS adlı ticari bir benzetim yazılımı kullanılarak NASA mühendisleri tarafından oluşturulmuştur. Yazılım 90.000 libre güç üretebilen bir turbofan motoru 0-40.000 ft irtifaları arasında 0-0.9 Mach hızları arasında ve -60 ila 103 °F ortam sıcaklıklarında benzetim yapılmaktadır. Yazılım

ayrıca motorun üretici tarafından belirlenen çalışma aralığının dışına çıkarılmasını engelleyen çeşitli düzenleyici ve sınırlayıcıları da bulunmaktadır.

Veri kümesi oluşturulurken motor, kontrol sistemi ile birlikte çalıştırılmıştır. Motorun sağlık indeksi sıfıra düştüğünde benzetim durdurulmuş ve elde edilen sensör verileri birer zaman serisi olarak kaydedilmiştir. Motor sağlık indeksi, motoru oluşturan parçaların her biri için ideal durumda 1, belirlenen çalışma koşullarının dışına çıktığı anda ise 0 olacak şekilde belirlenmiştir. Eğitim verileri motor sağlık indeksi 0 olana kadar devam ettirilirken test ve doğrulama verileri RUL ölçümü yapabilmek amacı ile motor bozulmadan önce sonlandırılmıştır. Herhangi bir anda içinde bulunan çevrim ile motor sağlık indeksinin sıfıra düştüğü çevrim arasındaki fark motorun RUL değerini vermektedir. Veri kümesinde farklı çalışma koşulları ve farklı senaryolar için hazırlanmış dört farklı alt küme mevcuttur. Bu çalışmada kullanılan alt kümede 100 adet motordan alınan 21 sensör verisi mevcuttur. Eğitim verisi için toplam 20631 döngü ve test için 13096 döngü kayıt edilmiştir. Veriler motor bakıma ihtiyaç duymaktadır veya sağlamdır şeklinde iki kategoriye etiketlenmiştir. RUL değeri 150'nin altına düşen motorlar bakıma ihtiyaç duymaktadır olarak nitelendirilmiştir.

Derin öğrenme yöntemlerinin yüksek işlemsel kapasite gereksinimi, bu tür çalışmalar için gereken donanımların yüksek maliyetli olmasına neden olmaktadır. Ancak, bazı özel ve kamu kuruluşları araştırmacılar için ücretsiz çalışma ortamları sunmaktadır. Bu çalışmada, tanımlanan problemin çözümünün yanı sıra ücretsiz servislerin görünürlüğünü de arttırmak için çalışma ortamı olarak ücretsiz bulut servisleri tercih edilmiştir. Çalışmada regresyon ve MLP gibi geleneksel yöntemleri uygulamaya olanak tanıyan Microsoft Machine Learning Studio ve derin öğrenme modellerinin verimli çalışması için çok önemli olan GPU (grafik işlem birimi) kullanımı özelliği olan Google Colab servislerinden faydalanılmaktadır. Google Colab, Tesla K80 GPU desteği sunmaktadır. Programlama dili olarak Python 3, kütüphane olarak ise TensorFlow ve Keras Kütüphanesi kullanılmaktadır.

