Journal of the Faculty of Engineering and Architecture of Gazi University 38:3 (2023) 1919-1929

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

du-CBA: Data-agnostic and incremental classification-based association rules extraction architecture

Büşra Büyüktanır[1]*, Kazım Yıldız[1], Eyüp Emre Ülkü[1], Tolga Bütüktanır[2,3]

[1]Department of Computer Engineering, Faculty Technology, Marmara University, 34854, Maltepe, Istanbul, Türkiye
[2]Loodos Tech,. R&D Office, 34485, Maslak, Istanbul, Türkiye
[3]Department of Computer Engineering, Faculty Electric-Electronics, Yıldız Teknik University, 34220, Esenler, Istanbul, Türkiye

Highlights:

- Machine learning model training in systems where clients and servers require to work together without sending raw data from clients to the server
- Protection of data privacy with training machine learning model without exporting the raw data from the clients
- Updating the machine learning model with incremental learning from new incoming data on the client

Keywords:

- Federated learning
- Data-unaware machine learning
- Data privacy
- CBA
- Associative classification

Article Info:

Research Article
Received: 14.03.2022
Accepted: 17.09.2022

DOI:

10.17341/gazimmfd.1087746

Correspondence:

Author: Büşra Büyüktanır
e-mail: busra.buyuktanir@marmara.edu.tr
phone: +90 553 843 4953

Graphical/Tabular Abstract

In this study, federated learning architecture is used for training machine learning models without sending raw data from clients to the server in systems where clients and servers need to work together. According to the architecture, a machine learning model is trained on each client from its own data. The trained model is sent to the server and a new model is created by merging these models on the server. The final model created is distributed to the clients again. In order to realize the proposed architecture in the simulation, an algorithm called Data Unaware Classification Based on Association (du-CBA) has been developed. Experimental results showed that the model training time was reduced by approximately 70% with du-CBA compared to CBA, providing almost the same accuracy. The working logic of the federated learning architecture is shown in Figure A.

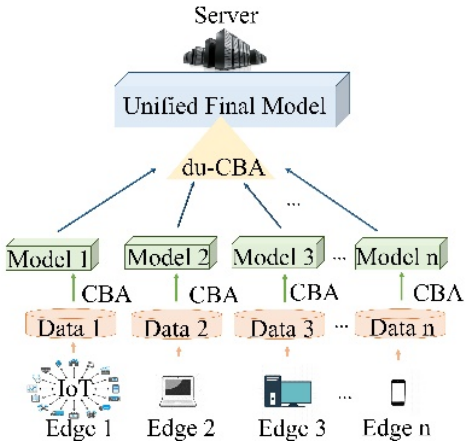


Figure A. The working structure of the federated learning architecture

Purpose: The aim of the study is to create an up-to-date model that provides data privacy by using machine learning methods for edge devices (phones, computers, IoT devices, etc.) whose place in our lives is gradually increasing.

Theory and Methods: The du-CBA algorithm was developed in the simulation environment to implement the federated learning architecture for the study. The model consisting of labeled association rules is trained with the algorithm. Support and trust parameters are used when creating rules. The process of combining the models in the algorithm is realized by updating the support and confidence values of the rules and reordering them. Formulas have been developed to update support and trust values.

Results: As a result of the experiments, it has been shown that the du-CBA algorithm developed for the federated learning architecture reduces the model training time by approximately 70% compared to the CBA algorithm and achieves almost the same accuracy. These results show that the proposed architecture has been successful.

Conclusion: The study shows that with the federated learning architecture, network traffic is reduced, energy needs are reduced, and data privacy is protected because meaningless data is sent instead of all data.

Journal of the Faculty of Engineering and Architecture of Gazi University 38:3 (2023) 1919-1929

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

du-CBA: Veriden habersiz ve artırımlı sınıflandırmaya dayalı birliktelik kuralları çıkarma mimarisi

Büşra Büyüktanır[1]*, Kazım Yıldız[1], Eyüp Emre Ülkü[1], Tolga Bütüktanır[2,3]

[1]Marmara Üniversitesi, Teknoloji Fakültesi, Bilgisayar Mühendisliği Bölümü, 34854, Maltepe, İstanbul, Türkiye
[2]Loodos Tech, ARGE Ofisi, 34485, Maslak, İstanbul, Türkiye
[3]Yıldız Teknik Üniversitesi, Elektrik-Elektronik Fakültesi, Bilgisayar Mühendisliği Bölümü, 34220, Esenler, İstanbul, Türkiye

ÖNEÇIKANLAR

- Federe öğrenme mimarisi üzerine çalışma yapılmıştır
- Mimari kullanılarak du-CBA isimli yeni algoritma geliştirilmiştir
- Algoritmanın benzetim ortamında başarısı ölçülmüş ve kıyaslanmıştır

