Araştırma Makalesi

SİBER GÜVENLİKTE KLAVYE DAVRANIŞ ANALİZİ

Nurgül AKŞİT†, Muhammed Ali AYDIN††, Abdül Halim ZAİM‡

† İstanbul Ticaret Üniversitesi, Siber Güvenlik, İstanbul, Türkiye
†† İstanbul Üniversitesi-Cerrahpaşa, Bilgisayar Mühendisliği, İstanbul, Türkiye
‡ İstanbul Ticaret Üniversitesi, Bilgisayar Mühendisliği, İstanbul, Türkiye

nurgul.aksit@gmail.com, aydinali@istanbul.edu.tr, azaim@ticaret.edu.tr

0000-0002-2898-4609, 0000-0002-1846-6090, 0000-0002-0233-064X

Atıf/Citation: AKŞİT, N., AYDIN, M. A., ZAİM, A. ,H., (2022). Siber Güvenlikte Klavye Davranış Analizi, Journal of Technology and Applied Sciences 5(1), 109-122

ÖZET

2019 yılında Çin'de ortaya çıkan ve tüm dünyayı etkisi altına alan Covid-19 salgını ile bilgi sistemleri üzerinde değişen çalışma koşullarını daha güvenli bir ortam haline getirme ihtiyacı artmıştır. Bu ihtiyaç araştırmacıları bilgi sistemlerini kullanan kişinin gerçek kişi olduğuna dair doğrulama sistemi geliştirmeye itmiştir. Geliştirilen Klavye Davranış Analizi programı ile her biri farklı alışkanlıklara sahip kullanıcıların verileri toplanmakta ve belirlenen örnekler derin öğrenme ile yapay zekada kullanmak üzere analiz edilmektedir. Bu analizlerin sonuçları, bilgisayarları ele geçiren kötü niyetli saldırganlar tarafından kullanıldığında kimlik doğrulama yöntemi ile tespitinin yapılması konusunda literatüre katkı sağlamaktadır. Çoklu kimlik doğrulama, kullanıcıların sahip oldukları kimliklerinin farklı kombinasyonlar ile bilgi sistemlerinde onaylanma yöntemidir. Çoklu kimlik doğrulamanın yönü, tekli kimlik doğrulama ile atlatılabilecek sistem açıklıklarının güvenliğini sağlamaktır. Bu çalışmanın amacı, iyi bir derin öğrenme yöntemi ile kullanıcıların klavye davranış analizlerini çıkarmak ve bilgi sistemlerine girişlerde kimlik doğrulaması yapmaktır.

Anahtar Kelimeler: Kullanıcı davranışı analizi, klavye kullanım alışkanlıkları, siber güvenlik önlemleri, siber savunma yöntemleri, makine öğrenmesi, derin öğrenme.

KEYBOARD BEHAVIOR ANALYSIS IN CYBER SECURITY

ABSTRACT

With the Covid-19 epidemic that emerged in China in 2019 and affected the whole world, the need to make the changing working conditions on information systems a safer environment has increased. This need has prompted researchers to develop a verification system that confirms that the person using information systems is a real person. With the developed Keyboard Behavior Analysis program, the data of users with different habits are collected and the determined examples are analyzed for use in artificial intelligence with deep learning. The results of these analyzes contribute to the literature in detecting computers by means of authentication when used by malicious attackers. Multiple authentication is a method of confirming the identities of users in information systems with different combinations. The aspect of multiple authentication is to secure system vulnerabilities that can be circumvented by single authentication. The aim of this study is to analyze the keyboard behavior of the users and to authenticate the logins to the information systems with a good deep learning method.

Keywords: User behavior analysis, keyboard usage habits, cyber security measures, cyber defense methods, machine learning, deep learning.

Geliş/Received	**:**	**17.12.2021**
Gözden Geçirme/Revised	**:**	**02.01.2022**
Kabul/Accepted	**:**	**03.01.2022**

1. GİRİŞ

Sürekli büyüyen ve gelişen teknolojinin yanı sıra, kamu sağlığını tehdit eden Covid-19 hastalığı ile birlikte bulaşıcılığı azaltmak adına insan temasını sınırlayan dijital işlemler daha fazla hayata dahil olmuştur. Online alışveriş işlemleri, dijital eğitim-öğretim faaliyetleri, fiziki olarak kurumda bulunma şartı taşımayan birçok sektörün uzaktan çalışması ile birlikte hızla artan işlemler, tüm kurum ve kuruluşların siber güvenliğin sağlanmasına verdiği önemi daha da artırmıştır. Çalışanlar kurum içerisinde iken kurumun bilgi güvenliği kurallarını daha fazla benimsemekte ve mahremiyet sınırlarını daha rahat çizebilmektedir. Uzaktan çalışmayla kişilerin aile bireyleri ile aynı çalışma ortamlarını paylaşması, kafeterya gibi dışarda çalışma ortamı yaratan mekanlarda bulunması, kurum ve şahsi işlemlerin tek bir bilgisayar üzerinden yapılmaya başlanması gibi durumlar bilgi güvenlikçilerini uzaktan çalışma risklerini değerlendirmeye ve pandeminin siber güvenliğe etkisi üzerinde çalışmalar yapmaya itmektedir. Siber tehditlere karşı teknik güvenlik önlemleri ne kadar sağlansa da zincirin en zayıf halkası olan insan faktörü üzerinde daha fazla durulması gerekmektedir. Düşük farkındalığa sahip çalışanlara eğitim vermek gibi farkındalık artırıcı faaliyetler yapılsa bile teknik önlemler ile birlikte bu kanaldan gelen tehditleri bertaraf etmek gerekmektedir. İnsanlar tarafından yapılan bilgi güvenliğini tehdit eden davranışlar; bilgisayarlarını kilitlemeden masalarından ayrılmaları, uzaktan çalışmanın artması ile kamuya açık alanlarda arkadaşları ile sohbet ederken bilgisayarlarının ekranını açık unutmaları, parolalarını not defterleri gibi herkesin erişebileceği yerlere yazarak temiz masa temiz ekran politikalarına uymamaları ve benzeri durumlar bilgilerin ifşasına sebep olabilecek siber saldırılara zemin hazırlamaktadır. Bu sebeplerden dolayı uzaktan çalışma risklerinin değerlendirilmesi elzem hale gelmiştir. Parolalar çok uzun zamandır bilgisayar sistemlerine erişimin birincil kontrol yöntemi olarak kullanılmakta fakat kullanıcıların kimliklerini doğrulamak için oldukça zayıf bir mekanizmadır. Yapılan araştırmalar kullanıcıların, parolalarını unutmamak için hayatlarında karşılığı olan kavramları içeren basit parolalar seçmeye eğitimli olduğunu göstermektedir (Monrose ve Rubin, 2000). Sıklıkla kullanılan parola sıkılaştırma politikaları hemen hemen tüm sistemlerde aynıdır (Monrose ve ark., 2002). Büyük-küçük harf, en az bir sayı ve özel karakter ile 8-10 karakter uzunluktan oluşmaktadır. Olabilecek tüm parolaların küçük bir alt kümesi ile kolaylıkla tahmin edilebilir niteliktedirler. Bu gibi durumlar bilgi koruyucularını erişim sistemlerine girişte ikincil hatta üçüncül doğrulama yöntemlerini araştırmaya sürüklemektedir.