3. UZUN KISA VADELİ BELLEK (LONG SHORT TERM MEMORY)

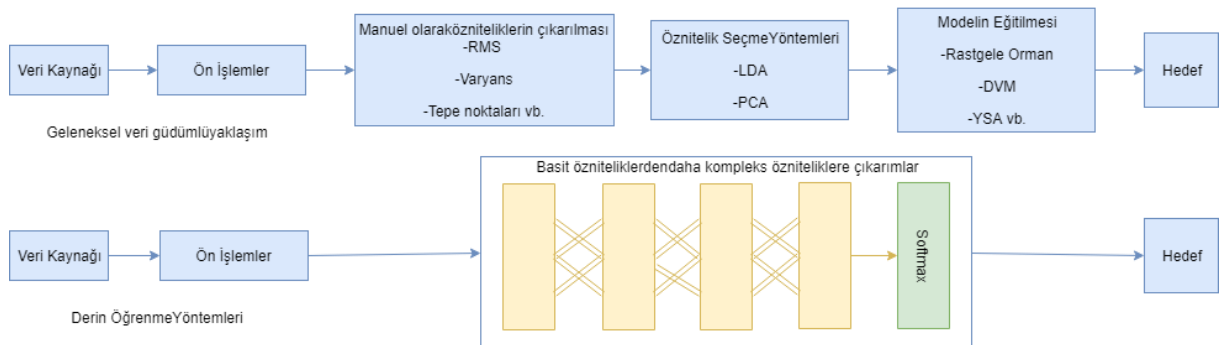
Derin öğrenme yöntemleri, son yıllarda popülerlik kazanmaya başlamışlardır. İlk yıllarında daha çok ses ve görüntü işleme alanlarında uygulanan bu yöntemler zaman içerisinde makinelerin sağlık yönetimi alanında da başarı ile kullanılmaya başlanmıştır [11]. Bir öğrenme probleminde önemli işlem adımlarından biri ham verinin anlamlı özniteliklerinin hesaplanmasıdır. Derin öğrenme yöntemlerinin en önemli avantajı öğrenme aşamasında kullanılacak özniteliklere ihtiyaç duymadan, doğrudan ham veri üzerinde çalışabilmesidir. Ancak bu avantaj, yanında çok yüksek bir işlem yükü getirdiğinden aynı zamanda bir dezavantaja da dönüşebilmektedir. Şekil 1'te geleneksel veri-güdümlü yaklaşımlar ile derin öğrenme yöntemlerinin karşılaştırılması gösterilmiştir.

Makine sağlık verileri büyük çoğunlukla zaman serisi ya da zamana bağlı bir fonksiyon şeklindedirler. Bu tür verileri işleyebilen derin öğrenme yöntemleri özyinelemeli sinir ağlarıdır. Bu çalışmada özyinelemeli derin öğrenme yöntemlerinden Uzun Kısa Vadeli Bellek (LSTM) kullanılmıştır. LSTM yapısının anlaşılabilirliği için öncelikle daha basit bir yapı olan Geçitlendirilmiş Özyinelemeli Birim (GRU) üzerinde durulacak, daha sonra LSTM açıklanacaktır.

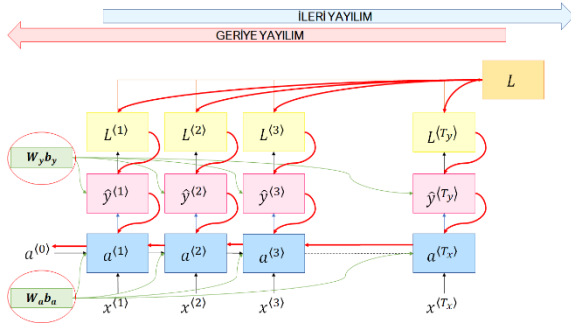
Geçitlendirilmiş Özyinelemeli Birim (GRU) temel olarak geçmişteki bilgilerin mevcut bilgiyi açıklamasında ve gelecek bilginin kestirilmesinde etkili olduğu bir özyinelemeli ağ (Recurrent Neural Networks-RNN) yapısıdır [12]. Özyinelemeli sinir ağları bir dizi veya zamanın bir fonksiyonu olarak ifade edilen problemlerin çözümünde başarılı sonuçlar üretmektedir.