Makale Bilgileri
Araştırma Makalesi
Geliş: 14.03.2022
Kabul: 17.09.2022

DOI:
10.17341/gazimmfd.1087746

Anahtar Kelimeler:
Federe öğrenme,
veriden habersiz makine öğrenmesi,
veri mahremiyeti,
CBA, ilişkisel sınıflandırma

ÖZ

İstemci sunucu sistemlerinde makine öğrenmesi modeli kullanılması bir ihtiyaçtır. Ancak istemcilerden verilerin toplanması, sunucuya aktarılması, makine öğrenmesi modeli eğitilmesi ve bu modelin istemcilerde çalışan cihazlara entegre edilmesi bir çok problemi beraberinde getirmektedir. Verilerin istemcilerden sunucuya transferi ağ trafiğine sebep olmakta, fazla enerji gerektirmekte ve veri mahremiyetini istismar edilebilmektedir. Çalışma kapsamında, bahsedilen problemlere çözüm için federe öğrenme mimarisi kullanılmaktadır. Mimariye göre, her bir istemcide istemcinin kendi verilerinden makine öğrenmesi modeli eğitilmektedir. Her bir istemcide eğitilen modeller sunucuya gönderilmekte ve sunucuda bu modeller birleştirilerek yeni bir model oluşturulmaktadır. Oluşturulan nihai model tekrar istemcilere dağıtılmaktadır. Bu çalışmada Veriden Habersiz İlişkili Kurallara Dayalı Sınıflandırma (Data Unaware Classification Based on Association, du-CBA) olarak adlandırılan ilişkisel sınıflandırma algoritması geliştirilmiştir. Federe öğrenme ile klasik öğrenme mimarilerini karşılaştırıp başarılarını ölçmek için çalışma kapsamında benzetim ortamı oluşturulmuştur. Benzetim ortamında du-CBA ve CBA algoritmaları kullanılarak modeller eğitilmiş ve sonuçlar kıyaslanmıştır. Modellerin eğitiminde University of California Irvine (UCI) veri havuzundan alınan beş veri seti kullanılmıştır. Deneysel sonuçlar, her bir veri seti için federe öğrenme ile eğitilen modellerin, klasik öğrenme ile eğitilen modellerle neredeyse aynı doğruluğu elde ettiğini ama eğitim sürelerinin yaklaşık %70 oranında azaldığını göstermiştir. Sonuçlar geliştirilen algoritmanın başarıya ulaştığını ortaya koymaktadır.

du-CBA: Data-agnostic and incremental classification-based association rules extraction architecture

HIGHLIGHTS

- A study was conducted on the Federated earning architecture
- A new algorithm named du-CBA was developed using the architecture
- The success of the algorithm in the simulation environment was measured and compared

Article Info
Research Article
Received: 14.03.2022
Accepted: 17.09.2022

DOI:
10.17341/gazimmfd.1087746

Keywords:
Federated learning,
data-unaware machine learning,
data privacy,
CBA, associative classification

ABSTRACT

It is a necessity to use machine learning model in client server systems. However, collecting data from the clients, transferring them to the server, training the machine learning model and integrating this model into the devices running on the clients bring along many problems. The transfer of data from the clients to the server causes network traffic, requires a lot of energy, and data privacy can be abused. Within the scope of the study, federated learning architecture is used to solve the mentioned problems. According to the architecture, the machine learning model is trained on each client from the client's own data. Models trained on each client are sent to the server and a new model is created by merging these models on the server. The final model created is distributed to the clients again. In this study, a relational classification algorithm called Data Unaware Classification Based on Association (du-CBA) was developed. In order to compare federated learning and classical learning architectures and measure their success, a simulation environment was created within the scope of the study. Models were trained using du-CBA and CBA algorithms in the simulation environment and the results were compared. Five data sets from the University of California Irvine (UCI) repository were used to train the models. Experimental results showed that for each dataset, the models trained with federated learning achieved almost the same accuracy as the models trained with classical learning, but the training times were decreased by about 70%. The results show that the developed algorithm has been successful.

*Sorumlu Yazar/Yazarlar / Corresponding Author/Authors : *busra.buyuktanir@marmara.edu.tr, kazim.yildiz@marmara.edu.tr, emre.ulku@marmara.edu.tr,tolga.buyuktanir@loodos.com / Tel: +90 553 843 4953

1. Giriş (Introduction)

Son zamanlarda, nesnelerin interneti, mobil cihazlar ve iş birliği içinde çalışan istemci-sunucu sistemler gibi uçlarda (edge) çalışan sistemlerde önemli teknolojik gelişmeler yaşanmaktadır. Bu gelişmeler, uçlarda üretilen verilerin yönetilmesi ve verilerden çıkarım yapılması gerekliliğini doğurmaktadır. Verilerden çıkarım yapılması için uçlarda makine öğrenmesi temelli çözümler kullanılmaktadır. Uçta çalışan bir cihazda makine öğrenmesi modelinin kullanılması elbette mümkündür [1-3] ancak öncelikli olarak modelin eğitilmesi gerekmektedir.