Biyometrik sistemler, tanımlama ve kimlik doğrulama altyapısında önemli bir rol oynamaktadır. (Rahman ve ark., 2021). Yapılan çalışmayla vurgulanmak istenen nokta; çoklu kimlik doğrulama yöntemlerinin ele geçirilme ihtimalleri yüksek olan SMS ve e-posta kanallarından kurtarıp aynı zamanda parmak izi gibi Kişisel Verilerin Korunması Kanunu (KVKK) tarafından ayrı teknik ve idari önlemler ile değerlendirilen biyolojik verilerin de alınmamasını sağlamaktır. Bu sayede hem gerçek verilerle doğrulama yapılır hem de hassas kişisel verilerin alınması gibi yüksek sorumluluklara girilmez. Bu doğrultuda siber saldırgan, yakından veya uzaktan Personel Computer (Kişisel Bilgisayar-PC)'ı ele geçirmek için klavyeyi kullandığında, geliştirilen program PC kullanıcısının davranışını öğrenmiş olacağından farklı davranışı (siber saldırganın hareketlerini) tespit edebilecektir. Çalışmanın bir sonraki aşamasında farklı davranışı tespit edecek program, ürettiği log ile birlikte merkezi log sistemine alarm oluşturacak ve saldırı teşebbüsünden Security Operation Center (Güvenlik Operasyon Merkezi-SOC) ekiplerini haberdar edebilecektir. Siber güvenlik alt yapısının proaktif olarak yönetildiği kurumlarda alarm ile birlikte ilgili PC'nin karantina Virtual Local Area Network (Sanal Yerel Alan Ağı-VLAN)'a alınması, şirket ağından çıkarılması, internet erişimlerinin kesilmesi veya kullanıcı hesaplarının pasife alınması gibi birçok aksiyonu insan müdahalesi gerektirmeden otomasyonla yapılması sağlanabilecektir.

Çalışmada, klavye verilerinin toplanması ve datasetlerinin oluşturulması için kullanıcı bilgisayarlarına yüklenilmek üzere bir agent programı geliştirilmiştir. Kullanıcılardan toplanan verilerin analiz sonuçları görsel araçlar ile paylaşılmıştır. Derin öğrenme algoritmaları incelenmiş ve LSTM ile analiz yapılmıştır. Gerçek 10 kişiden 3 aylık süre ile alınan klavye hareketleri algoritmaya öğretilerek farklı işlere sahip iki kişinin (yazılımcı, izlemeci) LSTM algoritması ile farklılığının tespiti yapılmıştır. Böylece kullanıcılardan alınan klavye verileri ile makine öğrenmesi yapılarak kişilerin tespit başarısı ölçülmüştür.

Bu araştırmada, kullanıcıların klavye kullanım alışkanlıklarının derin öğrenme ile analiz edilerek kimlik tespitlerinin yapılması amaçlanmıştır. Bunun için ek bir yazılım ve donanım masrafı olmadan kişisel verilerle kullanıcı kimliğini doğrulayan bir yöntem modellenmiştir. Daha önce literatürde yer alan bilimsel çalışmalarda mouse(fare) takibi ve klavye kullanımı ile kimlik doğrulama çalışması yapıldığı görülmüş fakat derin öğrenme algoritmaları ile konuyu inceleyip detaylıca ele alan bir çalışma görülememiştir. Bu makalede kullanıcı alışkanlıkları modellenmiş ve deneysel testler ile analizlerinin sayısal sonuçları gösterilerek bu eksikliğin giderilmesi hedeflenmiştir.

2. İLGİLİ ÇALIŞMALAR

Mondal ve Bours (2017)'un yaptıkları araştırma hem geliştirilen projeye yakın olduğundan hem de diğer makalelere göre daha kapsamlı olduğu için referans alınmış, yazarları ile iletişime geçilerek yeni bir model oluşturulmuştur. İlgili çalışma ilk başta doğrulama yöntemlerinden bahsetmekte ve bu yöntemleri iki tipe ayırmaktadır. Static Authentication (Statik Doğrulama-SA), bu tip yöntemlerde kullanıcı başarılı bir şekilde oturumu açtıysa kullanıcıyı sistemin içine almakta ve oturum süresi dolana kadar sistemlere erişimini sağlamaktadır (Mondal ve Bours, 2017). Continuous Authentication (Devamlı Doğrulama-CA), bu tip yöntemlerde kullanıcı sisteme başarılı bir şekilde giriş yaptıktan sonra ya belli zaman aralıklarında ya da dinamik olarak kullanıcının kimliğini kontrol etmektedir. Kullanıcı tespit edilmediği durumlarda oturum sonlandırılmaktadır (Mondal ve Bours, 2017). Bunlara dayanarak hem daha güvenilir olması hem de tuş verilerinin kullanıcıyı rahatsız etmeden kontrol edilmesi kolay olduğundan devamlı doğrulama tipi uygulanmaya karar verilmiştir.

Ülkemizdeki uzun çalışma saatleri çalışanlarda bir süre sonra yorgunluk yaratabildiğinden, bu da dolaylı olarak klavye kullanımını etkileyebileceğinden false positive (hatalı onaylanmış) alarm oluşmaması için çalışmamızda 1. yöntem amaçlanmış fakat başarılı sonuçlarından dolayı 2. yöntem teknikleri üzerinde durulmuştur. CA tip yöntemlerden bazıları periyodik zamanlar aralıklarında kullanıcı kimliğini kontrol etmektedir. Periyodik yöntemlerde, belirlenen periyodik zaman dilimi içerisinde başka bir kullanıcının (veya saldırgan) sisteme giriş-çıkışı tespit edilememektedir. Bu yüzden, bu çalışmada Periodic Authentication (Periyodik Doğrulama-PA) değil dinamik CA yöntemlerinin kullanılmasına karar verilmiştir.

PA tip yöntemlerinin başarısını ölçmek için kullanılan parametreler (Raul ve ark., 2020):
- FMR: False Match Rate (Yanlış eşleşme oranı)
- FAR: False Acceptance Rate (Yanlış kabul oranı)
- FNMR: False Non Match Rate (Yanlış eşleşmeme oranı)
- FRR: False Rejection Rate (Yanlış reddetme oranı)
- EER: Equal Error Rate (Eşit hata oranı)

Bu parametrelerden Eşit Hata Oranı'nın (EER) değerini düşürme anlamında, tuş vuruşu dinamiklerine dayalı bir kimlik doğrulama algoritmasının verimliliğini artırma olanaklarını analiz eder. Mesafe yöntemi ile, kullanıcılar arasındaki benzerliği hesaplamak için kullanılır (Iapa ve Cretu, 2021).

Dinamik CA tip yöntemlerinin başarısını ölçmek için makalede önerilen parameterler:
- ANIA: Average Number of Imposter Actions (Ortalama taklit işlem sayısı).
- ANGA: Average Number of Genuine Actions (Ortalama gerçek işlem sayısı).

Dinamik CA gerçekleştirmek için dinamik ve esnek bir kontrol sistemine ihtiyaç vardır (Singh, 2018). Makalede önerilen sistem bu şekilde çalışmaktadır. Kullanıcı verilerini alıp öğrendikten sonra bir kullanıcı profili oluşturulmakta ve doğrulama sistemi devreye girmektedir. Her bir tuş verisi (Action) geldikten sonra gelen sayıyı profildeki sayılarla karşılaştırmaktadır. Sayıların benzediği durumda artı puan vermekte (Reward) ve genel kullanıcı puanına eklemektedir. Sayılar benzemediği durumda eksi puan vermekte (Penalty) ve genel kullanıcı puanından çıkartmaktadır. Tek bir formül ile hem Reward hem de Penalty puanları hesaplanabilmektedir (Mondal ve Bours, 2017). Bu modeli kullanarak 49 kullanıcı verisinden alınan ön sonuçlar Tablo 1'deki gibidir (Mondal ve Bours, 2017).

Tablo 1. CA sistemi için geliştirilmiş performans raporlama örneği.