Şekil 2 bir özyinelemeli sinir ağının blok şemasını göstermektedir. Burada $x^{(t)}$ ilgili zamanda a katmanına beslenen giriş değeridir. Ağırlık değerleri W , eşik değerleri b ile ifade edilmektedir. $\hat{y}^{(t)}$ ilgili zamanda her bir nöronun çıkışında hesaplanan kestirimdir ve tüm zaman adımlarında hesaplanan kestirimler kullanılarak yitim değeri $L^{(T)}$ hesaplanır. Tüm zaman adımlarının sonucu kullanılarak genel yitim değeri hesaplanır. Yitim hesabı Denklem (1)'te gösterildiği gibi gerçekleştirilir ve buna bağlı olarak geriye yayılım algoritması ağırlıkları güncelleyerek öğrenme işlemi gerçekleştirilir.

$$L(\hat{y}, y) = \sum_{t=1}^{T_y} L^{(t)}(\hat{y}^{(t)}, y^{(t)}) \quad (1)$$

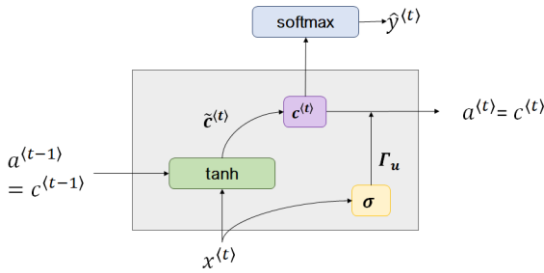


Şekil 1. Geleneksel veri-güdümlü yaklaşımlar ile derin öğrenme yöntemlerinin karşılaştırılması [13]
(Comparison of conventional data driven approaches and deep learning methods [13])



Şekil 2. Özyinelemeli sinir ağının blok şeması
(Block diagram of recurrent neural networks)

Özyinelemeli sinir ağları a katmanlarındaki aktivasyon fonksiyonları ve zamana bağlı işlemlerin yapılış şekline göre sınıflandırılabilir. Bu iç birimlerdeki bağlantı ve aktivasyon işlemleri modelin bellek özelliklerini doğrudan etkilemektedir. Hatırlama ve unutma gibi işlemler bu birimlerde gerçekleştirilir [12]. Geçitlenmiş özyinelemeli birimler (GRU) kullanılabilir en basit temel birimlerden biridir. Şekil 3'te biri GRU blok şeması gösterilmektedir.



Şekil 3. Geçitlendirilmiş özyinelemeli birim blok şeması
(Block diagram of Gated recurrent unit- GRU)

Burada $c^{(t)}$ olarak ifade edilen değer $a^{(t)}$ değerine eşittir ve belleği ifade etmektedir. $\hat{c}^{(t)}$ değeri de bellek için aday değeri simgelemektedir. Aktivasyon fonksiyonu olarak hiperbolik tanjant kullanılmaktadır. Aday değer (2) ile hesaplanır:

$$\hat{c}^{(t)} = \tanh(W_c[\Gamma_r * c^{(t-1)}, x^{(t)}] + b_c) \quad (2)$$

Ayrıca geçitlendirilmiş özyinelemeli birimde güncelleme ve ilgililik geçidi değerleri hesaplanmaktadır. Bu değerler için aktivasyon fonksiyonu olarak sigmoid kullanılmaktadır. Güncelleme geçidi Γ_u ve ilgililik geçidi Γ_r aşağıdaki gibi hesaplanır.

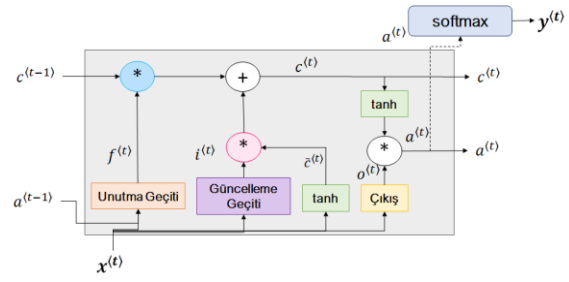
$$\Gamma_u = \sigma(W_u[\hat{c}^{(t-1)}, x^{(t)}] + b_u) \quad (3)$$

$$\Gamma_r = \sigma(W_r[\hat{c}^{(t-1)}, x^{(t)}] + b_r) \quad (4)$$

Γ_u değeri kullanılarak $c^{(t)}$ değeri yeni zaman adımında yeni bir güncelleme olmasa dahi önceki bilgi yardımıyla (5)'daki gibi hesaplanabilir [12].