Eğitim gerçekleştirilirken veri seti içindeki öz nitelikler ve eğitim yöntemi uçlarda çalışan bütün cihazlar için aynı olmalıdır. Mevcut sistemlerde uçlarda toplanan veriler merkezde bulunan bir sunucuya aktarılmakta ve sunucu üzerinde makine öğrenmesi modelleri eğitilmektedir. Eğitilen model, sunucu üzerinde tutularak veya uçlara tek tek gönderilerek uçta çalışan sistemlere özellik katılmaktadır. Modelin sunucuda tutulması, uç ve modelin haberleşmesi için ağ bağlantısının sürekliliğini gerektirmektedir.

Uçta çalışan sistemlere makine öğrenmesi modellerinin entegre edilmesi mümkün olabilse de birçok dar boğaz meydana gelmektedir. Makine öğrenmesi modelinin eğitilmesi için sistemde bulunan bütün uçlardan sunucuya yeterli miktarda veri toplanması gerekmektedir. Bunun için oldukça fazla ağ trafiğine ve enerjiye ihtiyaç duyulmaktadır. Bu birçok uçta çalışan sistem için problem oluşturabilmektedir. Uçlarda çalışan sistemlerdeki bütün verilerin sunucuya aktarılması veri mahremiyetini de istismar edebilmektedir [4].

Yapılan çalışma kapsamında, istemci-sunucu sistemlerinde makine öğrenmesi modelinin eğitilmesi sürecinde veri mahremiyetini koruyan, ağ trafiğini azaltan ve enerji ihtiyacını düşüren güncel bir teknoloji olan federe öğrenme mimarisi kullanılmaktadır [5]. Federe öğrenme akıllı cihazlar, mobil uygulamalar gibi istemci sunucu sistemlerinde kullanılmak üzere geliştirilmiştir. Klasik makine öğrenmesi gibi istemcilerde bulunan verileri sunucuda toplayıp, sunucuda model eğitimi yapmamaktadır. Bunun aksine her istemci için sadece kendinde bulunan veriler ile modeller oluşturulmakta ve sunucuya bütün veri yerine sadece model gönderilmektedir. Sunucuda toplanan modeller birleştirilerek yeni bir model oluşturulmakta ve tekrar istemcilere gönderilmektedir. İstemcilerde yeni verilerin toplanması ile yeni modeller eski modellerle birleştirilebilmekte ve yine sunucuya gönderilerek orada tekrar birleştirme işlemi gerçekleştirilebilmektedir. Verinin bulunduğu alanda işlenmesi ile veri mahremiyetinden ödün verilmemekte, veri güvenliği sağlanmaktadır. Ayrıca artırımlı öğrenme de gerçekleştirilmekte ve yeni gelen veriden dolayı her defasında bütün veriyle eğitim gerçekleştirilmemektedir.

Güncel bir konu olan federe öğrenme mimarisi için geliştirilecek algoritmalara ihtiyaç duyulmaktadır. Bu çalışma kapsamında federe öğrenme mimarisine ait bir benzetim ortamı oluşturulmuştur. Benzetim ortamında modellerin eğitimi iki farklı mimari yöntem ile gerçekleştirilmiş ve elde edilen modeller kıyaslanmıştır. İlk model eğitiminde klasik öğrenme yöntemi kullanılmıştır, yani modeler, istemcilerden sunucuya verilerin aktarılmasıyla ve bu verilerden model eğitilmesiyle elde edilmiştir. İkinci model eğitiminde ise federe öğrenme mimarisi kullanılmıştır. İstemcilerde model eğitilip sunucuda modellerin birleştirilmesi ile nihai model elde edilmiştir. Klasik öğrenme ile model eğitimini gerçekleştirmek için ilişkisel sınıflandırma algoritmalarından biri olan CBA [6] kullanılmıştır. Federe öğrenme mimarisini benzetimde gerçekleştirmek için ise çalışma kapsamında Veriden Habersiz İlişkili Kurallara Dayalı Sınıflandırma (Data Unaware Classification Based on Association, du-CBA) olarak adlandırılan algoritma geliştirilmiştir. UCI veri havuzundan alınan beş veri seti ile du-CBA ve CBA algoritmaları kullanılarak ayrı ayrı modeller eğitilmiş ve bu modeler kıyaslanmıştır. Deneysel sonuçlar, du-CBA kullanılarak oluşturulan modelin eğitim süresinin, CBA ile oluşturulan modelin eğitim süresinden neredeyse %70 oranında daha az olduğunu göstermiştir. Oluşturulan modellerin doğruluğu kıyaslandığında kayda değer fark oluşmamıştır. Bu sonuçlar, federe öğrenme mimarisi kapsamında geliştirilen algoritmanın başarıya ulaştığını ortaya koymaktadır. İstemcilerden sunucuya bütün veriler yerine sadece modelin gönderilmesiyle ağ trafiği azaltılmıştır. Her bir istemciden sunucuya veriler yerine, istemcilerde eğitilen modeller gönderilmektedir. Sunucuda birleştirilen modeller istemcilere geri gönderilmektedir. Veriler yerelden çıkarılmadığı için veri mahremiyeti korunmaktadır. Her bir istemcide çalışan model ise daha önce o istemcide üretilmemiş veriler ile eğitilmiş olmaktadır.