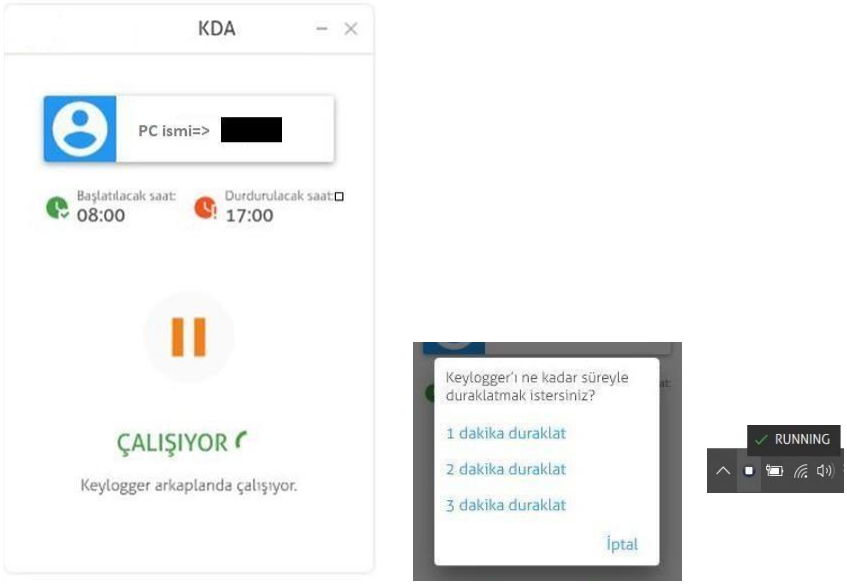
Category(Kategori)	Users (Kullanıcılar)	ANGA	ANIA
+/+	41		99
+/-	4		807
-/+	3	4630	164
-/-	1	90936	512

Summary(Özet)	49	45929	329

Keystroke Dynamics (Tuş Dinamikleri-KD) üzerinde toplanan veriler: Hold time (tutma zamanı) ve seek time (arama zamanı). Referans çalışmada modeli eğitmek için 4 yöntem önerilmiştir (Mondal ve Bours, 2017);
VP-1 internal (iç): Model, her kullanıcı için kurumdaki bütün kullanıcıların verilerini imposter (taklitçi) olarak eğitmektedir.
VP-2 mixed (karışık): Model, her kullanıcı için kurumdaki bazı kullanıcıların verilerine benzer olarak eğitmektedir. Doğrulama aşamasında ise hem daha önce karşılaştığı hem de hiç karşılaşmadığı veriler ile test etmektedir.
VP-3 external (dış): Model, her kullanıcı için kurumdan bağımsız bir kullanıcı datasetin (veri seti) verilerini taklitçi olarak eğitmektedir. Doğrulama aşamasında daha önce hiç karşılaşmadığı veriler ile test etmektedir.
VP-4: KD için VP-3 ve MD VP-1 kullanılmaktadır.

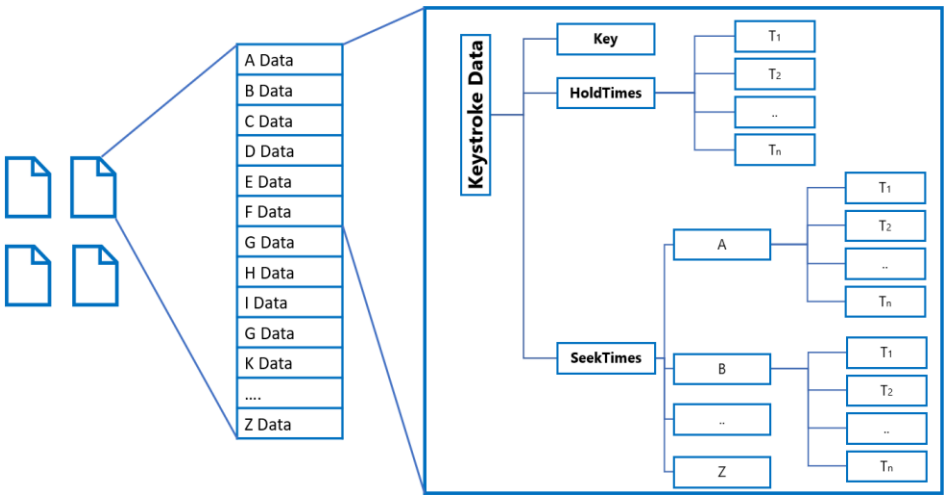
3. MATERYAL VE METOT

Literatür taramalarından sonra aşağıdaki aşamalar kullanılarak çalışmanın genel çerçevesi belirlenmiştir.
Veri Toplama: Kullanıcı alışkanlıklarının tespit edilebilmesi için ilk aşamada veri setlerinin oluşturulması gerekmektedir. Bu veri setlerini oluşturmak için C# tabanlı bir keylogger (tuş kaydedici) program (KDA agent) geliştirilmiştir. Geliştirilen programın çalışma akış diyagramı Şekil 6'da verilmiştir. Bu program kullanıcı bilgisayarlarına yerleştirildikten sonra farklı tip verileri (Tuş basma zamanı ve sıklığı, tuşlar arasında ki geçiş zamanı ve sıklığı, kullanılan programlar ve kullanma süreleri) toplanmış ve ana veri tabanına (bu çalışma kapsamında geliştirici PC) bir Application Programming Interface (Uygulama Programlama Arayüzü-API) üzerinden kaydedilmiştir. Kullanıcılarda kurulu olan programın arayüz görüntüleri Şekil 1'de verilmiştir. Bu süreci kontrol etmek için bir web uygulaması geliştirilmiş, onu kullanarak veri toplama istatistikleri ve oluşacak hatalar izlenebilir hale gelmiştir.



Şekil 1. Kullanıcıda kurulu KDA agent program.

Veri Analizi ve Model Oluşturma: Şekil 2'de gösterilen veri yapısında veri setleri oluştuktan sonra çıkan veriler temizlenip (hata giderme, tekrarları silme vb.) analiz edilmeye başlanmıştır. Analiz ederken bazı teknikleri kullanıp (veri görselleştirme, raporlar çıkarma) öngörülemeyen hatalar veya sonuçların tespiti yapılmıştır. Ardından çıkan sonuçlara göre veri setleri tekrar düzenlenmiş ve derin makine öğrenmesi kullanılarak bir model oluşturulmuştur. Geliştirilen model ile yeni veriler test edilmiş ve başarı oranı ölçülmüştür.



Şekil 2. Veri seti yapısı.

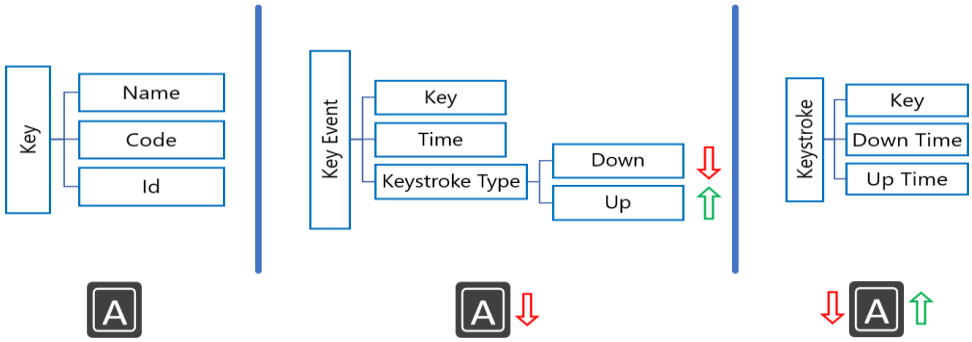
Modeli Kullanacak Yazılımı Geliştirme: Model ortaya çıktıktan sonra kullanıcılarda çalışan yazılıma entegre edilmesi, yazılımın yaptığımız çalışmada kullanıcı alışkanlarını öğrenip ikinci aşamada kullanıcı tespiti yapmaya başlaması ve sürekli bir şekilde log ve data kaydetmesi gelecek çalışma için planlamıştır.

3.1. KDA Agent Çalışma Mantığı

Tuş kaydedici görevi gören bu program kullanıcı bilgisayarlarına yerleştirildikten sonra; Şekil 5'te belirtilen biçimde tuş basma zamanı ile sıklığı, tuşlar arasında ki geçiş sıklığı ve geçiş süreleri toplanmıştır. Tuş vuruş dinamiklerinden yararlanarak kimlik doğrulamada;