$$c^{(t)} = \Gamma_u * \hat{c}^{(t)} + (1 - \Gamma_u) * \hat{c}^{(t-1)} \quad (5)$$

LSTM, GRU yapısının özelleştirilmiş bir halidir. LSTM yapısında geçmiş ve gelecekteki bilgiyi anlamlandıracak öznelilikler özyinelemeli olarak taşınmaktadır. Şekil 4'te basit bir LSTM yapısı gösterilmektedir. Bu modelde üç farklı noktada (giriş, hatırlama/unutma, çıkış) aktivasyon fonksiyonu kullanılmaktadır. Giriş ve çıkış katmanlarında genellikle hiperbolik tanjant fonksiyonu, hatırlama/unutma kapılarında ise her zaman sigmoid fonksiyonu kullanılmaktadır. Bu yapının GRU'dan en önemli farkı, ilgililik geçitinin LSTM yapısında özelleşerek unutma (Γ_f) ve çıkış geçidi (Γ_o) olarak iki yeni denklem ile elde ediliyor olmasıdır. Unutma geçidi sayesinde geçmişten aktarılan ancak gerekli olmayan bilgilerin ağırlıklarının azaltılması sağlanmaktadır. Güncelleme geçidi ile unutmanın da etkisi ile daha efektif bir çıkış üretilmektedir [14].



Şekil 4. Temel bir LSTM blok şeması
(Block diagram of basic LSTM)

$$\Gamma_f = \sigma(W_f[a^{(t-1)}, x^{(t)}] + b_f) \quad (6)$$

$$\Gamma_o = \sigma(W_o[a^{(t-1)}, x^{(t)}] + b_o) \quad (7)$$

Böylece hesaplanan yeni $a^{(t)}$ değeri denklem (8)'deki gibidir.

$$c^{(t)} = \Gamma_u * \hat{c}^{(t)} + (\Gamma_f) * \hat{c}^{(t-1)} \quad (8)$$

$$c^{(t)} * \Gamma_o = a^{(t)} \quad (9)$$

4. UYGULAMA ve DENEYLER (APPLICATION AND EXPERIMENTS)

Bu çalışmada uçak motorlarının kalan yaşam ömrünün tahminine dayalı bir uçak sağlık yönetim sistemi önerilmiştir. Çalışmada özyinelemeli bir derin sinir ağı modeli olan LSTM yapısının yukarıda tanımlanan probleme uygun bir çözüm sunabileceği savı üzerinde durulmuş ve LSTM ile elde edilen sonuçlar geleneksel yöntemler ile karşılaştırılmıştır. Denemelerde NASA tarafından hazırlanan C-MAPSS veri kümesinden seçilen 100 adet motorun sensör verileri kullanılmıştır. Verilerin rastgele seçilen 13096 adedi eğitime, 20631 adedi ise test için ayrılmıştır. Problemi bir sınıflandırma problemine dönüştürmek için test verisi elle etiketlenmiştir. Etiketler RUL değeri 150'den küçük olan motorlar bakıma girmeli, diğerleri ise sağlıklı olarak belirlenmiştir.