Makalenin geri kalanı şu şekilde organize edilmiştir: Bölüm 2'de literatürde yer alan konuyla alakalı çalışmalara yer verilmektedir. Çalışmadaki motivasyonumuz ve problem tanımı Bölüm 3'de, takip edilen metodoloji ise Bölüm 4'te açıklanmaktadır. Bölüm 5'de deney ortamı ve deneysel sonuçlar ayrıntılı olarak yer almaktadır. Çalışma sonunda elde edilen sonuçlar ve gelecekte yapılması planlanan çalışmalar Bölüm 6'da yer anlatılmaktadır.

2. İlgili Çalışmalar (Related Works)

Yapay zeka ve uygulamaları günümüzde pek çok alanda olduğu gibi istemci - sunucu olarak çalışan uç cihazlarda da kendini göstermektedir [2]. Otonom araçlardan [7] sanal asistanlara [8], nesnelerin internetinden (IoT) [9] mobil cihaz uygulamalarına [10] ve benzeri birçok alanda insanlara yardımcı olabilmek amacıyla çeşitli uygulamalar geliştirilmektedir. Geliştirilen bu uygulamalardan doğru ve güvenilir sonuçlar üretilmesi beklenmektedir. Doğru sonuca hızlı bir şekilde ulaşmak, veri mahremiyetinden ödün vermemek ve değişen talepleri karşılamak uygulama kullanıcılarının memnuniyetini arttırmaktadır. Tüm bu nedenlerden dolayı, yapay zeka uygulamalarında kullanılan modellerin daha doğru ve daha hızlı çalışabilmesi için, yeni model eğitme yöntemleri geliştirilmekte, makine öğrenmesi metotları üzerinde iyileştirmeler yapılmaktadır. Yeni bir makine öğrenmesi algoritması imar edilmekte veya mevcut algoritma üzerinde düzenlemeler yapılmaktadır. Ayrıca bu algoritmalar ile eğitilen modellerin güncel kalıp değişen taleplere adaptasyon sağlayabilmesi için artırımlı öğrenme yöntemleri geliştirilmektedir.

Son yıllarda birliktelik kurallarına dayalı olarak sınıflandırma yapan yeni bir veri madenciliği yöntemi ortaya çıkmıştır [11]. İlişkisel sınıflandırma (Associative classification) adı verilen bu yöntemde, oluşturulan kurallar etiketlenerek sınıflandırılır ve kullanıcı tarafından daha kolay yorumlanır. Yeni oluşturulan bu yöntem ile birlikte, yeni bir algoritma da ortaya çıkmıştır. Önerilen algoritma, birliktelik kurallarına dayalı sınıflandırma (classification based on association rules, CBA) olarak adlandırılmaktadır. Algoritmanın çalışma mantığı üç aşamadan oluşmaktadır: Veriler ile kural oluşturma ilk aşamadır. Zayıf kuralların budanma işlemi ikinci aşamadır. Son aşama ise en iyi sınıflandırma yapan kuralların elde edilmesi aşamasıdır. Çoklu sınıf ilişkilendirme kurallarına dayalı sınıflandırma (Classification based on multiple class-association rules, CMAR) [12] ve birliktelik kuralına dayalı çok sınıflı sınıflandırma (multiclass classification based on association rule, MCAR) [13] adı verilen algoritmalar da ilişkisel sınıflandırma yöntemi için geliştirilen algoritmalardır. Bahsedilen bu algoritmaların çalışma mantığı ortaktır ancak kural oluşturma süreçleri birbirinden farklıdır. Bir diğer ortak özellikleri de