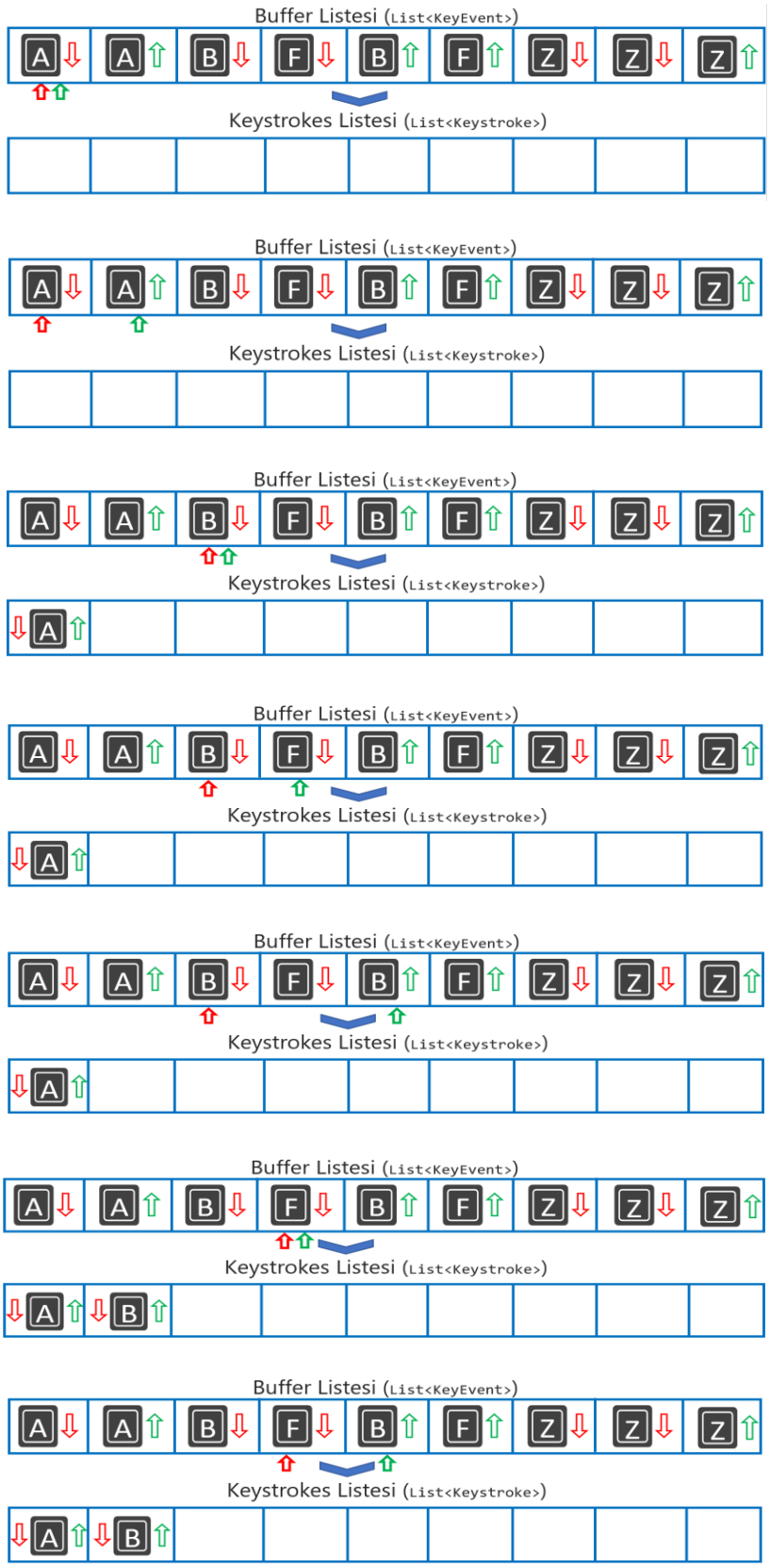
- Key Up: Tuşa basma olayını,
- Key Down: Tuşu bırakma olayını ifade eder (Di Tommaso ve ark., 2019).

Şekil 3'te ana bileşenleri gösterilmiş olan programın çalışma mantığı Şekil 4'te görseller üzerinden adım adım gösterilmiştir.

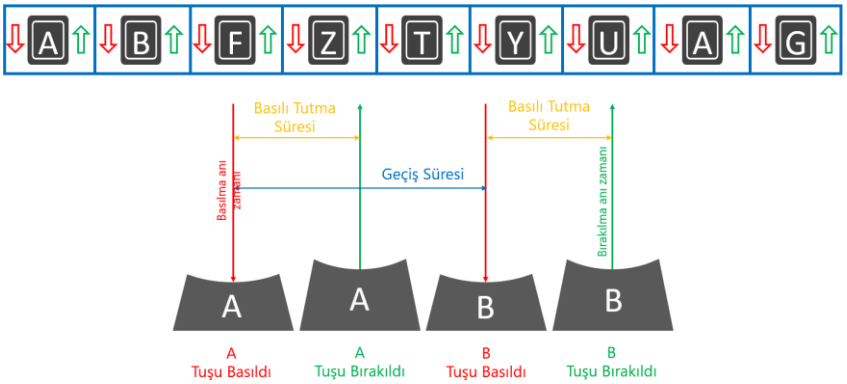


Şekil 3. KDA agent ana bileşenleri.

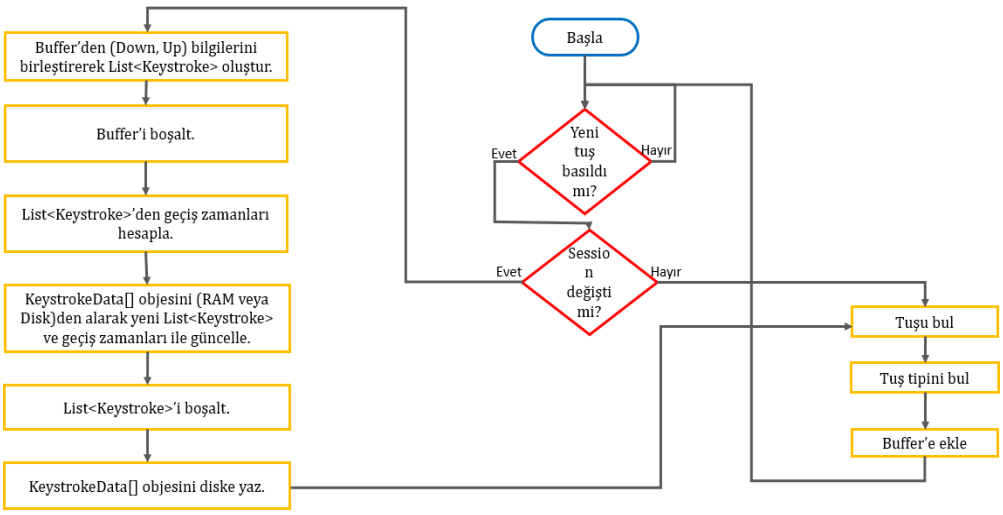
Kullanıcının Şekil 4'te gösterilen tuşlara bastığını ve tuşların sırası ile buffer listesine kaydedildiği varsayılmaktadır. 'A' harfine tuş vuruşu yapılıp (↓) sonra tuş vuruşundan parmağın çekildiği (↑) anı sırasıyla aramakta ve arada geçen zamanı vuruş hızı olarak almaktadır. 'A' harfinden sonra basılan harf iki karakter arasında geçen süre olarak kaydedilmekte olup listedeki tüm tuşlar aynı mantık ile yeni listeye (Keystrokes) dizilmektedir. Şekil 4'te ki görselde iki harf için çalışma şekli resmedilmiştir.



Şekil 4. İki harf için KDA çalışma mantığı**.**



Şekil 5. KDA analyzer topladığı veriler.



Şekil 6. KDA agent (keylogger) çalışma akış diyagramı.