LSTM modeli, Tesla K80 GPU platform desteği sunan Google Colabs aracı yardımı ile Python 3 programlama dili kullanılarak gerçekleştirilmiştir. Temel kütüphane olarak TensorFlow ve Keras Kütüphaneleri kullanılmıştır. Modelde 3 katmanlı bir LSTM yapısı kurulmuş ve bir seyreltme (dropout) katmanı uygulanmıştır. İlk iki LSTM katmanında 100 LSTM birimi, son katmanda ise 75 birim oluşturulmuştur. Seyreltme işlemi için %50 değeri kullanılmıştır. Yitim hesabında ikili çapraz entropi kullanılmıştır. Çapraz entropi kaybı, çıktısı 0 ile 1 arasında bir olasılık değeri olan bir sınıflandırma modelinin performansını ölçmeye yarar. Tahmin edilen olasılık asıl etiketten uzaklaştıkça çapraz entropi kaybı artar. Çapraz entropi hesabı denklem (10)'da ifade edilmektedir. Burada M sınıf sayısını, y 0-1 olarak sınıfın etiketini, p kestirim sonucunu, s sınıfı, g ise gözlem bilgisini ifade etmektedir.

$$H_{g,s} = - \sum_{s=1}^M y_{g,s} \log(p_{g,s}) \quad (10)$$

Tasarlanan modelde 183.676 adet öğrenilen parametre mevcuttur. Bu parametrelerin en uygun değerleri stokastik bir optimizasyon yöntemi olan Adam Algoritması [15] ile belirlenmiştir. Adam, düşük dereceden matematiksel ifadelerin uyarlanabilir tahminlerine dayanan, stokastik hedef fonksiyonların birinci dereceden gradyan tabanlı optimizasyonu için kullanılmaktadır. Küçük bellek gereksinimlerine sahiptir, gradyanların çapraz yeniden ölçeklendirmesi için değişmezdir. Çok miktarda veri ve parametre hesaplamaları için çok uygundur. Aynı zamanda durağan olmayan problemler ve çok gürültülü ya da seyrek gradyanlarla ilgili problemler için de başarılı sonuçlar veren, literatürde tercih edilen bir yöntemdir [15]. Tasarlanan ağız özeti Tablo 1'de verilmiştir.

Tablo 1. Tasarlanan LSTM modelinin özeti
(Summary of the designed LSTM model)

Katman No	Katman Tipi	Birim sayısı	Parametre Sayısı
1	LSTM	100	50400
2	LSTM	100	80400
3	LSTM	75	52800
4	Seyreltme	75	0
5	Yoğunlaştırma	1	76

Elde edilen sonuçları değerlendirmek için doğruluk, hassasiyet, hatırlama ve F1-skoru metrikleri hesaplanmıştır. Metriklerin hesaplamaları denklem (11-14)'te gösterilmektedir. Bu denklemlerde TP bakım gerektiriyor olarak doğru tanımlanmış örnek sayısını, FP bakım gerektiriyor olarak yanlış tanımlanmış örnek sayısını, TN bakım gerektirmiyor olarak doğru tanımlanmış örnek sayısını ve FN bakım gerektirmiyor olarak yanlış tanımlanmış örnek sayısını göstermektedir. Elde edilen sonuçlar Tablo 2'de özetlenmiştir. LSTM için eğitim ve test yitim-iterasyon grafiği Şekil 5'teki gibi elde edilmiştir. Ayrıca, LSTM'nin regresyon başarımını ölçmek amacıyla eğitim verisinin RUL değeri ile LSTM çıkışında hesaplanan kestirimi arasındaki ortalama mutlak hata (OMH) da ölçülerek Tablo 2'de sunulmuştur.

$$\text{Doğruluk} = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

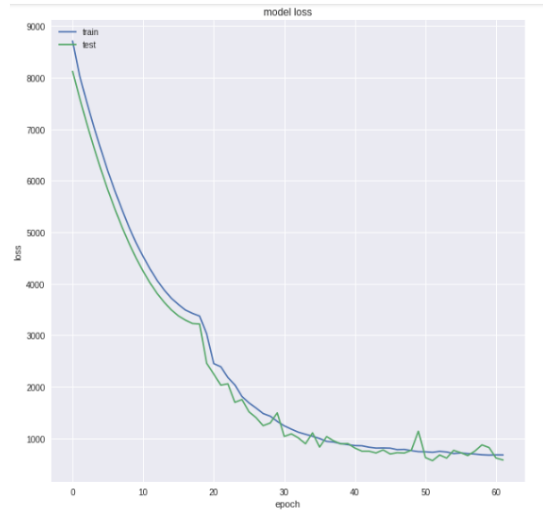
$$\text{Hassasiyet} = \frac{TP}{TP+FP} \quad (12)$$