geleneksel sınıflandırma yöntemlerinden biri olan karar ağaçlarından daha doğru sınıflandırma başarısına sahip olmalarıdır. İlişkisel sınıflandırma metodunda kullanılan CBA algoritması üzerinde, modelin doğruluk değerini ve eğitim süresini arttıran iyileştirmeler yapılarak; gelişmiş CBA (enhanced CBA, ECBA) [14], birliktelik kuralına göre hızlı sınıflandırma (fast classification based on association rule, FCBA) [15] ve birliktelik kurallarına dayalı ağırlıklı sınıflandırma (weighted classification based on association rules, WCBA) [16] isimli algoritmalar geliştirilmiştir. Fakat bu algoritmalar ile geliştirilen modeller güncelliğini koruyamamaktadır. Sürekli değişen talepler neticesinde veriler değişime uğrar. Eğitilen modellerin de doğru ve hızlı çalışmasının yanı sıra bu değişime uyum sağlaması beklenir. Değişen ve sürekli gelen veri setlerine modelin uyum sağlayarak güncel kalmasına artırımlı öğrenme (incremental learning) denilmektedir [17]. Model sınırlı bellek kullanımı ile doğruluğundan ödün vermeden öğrenme işlemine devam etmektedir. Artırımlı öğrenmenin tek seferde öğrenmeden farkı, modelin gelen her yeni bilgi ile kendini güncellemesidir [18]. Bu alanda birliktelik kurallarına dayalı artırımlı sınıflandırma algoriması (incremental classification based on association rules algorithm, I-CBA) [19] geliştirilmiştir. Eski eğitim veri kümesine yeni eğitim veri kümesi eklendiğinde, sınıflandırma sistemi oluşturmak için ilişkisel sınıflandırma yöntemi üzerine artırımlı bir güncelleme tekniği uygulanmıştır. I-CBA algoritması, CBA aşamalarına hızlı güncelleme algoritması (fast update algorithm, FUP) uygulanması ile geliştirilmiştir. Geliştirilen bu algoritma ile model eğitilirken yürütme süresinin (execution time) azaltılması ve modelin güncel kalması amaçlanmıştır. Ayrıca CBA algoritmasının geliştirilerek modelin güncel kalmasını sağlayan, artırımlı madenciliğe dayalı ilişkili sınıflandırma (associative classification based on incremental mining, ACIM) [20] ve artırımlı madencilik algoritmasına dayalı gelişmiş ilişkisel sınıflandırma (enhanced associative classification based on incremental mining algorithm, E-ACIM) [21] algoritmaları da artırımlı öğrenme yöntemine örnek olmaktadır. Artırımlı öğrenme yöntemi ile ilgili çalışmaların çoğu, görüntü tanıma ve sınıflandırma alanlarında yapılmaktadır [22, 23].

Makine öğrenmesinin uygulandığı alanlardan bir diğeri uç cihazlardır [2]. IoT cihazlarının geliştirilmesi için tasarlanmış Raspberry Pi üzerinde makine öğrenmesi algoritmalarının uygulanması [1], Iot cihazları makine öğrnemesi yöntemleri ile eğitme [3], Facebook uygulamasının akıllı telefonlarda ve diğer uç cihazlarda makine öğrenmesi yöntemleri ile veri işlemesi [24] bu alana yönelik yapılan çalışmalara örnek verilebilir.

Veri mahremiyeti ve kişisel verilerin güvenliği adına oluşan problemlere çözüm olarak, ilk defa Google 2016 yılında federe öğrenme kavramını ortaya koymuştur [25]. Veri gizliliği temelli oluşturulan bu öğrenme modeli sağlık, eğitim, akıllı şehir, giyilebilir cihazlar, finans, blok zincir ve nesnelerin interneti gibi önemli ve geniş bir alanda uygulamalara sahiptir [26, 27].

Bahsedilen çalışmalar uç cihazlar için federe öğrenme mimarisi ile makine öğrenmesi yöntemlerini uygulamanın büyük bir potansiyele sahip olduğunu göstermektedir. Bu çalışma kapsamında istemci-sunucu sistemleri için federe öğrenme mimarisi ile geliştirilen algoritma diğer araştırmacılara yol gösterecektir.

3. Problem Tanımı (Problem Definition)

Mobil uygulamalar, IoT cihazlar gibi istemci-sunucu sistemlerindeki uç cihazlarda makine öğrenmesi içeren çözümler kullanılmaktadır. Uçta makine öğrenmesi çözümü sunmak için sunucu tarafında (server-side) ve istemci tarafında (client-side) makine öğrenmesi modelleri hazırlanmaktadır. Sunucu tarafında bir modelin hazırlanması için uçlardan sunucuya verilerin aktarılması gerekmektedir. Veriler alındığında ise verilerden makine öğrenmesi modeli eğitilmektedir. Eğitilen modelin uçlarda kullanılması istek-yanıt iletişim örüntüsüyle veya modelin uçlara gönderilmesi kaydıyla yapılmaktadır. Bu çözümde hem veriler aktarılırken hem de sunucu-istemci iletişiminde ağın çevrimiçi olması gerekmektedir. İstemci tarafında da makine öğrenmesi modeli hazırlanmasına olanak sağlayan araçlar ve yaklaşımlar bulunmaktadır. Eğer istemci tarafında yani uçlarda bir model hazırlanmak isteniyorsa, uçtaki cihazın yerel kaynakları ve verileri ile eğitim sağlanmaktadır.