3.2. Veri Tabanı Tasarımı

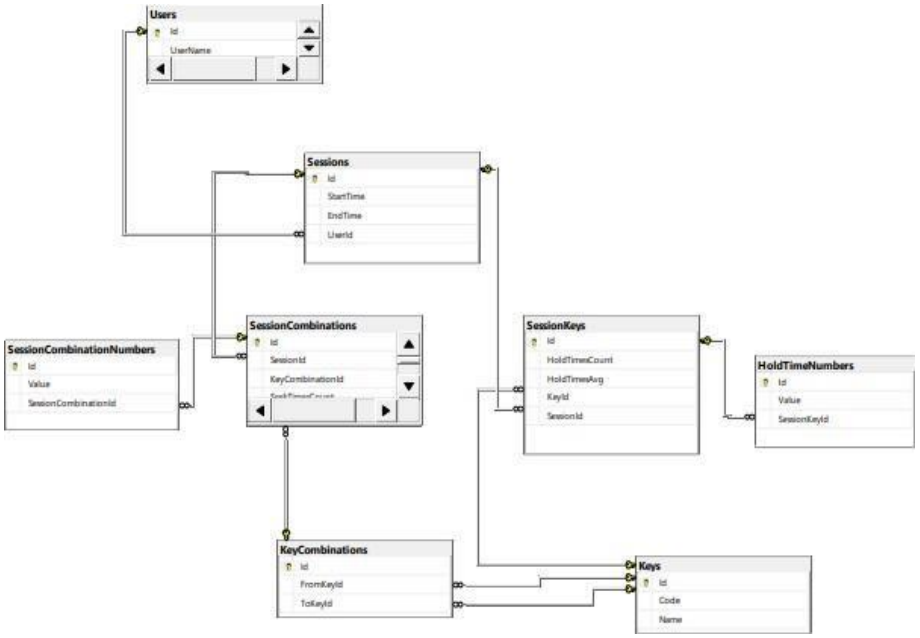
Kullanıcı bilgisayarına yüklenen KDA program ile birlikte toplanacak farklı tipte klavye bilgileri Şekil 8'de şeması verilen veri tabanına eklenmiştir. Veriler şu başlıklar altında toplanmıştır;
Users: Araştırmaya dahil olan kişilerin atanan ID ve isim bilgileri tutulur.
Sessions: Verilerin saatlik olarak saklanması için oluşturulan session bilgileri tutulur.
Keys: Araştırmada verisi tutulan tuşların isimleri ve ID'leri tutulur.
KeyCombinations: İki tuş arasında olabilecek tüm geçiş kombinasyonları, Şekil 7'de gösterildiği şekilde From-To olarak tutulur.

To

From	A	B	C	D	E	F	G	H	...	...
A	A → A	A → B	A → C	A → D	A → E	A → F	A → G	A → H	...	...
B	B → A	B → B	B → C	B → D	B → E	B → F	B → G	B → H	...	...
C	C → A	C → B	C → C	C → D	C → E	C → F	C → G	C → H	...	...
D	D → A	D → B	D → C	D → D	D → E	D → F	D → G	D → H	...	...
E	E → A	E → B	E → C	E → D	E → E	E → F	E → G	E → H	...	...
F	F → A	F → B	F → C	F → D	F → E	F → F	F → G	F → H	...	...
G	G → A	G → B	G → C	G → D	G → E	G → F	G → G	G → H	...	...
H	H → A	H → B	H → C	H → D	H → E	H → F	H → G	H → H	...	...
...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...

Şekil 7. Tuşlar arası geçiş matrisi.

SessionKeys: Bir sessionda basılan tüm tuşların verileri (ilgili tuşa basıldı veya basılmadı bilgisi) tutulur.
HoldTimesNumbers: Her SessionKey için tuşa basılan süreden, tuşu bırakma süresine kadar geçen süre bilgisi tutulur.
SessionCombinations: Bir sessionda yapılan tüm tuşlar arasındaki geçiş bilgisi tutulur.



Şekil 8. Veri tabanı yapısı.

3.3. Verileri Veri Tabanına Kaydeden Kodları Geliştirmek

Kullanıcı bilgisayarından agent ile toplanan verilerin, binary dosyalarından veri tabanına aktarılan kodları ve SQL sorguları oluşturulmuştur. Birinci aşama, verileri veri tabanına kaydeden ve veri tabanından çekilmesini sağlayan, verilerin saklanacağı veri yapısı modeli geliştirilmiştir. İkinci aşama, verileri modele aktardıktan sonra modelleri veri tabanına kaydetmek için bir C# class'ı oluşturulmuştur. Bu class'ta verileri SQL veri tabanına

göndermek için kullanılan dapper kütüphanesine yazılan fonksiyonların kodları yazılmıştır. Üçüncü aşamada toplanan verileri veri tabanına girmek için SQL sorguları stored procedure olarak geliştirilmiştir.

3.4. Derin Öğrenme Kullanarak Kullanıcı Tespiti

Makine öğrenmesi, makinenin büyük veri setlerini kullanarak öğrenmesini sağlar. Derin öğrenme ise bir makine öğrenme yöntemidir. Derin öğrenme sürecinde hedeflenen başarı oranı belirlenen seviyeye ulaşana dek sürekli öğrenme için tekrarlanır. Verilen veri kümesi ile sonuçları tahmin edebilecek yapay zekanın geliştirilmesine imkan sağlar.

3.4.1. Derin Öğrenme Mimarileri

Derin öğrenme algoritmaları içinde en yaygın kullanılanlar, convolutional neural network (konvolüsyonel sinir ağları-CNN), recurrent neural network (tekrarlayan sinir ağları-RNN), long-short term memory (uzun kısa süreli hafıza ağları-LSTM)'dir. Bu algoritmalar ile modellemeler güçlendirilip makine öğrenme adımları yapılmaktadır. Derin öğrenme mimarileri yapay zeka problemlerinin çözümü için pek çok yaklaşım sunmaktadır.

2.4.1.1. CNN (Konvolüsyonel Sinir Ağları)

Makine öğrenmesinde CNN görüntülerin analiz edilmesine başarıyla uygulanmış derin ileri beslemeli yapay sinir ağıdır. CNN ile görüntüyü sınıflandırma, nesneleri tanımlama, görüntü segmentasyonu gibi işlemler başarılı olarak yapılmaktadır. İnsanların görme sistemini örnek olarak benimseyen bu sinir ağları ile yapay zeka sistemlerde, nesnelerin algılanması, tanımlanması ve sınıflandırılması amaçlanmıştır (Tüfekçi ve Karpat, 2019).

2.4.1.2. RNN (Tekrarlayan Sinir Ağları)

Tekrarlayan sinir ağları, düğümler arasındaki bağlantıların yönlendirilmiş bir döngü oluşturduğu yapay sinir ağıdır. Bu ağda asıl amaç ardışık bilgileri kullanmaktır. Gizli katman çıktısını aynı yere girdi olarak gönderen derin öğrenmedir (Doğan ve Türkoğlu, 2019).

2.4.1.3. LSTM (Uzun Kısa Süreli Hafıza Ağları)

LSTM, uzun vadeli bağımlılıkları öğrenebilen özel bir RNN türüdür. İnsan zekasından örnekleyecek olursak uzun süreli bilgileri hatırlamak mücadele ile sürekli öğrendiğimiz değil varsayımsal bir davranışımızdır. LTSM, uzun veya kısa periyotları hatırlar. Farklı türdeki problemlerde çok iyi çalıştıklarından çalışmalarda yaygın bir şekilde kullanılmaktadır (Uzun / Kısa Süreli Bellek, 2017). Tablo 2'de derin öğrenme algoritma platformları kıyaslamalı olarak verilmiştir.

Tablo 2. Derin öğrenme karşılaştırma (Zhu ve ark., 2018).

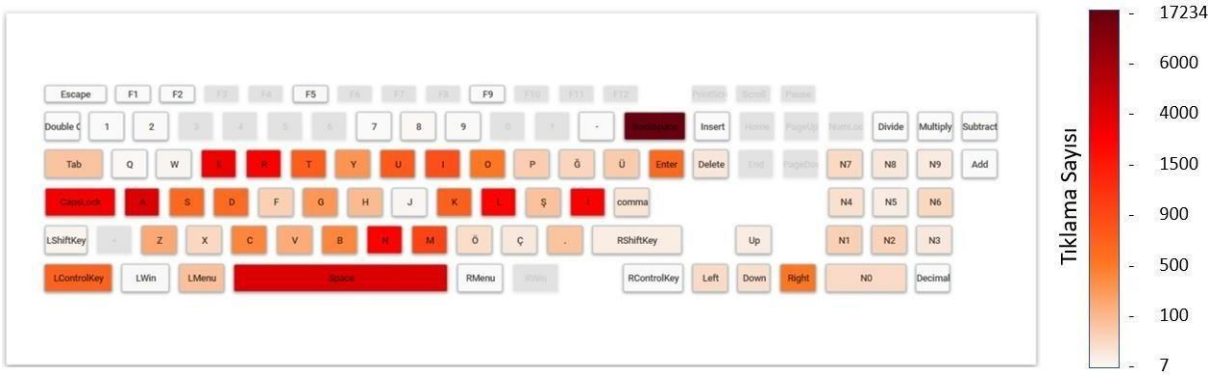
Type(Tip)	Variant (Varyant)	Network structure (Ağ Yapısı)	Applications (Uygulamalar)
CNN	LeNet	Input Layer(Giriş Katmanı) Output Layer(Çıkış Katmanı) Hidden Layer(Gizli Katman)	Image processing(Görüntü İşleme) Speech signal(Konuşma Sinyali) Natural Language (Doğal Dil) Processing(İşleme)
RNN	LSTM	Input Layer(Giriş Katmanı) Output Layer(Çıkış Katmanı)	Time series analysis(Zaman serisi analizi) Emotion analysis(Duygu

		Hidden Layer(Gizli Katman)	analizi) Natural Language Processing

Bir bilgisayarla etkileşim kurmanın en önemli yollarından biri klavye ve faredir. Klavye etkileşimi yalnızca davranışsal verileri (yazma hızı gibi) değil, aynı zamanda bilişsel ve dilsel verileri de içermektedir (Juola ve ark., 2013). Derin öğrenme algoritmaları ile klavye üzerindeki davranışsal verilerin analizinin çıkarılması için yeni modeller geliştirilmiştir.