$$\text{Hatırlama} = \frac{TP}{TP+FN} \quad (13)$$

$$F1 - \text{skoru} = 2 \times \left(\frac{\text{Hassasiyet} \times \text{Hatırlama}}{\text{Hassasiyet} + \text{Hatırlama}} \right) \quad (14)$$

Tablo 2. Tasarlanan LSTM modeli ile elde edilen sonuçlar

Tasarlanan LSTM modeli ile elde edilen sonuçlar	
OMH	1,343
Doğruluk	%98,876
Hassasiyet	%93,882
Hatırlama	%100
F1-skor	%96,84



Şekil 5. LSTM için eğitim ve test kümelerinin yitim fonksiyonlarının iterasyona göre değişimi
(LSTM Model loss for training and test sets)

LSTM ile elde edilen çözümün ne kadar başarılı olduğunu ölçmek amacıyla aynı veri kümesi farklı öğrenme algoritmaları ile işlenmiş ve elde edilen sonuçlar karşılaştırma için kullanılmıştır. Verinin sınıflandırılmasında kullanılan RUL değeri bir zaman serisi oluşturduğu için problem önce bir regresyon problemi gibi çözümlenip daha sonra öngörülen RUL değerine göre sınıflandırma da yapılabilir. Bu çalışmada karşılaştırma için karar ağacı regresyonu, artırılmış karar ağacı regresyonu, Poisson regresyonu, lojistik regresyon ve çok katmanlı almaç (MLP) regresyonu kullanılmıştır. Bu yöntemler Microsoft Azure Machine Learning Studio aracı üzerinde uygulanmıştır.

Bu yöntemlere, ortalama mutlak hata (OMH), ortalama karesel hata (OKH), göreceli mutlak hata (GMH), göreceli karesel hata (GKH) gibi farklı değerlendirme kriterleri

uygulanmıştır. Bu değerlendirme kriterlerinin hesaplanış yöntemleri denklem (15-18)'de gösterilmektedir.

$$OMH = \frac{1}{N} \sum_{i=1}^N |y_i - y_i^*| \quad (15)$$

$$OKH = \sqrt{\frac{1}{N} \sum_{i=1}^N |y_i - y_i^*|^2} \quad (16)$$

$$GMH = \frac{\sum_{i=1}^N |y_i^* - y_i|}{\sum_{i=1}^N |y^* - y_i|} \quad (17)$$

$$GKH = \frac{\sqrt{\sum_{i=1}^N (y_i^* - y_i)^2}}{\sqrt{\sum_{i=1}^N (y^* - y_i)^2}} \quad (18)$$

Burada N test örneklerinin sayısı, y_i bilinen değerler, y^* tüm y 'lerin ortalaması ve y_i^* ise kestirimdir.

Tablo 3. Geleneksel yöntemlerle elde edilen sonuçlar
(Results obtained by traditional methods)

	OMH	OKH	GMH	GKH	Doğruluk
Karar Ağacı Regresyonu	20,63	28,572	0,561	0,472	%74,26
Artırılmış Karar Ağacı Regresyonu	21,28	29,615	0,578	0,507	%76,02
Poisson Regresyonu	23,24	29,973	0,632	0,520	%78
Çok Katmanlı Lojistik Regresyon	26	75,301	0,827	0,773	%92
MLP	17,14	85,689	0,903	0,894	%92,66

Tablo 3 incelendiğinde MLP ve çok katmanlı lojistik regresyon (LR) yöntemlerinin dışındaki yöntemlerin başarımının oldukça düşük olduğu görülmektedir. Görece daha başarılı sonuçlar üretebilen LR ve MLP yöntemleri ile LSTM'nin başarımlarının daha detaylı bir karşılaştırması Tablo 4'te sunulmuştur.