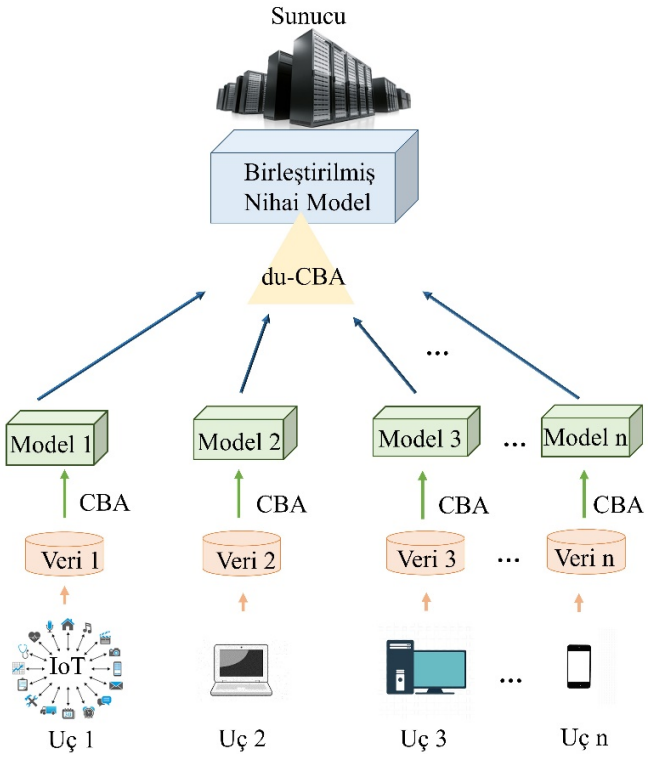
Uçta çalışan sistemlerde makine öğrenmesi çözümlerinin kullanılması için bazı zorlukların da aşılması gerekmektedir. Bu zorluklar şöyle sıralanabilir:

- Makine öğrenmesi modüllerinin çevrimdışı çalışmasına ihtiyaç duyulması
- Uçta çalışan sistemlerin ağ trafiğinin minimize edilmesi gerekliliği
- Veri önişlemenin yerelde yapılması gerekliliği ve bağlantılı olarak gecikmenin azaltılması ihtiyacı
- Veri mahremiyeti problemi

Uçta çalışan sistemler her zaman bir ağa bağlı olmayabilir. Örneğin; bir mobil uygulama düşünelim. Uygulamanın bir makine öğrenmesi modülüne bağlı olduğunu ve bu modülün sunucu tarafında çalıştığını varsayalım. İnternet erişiminin olmadığı bir zamanda uygulama ve makine öğrenmesi modülü arasında iletişim sağlanamayacak ve uygulama eksik kalacaktır. İnternet erişimi olması durumunda mobil uygulamadan sunucuya makine öğrenmesi modülüne girdi olması için veri gönderilecek ve sunucudan da yanıt alınacaktır. Bu durum mobil uygulamanın ağ trafiği oluşturmasına sebep olacaktır. Ağ trafiği de uygulama kullanıcısına maliyet oluşturacaktır. Bu iki problemin çözümü ancak uçta eğitilen ve yine uçta çalıştırılan makine öğrenmesi modellerinin kullanılması ile mümkün olmaktadır. Böylesi bir durumda ise yerelde var olan verinin makine öğrenmesi modeli eğitmek için yeterli olup olmaması hesaba katılmalıdır. Verinin yetersiz olması yetersiz öğrenmeye sebebiyet vermektedir. Bir diğer zorluk veri önişleme adımlarının sunucuda mı istemcide mi yapılacağıdır. Önişleme adımlarının sunucuda yapılması, sunucuya daha fazla veri göndermek yani fazla ağ trafiği oluşturmak anlamına gelmektedir. Buna bağlı makine öğrenmesi modülünün yavaş çalışması da söz konusu olmaktadır. Verilerin yerel sistemden sunucuya taşınması veri mahremiyeti hususunda zayıflığa sebep olmaktadır. Bu yayın kapsamında bahsedilen tüm problemlere çözüm sunan ve veri mahremiyetini suistimal etmeden makine öğrenmesi modellerini eğitip makine öğrenmesi modülüne sahip uygulamaların oluşturulmasını sağlayan federe öğrenme mimarisi kullanılmıştır. Ayrıca Veriden Habersiz İlişkili Kurallara Dayalı Sınıflandırma (Data Unaware Classification Based on Association, du-CBA) olarak adlandırılan algoritma geliştirilmiştir. Algoritma, federe öğrenme mimarisi için geliştirilmiş ilişkisel sınıflandırma algoritmasıdır.