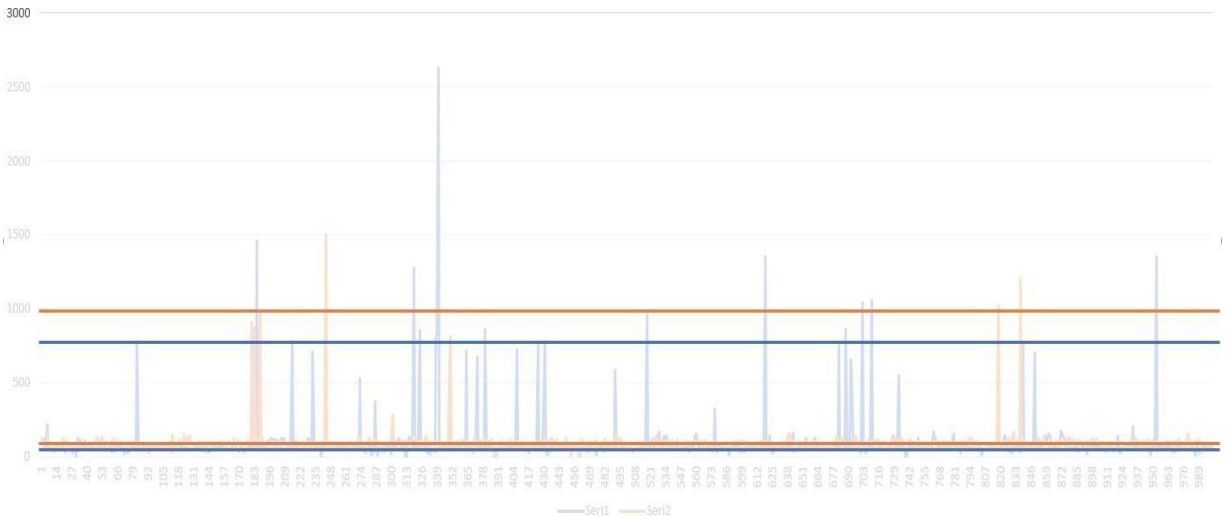
4. BULGULAR

Toplanan veriler ile ilgili doğru sonuçların tahminini yapabilmek için deneme çalışmaları yapılmıştır. Tıklama sayısını görselleştirmek için kodlama yapılmış ve monitoring (izleme) yapan bir çalışana ait tuş vuruş sayısının çıktısı Şekil 9'da gösterilmiştir.



Şekil 9. Tıklama sayısını gösteren araç.

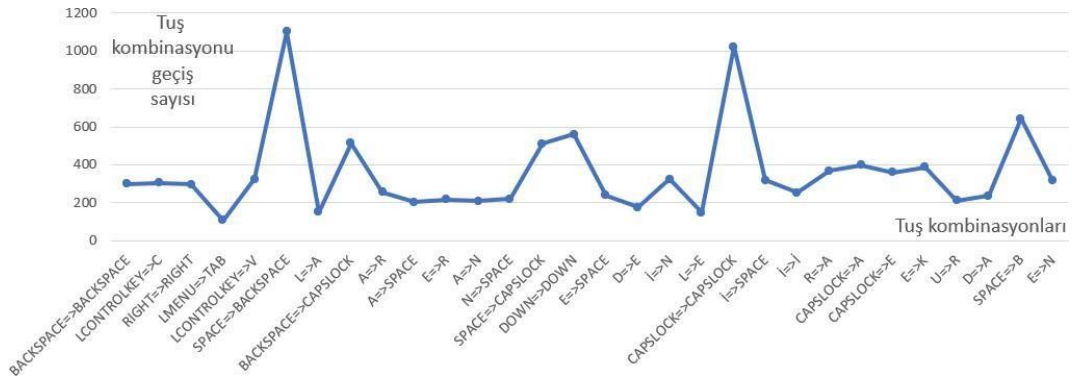
Tıklama sayısı verilerini gözlemledikten sonra tuşların basılı kalma süreleri arasındaki farklar incelenmiştir. Monitoring çalışan (series1) ve kod yazan yazılımcı çalışan (series2) arasındaki backspace ve enter tuşları basılı tutma süre farkı Şekil 10 ve Şekil 11'de gösterilmiştir.



Şekil 10. Iki kişinin backspace tuşu basılı tutma sürelerinin karşılaştırılması.

Şekil 11. İki kişinin enter tuşu basılı tutma sürelerinin karşılaştırılması.

İzleme yapan (series 1) çalışana ait klavye kombinasyonları ve kombinasyonlara kullanma sayıları Şekil 12'de gösterilmiştir.

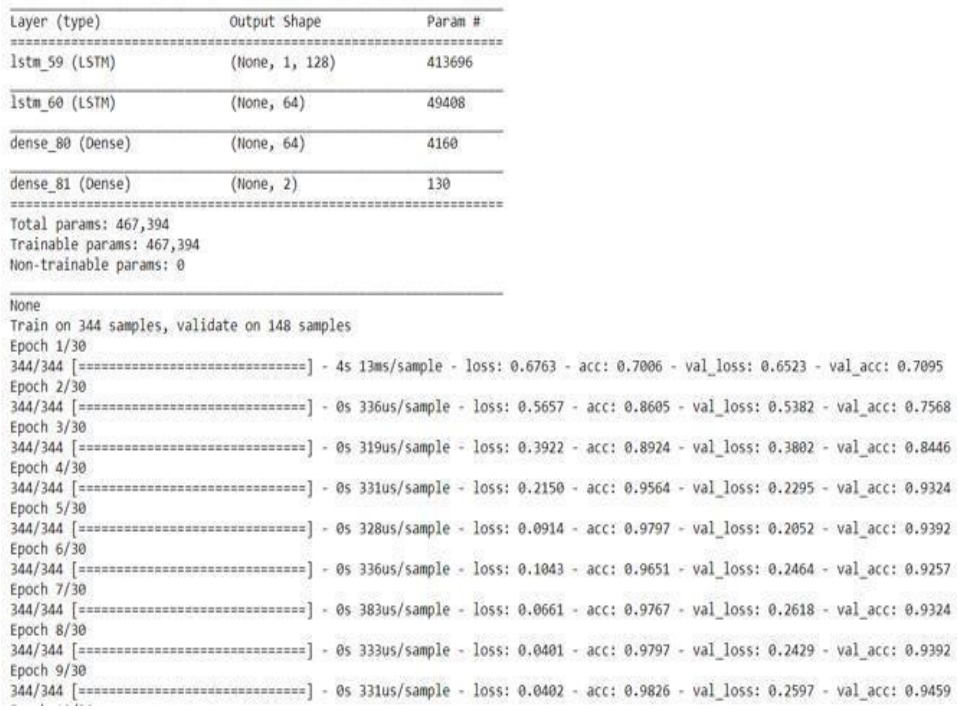


Şekil 12. Series 1 kullanıcı klavye kombinasyonları ve geçiş sayıları.