Tablo 4. LR, MLP ve LSTM Uygulaması Sonuçlarının Karşılaştırılması
(Comparison of LR, MLP and LSTM)

LR, MLP ve LSTM Uygulaması Sonuçlarının Karşılaştırılması.			
	LR	MLP	LSTM
OMH	26	17,139	1,343
Doğruluk	%92	%92,667	%98,876
Hassasiyet	%79,96	%89	%93,88
Hatırlama	%88	%89	%100
F1-skor	%83,4	%89	%96,8

5. SONUÇ (CONCLUSION)

Bu çalışmada uçak motorları için bir öngörücü bakım yöntemi önerilmiştir. Önerilen yöntem literatürde yaygın olarak kullanılan bir veri kümesi olan NASA NASA Turbofan Engine Corruption Simulation veri kümesi ile test edilmiştir. Elde edilen sonuçlar, geleneksel makine öğrenmesi algoritmaları ile kıyaslanmıştır. Elde edilen sonuçlar (Tablo 2, 3 ve 4) karşılaştırıldığında LSTM'nin tüm metrikler için en yüksek başarımları verdiği

görülmektedir. Tablo 5, literatürde LSTM ve türevleri ile yapılmış olan çalışmalara ait bazı sonuçlar özetlenmiştir.

Tablo 5. Önerilen LSTM modeli ile geçmişteki çalışmalarla karşılaştırılması
(Comparison of the proposed LSTM model with other studies in the literature)

	Yöntem	OMH
[16]	LSTM	91,42
[18]	LSTM En-De	98,3
[19]	Vanilla LSTM	90
[20]	Deep LSTM	91,7
Önerilen	LSTM	98,657

Malhotra ve çalışma arkadaşları LSTM Kodlayıcı-Kodçözücü tabanlı yeniden yapılandırma modelinin zaman serilerindeki anormallikleri tespit etmek için uygun bir yaklaşım olabileceğini göstermişlerdir [4, 18]. Benzer bir şekilde, [17] ve [18]'da da LSTM yapısının öngörücü bakım için uygun bir yöntem olduğu belirtilmiştir. Bu bilgilerin ışığında, bu çalışmada elde edilen sonuçların literatür ile örtüşmekte olduğu açıktır. Geçmiş çalışmalardaki sonuçlar ile kıyaslandığında bu çalışmada elde edilen sonuçların daha başarılı olduğu görülmektedir.

KAYNAKLAR (REFERENCES)

- [1] V. Fornlöf, **Improved Remaining Useful Life Estimations for On-Condition Parts in Aircraft Engines**, University of Skövde, Sweden, Runit AB, Skövde, ISBN 978-91-981474-9-0, 2016.
- [2] A. Kumar, R. Shankar, L. S. Thakur, "A Big Data Driven Sustainable Manufacturing Framework for Condition-Based Maintenance Prediction", *Journal of Computational Science*, Elsevier B.V., (27) 2018, 2018.
- [3] J. Xu, Y. Wang, Y. Xu, "PHM-Oriented Integrated Fusion Prognostics for Aircraft Engines Based on Sensor Data", *IEEE Sensors Journal*, 14(4), 2014.
- [4] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, G. Shroff, "LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection", **Presented at ICML 2016 Anomaly Detection Workshop**, New York, NY, USA, 2016.
- [5] E. Ramasso, A. Saxena, "Performance Benchmarking and Analysis of Prognostic Methods for CMAPSS Datasets", *International Journal of Prognostics and Health Management*, ISSN 2153-2648, 014, 2014.
- [6] X. Zhang, L. Tang, "Decastro J., Robust Fault Diagnosis of Aircraft Engines: A Nonlinear Adaptive Estimation-Based Approach", *IEEE Transactions on Control Systems Technology*, 21(3), 2013.
- [7] G. S. Babu, P. Zhao, X-L. Li, "Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life", *DASFAA: Database Systems for Advanced Applications*, 214-228, 2016.
- [8] C. Zhang, J. H. Sun, K. C. Tan, "Deep Belief Networks Ensemble with Multi-objective Optimization for Failure Diagnosis", **IEEE International Conference on Systems, Man, and Cybernetics**, 32-37, 2015.
- [9] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, J. Lin, "Machinery health