4. Deneysel Metot (Experimental Method)

Bu bölümde; mobil uygulamalar, IoT cihazlar ve iş birliği içinde çalışan istemci-sunucu sistemler gibi uçta çalışabilecek cihazlar için federe öğrenme mimari metodolojisi ve bu kapsamda geliştirilen algoritma detaylarıyla anlatılmaktadır. Şekil 1.'de federe öğrenmenin işleyişi ile ilgili yapıya yer verilmiştir. Şekille göre sunulan mimaride uçta çalışan cihazlar bir mobil cihaz, bilgisayar ya da IoT cihazlar olarak temsil edilebilir. Bu cihazlar birbiri ile aynı ya da birbirinden farklı olabilir, farklı lokasyonlarda olabilir ancak cihazlarda aynı uygulama çalıştırılmakta ve eğitilen modeller her cihaz için aynı parametreler ve değişkenler ile olmaktadır. Uçlarda bulunan bu cihazlar üzerinde çalışan uygulama içerisinde, yerel veriler ile federe öğrenme mimarisini gerçekleştirmek için CBA algoritması ile modeller eğitilmektedir. Yerel veriler ile eğitilen modeller



Şekil 1. Federe öğrenme mimarisinin işleyişi (The working structure of the federated learning architecture)

kaydedilmekte ve ardından uygulamaların ulaşabildikleri ortak bir sunucuya gönderilmektedir. Her bir istemciden eş zamanlı sunucuya gelen modeller, sunucuda çalışan uygulama içerisinde Veriden Habersiz İlişkili Kurallara Dayalı Sınıflandırma (Data Unaware Classification Based on Association, du-CBA) olarak adlandırılan algoritma kullanılarak birleştirilmektedir. Modellerin birleştirilmesi ile sunucuda nihai model oluşturulmaktadır.

Bu çalışma kapsamında federe öğrenme mimarisine ait bir benzetim ortamı oluşturulmuştur. Bu doğrultuda geliştirilen du-CBA algoritması içerisinde, uçtaki cihazlarda model oluşturmak için CBA algoritması kullanılmaktadır. CBA, ilişkisel sınıflandırma algoritmasıdır. Denetimli veri madenciliği yöntemlerinden biri olan CBA, eğitim sonunda etiketli birliktelik kuralları oluşturur. Kurallar oluşturulurken destek ve güven parametreleri kullanılır [11]. Her bir uçta model eğitimi gerçekleştirildikten sonra modeller sunucuya gönderilmektedir. Eğitilmiş bu modellerin sunucuda birleştirilmesi için du-CBA içerisinde bir modül geliştirilmiştir. Modül ile tüm uçlarda oluşturulan modeller ve bu modellerin eğitiminde kullanılan veri içerisindeki örnek sayı (N) bilgisi alınmaktadır. Uçlarda eğitilen modeller CBA algoritması ile eğitildiği için etiketli kurallardan oluşmaktadır. Aslında modellerin birleştirilmesi modeller ile birlikte gelen kuralların birleştirilmesi anlamına gelmektedir. Modeller birleştirilirken gelen tüm kurallar kontrol edilmekte ardından birleştirme işlemi yapılmaktadır. Modellerin birleştirilmesi işlemi, kuralların destek ve güven değerlerinin güncellenmesi ve tekrar sıralanması ile gerçekleşmektedir. Farklı modellerden gelen aynı etikete sahip aynı kurallar için tüm veri içerisinde bulunma sıklığı değişeceğinden dolayı kurala ait destek ve güven değeri güncellenmektedir. Farklı etiketlenmiş aynı kurallar için ise önce bulunma sıklıkları kontrol edilmektedir. Sonra destek değeri büyük olan kural elde tutulmaktadır. Güncelleme işlemi yapıldıktan sonra yeni güven ve destek değerlerine sahip olan her kural önce güven değerine göre sıralanmaktadır. Güven değeri aynı ise destek değerine göre ikincil sıralama yapılmaktadır. Destek değerlerinin de aynı olması durumunda, önce gelen kural daha önde yer almaktadır. Elde edilen yeni kural listesi nihai modeli oluşturmaktadır. Destek ve güven değerlerinin güncellenmesi için formüller geliştirilmiştir.

Destek (Support): Bir ilişkinin veri seti içinde tekrarlanma oranıdır [28, 29]. Eş. 1

$$\text{Destek } (X \rightarrow Y) = \frac{\text{Frekans } (X,Y)}{N} \quad (1)$$

N: Toplam örnek sayısı

Güven (Confidence): X'in bulunduğu ilişkide Y'nin bulunma olasılığıdır [28, 29]. Eş. 2

$$\text{Güven } (X \rightarrow Y) = \frac{\text{Frekans } (X,Y)}{\text{Frekans } (X)} \quad (2)$$