Çalışmanın odak noktasına uygun yeni bir derin öğrenme modeli oluşturulup farklı davranışlara sahip kullanıcıların klavye tuş vuruşları göz önüne alınmıştır. Bir kullanıcının davranışlarını öğrenen algoritma ile kimlik doğrulama yapmasını ve bir siber saldırganın sisteme giriş denemelerinin tespit edilmesi incelenmiştir. Oluşturulan algoritmaların performansı, araştırmada iki derin öğrenme modeli ile kıyaslanmıştır. RNN ve LSTM algoritmaları denenmiş, sonuçları paylaşılmıştır. Veri setleri 3 ay boyunca teknoloji şirketinde çalışan 10 kişiden geliştirilen tuş kaydedici program ile alınmıştır. Algoritmalar, 10 kişilik veri havuzundan öğrendiği davranışlarla 2 kişinin verisini birbirinden %90 üzerinde ayırabildiğinin çıktısı Şekil 13 ve Şekil 14'te gösterilmiştir.

```
344/344 [==============================] - 0s 177us/sample - loss: 0.0316 - acc: 0.9884 - val_loss: 0.3073 - val_acc: 0.9257
Epoch 21/30
344/344 [==============================] - 0s 194us/sample - loss: 0.0394 - acc: 0.9826 - val_loss: 0.3192 - val_acc: 0.9189
Epoch 22/30
344/344 [==============================] - 0s 183us/sample - loss: 0.0317 - acc: 0.9826 - val_loss: 0.2975 - val_acc: 0.9257
Epoch 23/30
344/344 [==============================] - 0s 203us/sample - loss: 0.0515 - acc: 0.9738 - val_loss: 0.3081 - val_acc: 0.9189
Epoch 24/30
344/344 [==============================] - 0s 194us/sample - loss: 0.0343 - acc: 0.9884 - val_loss: 0.3025 - val_acc: 0.9324
Epoch 25/30
344/344 [==============================] - 0s 185us/sample - loss: 0.0337 - acc: 0.9855 - val_loss: 0.3366 - val_acc: 0.9122
Epoch 26/30
344/344 [==============================] - 0s 209us/sample - loss: 0.0344 - acc: 0.9826 - val_loss: 0.3072 - val_acc: 0.9257
Epoch 27/30
344/344 [==============================] - 0s 200us/sample - loss: 0.0317 - acc: 0.9855 - val_loss: 0.3182 - val_acc: 0.9257
Epoch 28/30
344/344 [==============================] - 0s 191us/sample - loss: 0.0322 - acc: 0.9738 - val_loss: 0.3147 - val_acc: 0.9324
Epoch 29/30
344/344 [==============================] - 0s 183us/sample - loss: 0.0400 - acc: 0.9826 - val_loss: 0.3154 - val_acc: 0.9324
Epoch 30/30
344/344 [==============================] - 0s 193us/sample - loss: 0.0381 - acc: 0.9767 - val_loss: 0.3154 - val_acc: 0.9324
Test loss: 0.31544431812457135
Test accuracy: 0.9324324
```

Şekil 13. RNN algoritma çıktısı.