- prognostics: A systematic review from data acquisition to RUL prediction”, *Mechanical Systems and Signal Processing*, Elsevier Ltd, 104, 799–834, 2017.
- [10] Internet: A. Saxena, K. Goebel, Turbofan Engine Degradation Simulation Data Set, NASA The Prognostics Data Repository, <http://ti.arc.nasa.gov/project/prognostic-data-repository>, NASA Ames Research Center, Moffett Field, CA, 01.05.2018.
- [11] W. Zhang, C. Li, G. Peng, Y. Chen, Z. Zhang, “A Deep Convolutional Neural Network with New Training Methods for Bearing Fault Diagnosis Under Noisy Environment and Different Working Load”, *Mechanical Systems and Signal Processing*, 100, 439–453, 2018.
- [12] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling”, *Advances in Neural Information Processing Systems (NIPS), Deep Learning and Representation Learning Workshop*, 2014.
- [13] S. Khan, T. Yairi, “A review on the Application of Deep Learning in System Health Management”, *Elsevier Journal of Mechanical Systems and Signal Processing*, 107, 241–265, 2018.
- [14] S. Hochreiter, J. Schmidhuber, “Long Short-Term Memory”, *Neural Computation*, 9(8), 1735-1780, 1997.
- [15] P. K. Diederik, B. Jimmy, “ADAM: A Method for Stochastic Optimization”, **3rd International Conference for Learning Representations**, San Diego, 2014.
- [16] M. Yuan, Y. Wu, L. Lin, “Fault Diagnosis and Remaining Useful Life Estimation of Aero Engine Using LSTM Neural Network”, **IEEE International Conference on Aircraft Utility Systems (AUS)**, 135–140, 2016.
- [17] N. Gugulothu, V. Tv, P. Malhotra, L. Vig, P. Agarwal, G. Shroff, “Predicting Remaining Useful Life Using Time Series Embeddings Based on Recurrent Neural Networks”, **2nd ML for PHM Workshop at SIGKDD**, Halifax, Canada, preprint arXiv:1709.01073, 2017.
- [18] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, G. Shroff, “Multi-sensor Prognostics Using an Unsupervised Health Index Based on LSTM Encoder-Decoder”, **1st ACM SIGKDD Workshop on Machine Learning for Prognostics and Health Management**, USA, preprint arXiv:1608.06154, 2016.
- [19] Y. Wu, M. Yuan, S. Dang, L. Lin, Y. Liu, “Remaining Useful Life Estimation of Engineered System Using Vanilla LSTM Neural Networks”, *Journal of Neurocomputing, Elsevier Pub.*, 275, 167-179, 2018.
- [20] R. Zhao, J. Wang, R. Yan, K. Mao, “Machine Health Monitoring with LSTM Networks”, **Tenth International Conference on Sensing Technology**, 2016.
- [21] R. Zhao, R. Yan, J. Wang, K. Mao, “Learning to Monitor Machine Health with Convolutional Bi-Directional LSTM Networks”, *Sensors (Basel)*, (2), 273, doi: 10.3390/s17020273, 2017.
- [22] Internet: M. A. Kızrak, “Uçak Motoru Sağlığı için Uzun-Kısa Süreli Bellek Yöntemi ile Öngörücü Bakım”, GitHub Repo kaynak kodları, <https://github.com/ayyucekizrak>, 30.11.2018.