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_59 (LSTM)               (None, 1, 128)            413696
_________________________________________________________________
lstm_60 (LSTM)               (None, 64)                49408
_________________________________________________________________
dense_80 (Dense)             (None, 64)                4160
_________________________________________________________________
dense_81 (Dense)             (None, 2)                 130
=================================================================
Total params: 467,394
Trainable params: 467,394
Non-trainable params: 0
_________________________________________________________________
None
Train on 344 samples, validate on 148 samples
Epoch 1/30
344/344 [==============================] - 4s 13ms/sample - loss: 0.6763 - acc: 0.7006 - val_loss: 0.6523 - val_acc: 0.7095
Epoch 2/30
344/344 [==============================] - 0s 336us/sample - loss: 0.5657 - acc: 0.8605 - val_loss: 0.5382 - val_acc: 0.7568
Epoch 3/30
344/344 [==============================] - 0s 319us/sample - loss: 0.3922 - acc: 0.8924 - val_loss: 0.3802 - val_acc: 0.8446
Epoch 4/30
344/344 [==============================] - 0s 331us/sample - loss: 0.2150 - acc: 0.9564 - val_loss: 0.2295 - val_acc: 0.9324
Epoch 5/30
344/344 [==============================] - 0s 328us/sample - loss: 0.0914 - acc: 0.9797 - val_loss: 0.2052 - val_acc: 0.9392
Epoch 6/30
344/344 [==============================] - 0s 336us/sample - loss: 0.1043 - acc: 0.9651 - val_loss: 0.2464 - val_acc: 0.9257
Epoch 7/30
344/344 [==============================] - 0s 383us/sample - loss: 0.0661 - acc: 0.9767 - val_loss: 0.2618 - val_acc: 0.9324
Epoch 8/30
344/344 [==============================] - 0s 333us/sample - loss: 0.0401 - acc: 0.9797 - val_loss: 0.2429 - val_acc: 0.9392
Epoch 9/30
344/344 [==============================] - 0s 331us/sample - loss: 0.0402 - acc: 0.9826 - val_loss: 0.2597 - val_acc: 0.9459
```

Şekil 14. LSTM algoritma çıktısı.

LTSM algoritması en başarılı accuracy (doğruluk) değerine 9. epoch (adım) denemede, RNN algoritması ise 30. denemede yaklaşmıştır.

Tablo 3. Algoritma karşılaştırma.

Algorithm(Algoritma)	Epoch (Adım)	Users (Kullanıcılar)	Loss (Kayıp)	Accuracy (Doğruluk)
LSTM	9	10	0.0402	0.9826
RNN	30	10	0.0381	0.9767

Tablo 3'teki değerlere göre iki algoritmanın performans karşılaştırması verilmiştir. LSTM algoritmasının daha başarılı sonuç verdiği doğruluk değerinin 1 sayısına yakınlaşması ile görülmüştür. LSTM ile 1. adımda kayıp değeri 0.6763, doğruluk değeri 0.7006 iken 3.adımdaki kayıp değeri 0.3922 olarak azalmış, doğruluk değeri ise 0.8924 olarak artığı görülmüştür. Algoritmayı eğitmeye devam ettiğimizde 9.adımdan sonra kayıp değeri 0.0402, doğruluk değerimizde 0.9826 değerlerine ulaşmıştır. RNN algoritmasının 21.adımından itibaren incelediğinde kayıp değeri 0.0316, doğruluk değeri 0.9884 olarak görülmüştür. Bu sonuçlar, kayıp değeri 0'a, doğruluk değeri de 1'e yaklaştığında algoritmaların başarılı olarak öğrenme sürecini gerçekleştirdiğini göstermektedir. LSTM 9. adımında kayıp değerini 0'a, doğruluk değerini 1'e, RNN ise 30.adımında bu değerlere yaklaşabilmiştir. Bu kıyas ışığında çabuk öğrenme gösteren LSTM algoritmasının daha başarılı olduğunu görmekteyiz.

5. TARTIŞMA

Tuş vuruş dinamiklerine dayalı parola güçlendirmesini hedefleyen çalışmada, parola güvenliğini artırmak için tuş vuruş dinamikleri kullanılmaktadır (Monrose ve ark., 2002). Buradaki eksiklik derin öğrenme teknikleri ile kimlik doğrulamanın kullanılmamış olmasıdır. Derin öğrenme algoritmaları ile modeller geliştirilerek literatürdeki bu boşluk tarafımızca doldurulmuştur.

Klavye ve mouse vuruşlarını biyometrik kimlik doğrulama yöntemi olarak kullanan çalışmanın veri seti oluşturma aşamasında veri toplama programı tuş verilerini sıralı bir şekilde aldığı için kullanıcıların verileri bilgilere dönüştürülebilir (Mondal ve Bours, 2017). Kullanıcının sistemlere girişi esnasında kullandığı hesap parolaları, alışveriş yaparken girdiği kredi kartı bilgileri ve e-devlet işlemleri sırasında vatandaşlık bilgileri sistem yöneticileri tarafından görülebilir ve bu durum istenmeyen güvenlik açıklıklarına sebep olabilir. Bu açığı kapatmak için çalışmamızda verilerin sırasız bir şekilde tutulabilmesi göz önüne alınarak yeni bir veri yapısı geliştirilip literatüre katkı sunulmuştur.

Tuş vuruşu dinamiği tabanlı kimlik doğrulama mekanizmasını inceleyen çalışmada, bir grup yazarın tuş vuruşları analiz edilerek benzersiz yazma kalıpları olduğu görülmüş ve bu sonuçlardan kullanıcının gerçekliğini doğrulamak için yararlanılabileceği öne sürülmüştür (Raul ve ark., 2020). Burada veri setlerin incelenmesi için örneklem alınan grup sadece aynı yetkinliklere sahip yazarlardan oluşmaktadır. Bunun farklı sektörlerde başarılı sonuçlar vermesi beklenilemez. Bu eksikliği gidermek için tarafımızca farklı tipte kullanıcılardan veri setleri toplanmış ve üzerinde analizler yapılmıştır. İki farklı örneklem kıyaslanmış ve yine literatüre katkı sunulmuştur.

Bir dizi tuş vuruşu özelliği kullanılarak farklı kullanıcılar arasında ayrım yapmak için mantıksal zamansal özelliklerle model kontrolünü kullanan çalışma yapılmıştır (Di Tommaso ve ark., 2019). Destek vektör makineleri algoritması, tanımlama ve doğrulama görevlerinde iyi sonuçlar göstermiştir fakat değerlendirmeleri kendi veri kümelerinden çıkarılan bir dizi kısıtlı özelliklerdir. Bu kısıtı aşmak için çalışmamızda kullanıcılardan tüm klavye tuş ve kombinasyonlarından alınan veri setleri üzerinde analizler yapılmıştır.

Klavye, fare ve web sitesi ziyaret sıklığına bakılarak davranış analizi çıkaran çalışmada, kimlik doğrulama konusunda incelemeler yapılmıştır (Juola ve ark., 2013). Bu çalışmada, veri setlerini toplamak için klavye ve fare üzerine sensörler yerleştirilmiştir. Bu sensörler ekstra donanım ve yazılım maliyeti oluşturduğundan tarafımızca KDA agent adındaki keylogger program geliştirilmiş ve bu sayede klavye hareketleri masrafsız olarak toplanmıştır.

Kişisel verilerin yasal olmayan yollar ile elde edilmesinin artması ve ülkemizde bu konudaki regülatif düzenlemelerle ifşaların daha çok gündeme gelmesi, kullanıcıların klavye hareketlerinin analiz edilmesini istememesine sebep olmuştur. Bu da çalışmanın daha büyük veri setlere ulaşmasında en büyük sorunu oluşturmuştur.

6. SONUÇLAR VE ÖNERİLER

Bu çalışmada kullanıcılardan sağlanan klavye verileri ile makine öğrenmesi yapılarak kişilerin klavye kullanım farklılığının tespit başarısı ölçülüp kıyaslanmıştır. Kullanıcılardan toplanan verilerin analiz sonuçları ile kullanıcının klavye alışkanlıkları tespit edilmiştir. Derin öğrenme algoritmalarından LSTM ve RNN incelenmiş ve analizlerde kullanılmıştır. Bu analizlerin çıktıları klavye kullanımlarının kişilerde benzersiz olduğunu göstermiştir. Kişilerin klavye kullanım şekli kişinin kendisi hakkında bilgi sahibi olmamızı sağlamıştır. Bu bilgi siber savunma yöntemlerinde kullanılabilecek değerli bir parametredir. Küçük bir örneklemde elde edilen sonuçlara göre çalışmanın daha büyük veri setlerle yapılarak kurumlarda kimlik doğrulama yöntemlerinde kullanılabileceği ve bu sayede siber güvenliği artırabileceğine dair literatüre katkı sağlanmıştır.

Sonraki çalışmalar için daha geniş veri setleri elde edilerek, algoritmanın bu veri setleri ile başarılı sonuçları incelenebilir ve model geliştirilebilir. Bu modeller sayesinde siber güvenlik perspektifinden değerlendirilip kişilerin sistemlere girişleri (MFA- Multi Factor Authentication) birden fazla yöntem ile doğrulanabilir. MFA'ya ek olarak, dinamik analiz ile birlikte sürekli takip edilen kullanıcı makinesine fiziken veya uzaktan erişen siber saldırgan farklı klavye davranışı sergileyeceği için programdan alarmlar üretilebilir ve alarmlar merkezi log sistemine gönderilip ilgili güvenlik ekiplerine bilgi verebilir. Bilgisayarlar, kullanıcı tarafından oturum açıldıktan sonra erişim boyunca sistemde aynı kişinin olduğunu varsayar ve kişi oturumu kapatana kadar aynı kişi olduğunu varsaymaya devam eder. Dinamik doğrulama ile anti virüs sistemleri mantığında kullanıcı hareketleri sürekli taranabilir ve doğru kişinin erişimi olup olmadığı kontrol edilebilir. Bu çalışmanın yukarıda bahsedilen alanlarda yapılacak geniş çaplı güvenlik süreçlerine referans olacağı görülmüştür.

KAYNAKLAR

Di Tommaso, F., Guerra, M., Martinelli, F., Mercaldo, F., Piedimonte, M., Rosa, G., & Santone, A. (2019, December). User authentication through keystroke dynamics by means of model checking: A proposal. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 6232-6234). IEEE.

Doğan, F., & Türkoğlu, İ. (2019). Derin öğrenme modelleri ve uygulama alanlarına ilişkin bir derleme. Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi, 10(2), 409-445.

Iapa, A. C., & Cretu, V. I. (2021, May). Modified Distance Metric That Generates Better Performance for the Authentication Algorithm Based on Free-Text Keystroke Dynamics. In 2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics (SACI) (pp. 000455-000460). IEEE.

Juola, P., Noecker, J. I., Stolerman, A., Ryan, M. V., Brennan, P., & Greenstadt, R. (2013). Keyboard-behavior-based authentication. IT Professional, 15(4), 8-11.

Mondal, S., & Bours, P. (2017). A study on continuous authentication using a combination of keystroke and mouse biometrics. Neurocomputing, 230, 1-22.

Monrose, F., & Rubin, A. D. (2000). Keystroke dynamics as a biometric for authentication. Future Generation computer systems, 16(4), 351-359.

Monrose, F., Reiter, M. K., & Wetzel, S. (2002). Password hardening based on keystroke dynamics. International journal of Information security, 1(2), 69-83.

Rahman, A., Chowdhury, M. E., Khandakar, A., Kiranyaz, S., Zaman, K. S., Reaz, M. B. I., ... & Kadir, M. A. (2021). Multimodal EEG and keystroke dynamics based biometric system using machine learning algorithms. IEEE Access, 9, 94625-94643.

Raul, N., Shankarmani, R., & Joshi, P. (2020). A comprehensive review of keystroke dynamics-based authentication mechanism. In International Conference on Innovative Computing and Communications (pp. 149-162). Springer, Singapore.

Singh, S. (2018, January). Keystroke Dynamics for Continuous Authentication. In 2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 205-208). IEEE.

Tüfekçi, M., & Karpat, F. (2019). Derin Öğrenme Mimarilerinden Konvolüsyonel Sinir Ağları (CNN) Üzerinde Görüntü İşleme-Sınıflandırma Kabiliyetininin Arttırılmasına Yönelik Yapılan Çalışmaların İncelenmesi.

Uzun / Kısa Süreli Bellek (Long / Short Term Memory). (2017, September 26). Veri Bilimcisi. https://veribilimcisi.com/2017/09/26/uzun-kisa-sureli-bellek-long-short-term-memory/. Erişim Tarihi: 10.08.2021.

Zhu, N., Liu, X., Liu, Z., Hu, K., Wang, Y., Tan, J., ... & Guo, Y. (2018). Deep learning for smart agriculture: Concepts, tools, applications, and opportunities. International Journal of Agricultural and Biological Engineering, 11(4), 32-44.

Not: *Bu makale, İstanbul Ticaret Üniversitesi Fen Bilimleri Enstitüsü, Siber Güvenlik Tezli Yüksek Lisans Programı'nda, Muhammed Ali AYDIN ve Abdül Halim ZAİM danışmanlığında, Nurgül AKŞİT tarafından yürütülecek olan, "SİBER GÜVENLİKTE KLAVYE DAVRANIŞ ANALİZİ" başlıklı yüksek lisans tezinin ön çalışmalarından yararlanılarak hazırlanmıştır.*