İstemci sunucu sistemlerinde uçta çalışan cihaz sayısı bilinmediği için uç sayısı n olarak ifade edilmiştir. n tane uçta bulunan cihazlar için eğitilen her bir modeldeki kurallarının Destek değeri Eş. 1, Güven değeri ise Eş. 2'deki gibi hesaplanır. N model eğitiminde kullanılan veri seti içinde bulunan toplam örnek sayısıdır. X kuralın sol tarafını (lhs: Left hand side [30]), Y ise kuralın sağ tarafını (rhs: Right hand side [30]), yani kuralın etiketini göstermektedir. Her uçtan gelecek modellerin sunucuda birleştirilme işlemi kuralların destek ve güven değerlerinin güncellenmesi ve tekrar sıralanması ile gerçekleşmektedir. Model birleştirme işlemini gerçekleştirmek amacıyla verinin destek değerini güncelleyen matematiksel formül Eş. 5'te, güven değerlerini güncelleyen formül ise Eş. 9'da yer almaktadır. Fomüllerde bilinmeyen değerler yalnız bırakılarak, bilinen değerler ile yeni destek ve güven değerleri bulunmuştur.

$$D1 = \frac{F(X1,Y1)}{N1}, D2 = \frac{F(X2,Y2)}{N2} \ldots Dn = \frac{F(Xn,Yn)}{Nn} \quad (3)$$

D: Destek, F: Frekans

$$D_{birleştirme} = \frac{F(X1,Y1)+ F(X2,Y2)+\cdots+ F(Xn,Yn)}{N1+N2+\cdots+Nn} \quad (4)$$

$$D_{birleştirme} = \frac{(D1*N1)+(D2*N2)+\cdots+(Dn*Nn)}{N1+N2+\cdots+Nn} \quad (5)$$

Eş. 3'te, 1'den n'ye kadar uçta bulunan her cihazdan gelen kuralların destek formülleri yer almaktadır. Formüldeki N ve destek değeri bilinmektedir. Model birleştirilirken kuralların destek değerleride Eş. 4'te gösterildiği gibi birleştirilmektedir. Eş. 4'te 1'den n'ye tüm Frekans (X,Y) değerleri yerine, Eş. 3'e bakılarak elde edilen Destek*N değeri yazılmaktadır. Tüm bu işlemlerin sonucunda verinin destek değerini güncelleyen Eş. 5'teki formül elde edilmektedir.

$$G = \frac{F(X,Y)}{F(X)} => \frac{D*N}{G} = F(X) \quad (6)$$

G: Güven

$$G1 = \frac{F(X1,Y1)}{F(X1)}, G2 = \frac{F(X2,Y2)}{F(X2)} \dots Gn = \frac{F(Xn,Yn)}{F(Xn)} \quad (7)$$

$$G_{birleştirme} = \frac{F(X1,Y1)+ F(X2,Y2)+\cdots+ F(Xn,Yn)}{F(X1)+ F(X2)+\cdots+ F(Xn)} \quad (8)$$

$$G_{birleştirme} = \frac{(D1*N1)+(D2*N2)+\cdots+(Dn*Nn)}{\frac{D1*N1}{G1}+\frac{D2*N2}{G2}+\cdots+\frac{Dn*Nn}{Gn}} \quad (9)$$

Eş. 3'te bulunan destek formülündeki değerler, Eş. 6'da bulunan güven formülünde yerine yazarak Frekans (X), (F(X)) değeri elde edilmektedir. Eş. 7'de, 1'den n'ye kadar uçta bulunan her cihazdan gelen kuralların güven formülleri yer almaktadır. Model birleştirilirken kuralların güven değerleride Eş. 8'de gösterildiği gibi birleştirilmektedir. Eş. 9'da 1'den n'ye formülün pay kısmı olan tüm Frekans (X,Y) değerleri yerine, Eş. 3'e bakılarak elde edilen Destek*N değeri yazılmıştır. Payda kısmındaki Frekans (X) değerleri ise Eş. 6'ya bakılarak düzenlenmiştir. Tüm bu işlemlerin sonucunda verinin destek değerini güncelleyen Eş. 9'daki formül elde edilmektedir.

5. Deneysel Sonuçlar (Experimental Results)

Bu bölümde federe öğrenme mimarisi ile oluşturulan prototip uygulamanın deney süreçleri ve elde edilen sonuçlar paylaşılmaktadır. Federe öğrenme mimarisi kapsamında geliştirilen algoritma du-CBA, CBA algoritması üzerine inşa edilmiştir ve python dilinde uygulanmıştır. Deneylerde, CBA ile bu algoritmanın federe öğrenme mimarisine uygun bir şekilde geliştirilmiş hali olan du-CBA kıyaslanmıştır. Yani federe öğrenme mimarisine göre model eğitimi yapan du-CBA algoritması ile klasik öğrenme mimarisine göre model eğitimi yapan CBA algoritması karşılaştırılmıştır. Her iki algoritma ile eğitilen modellerin eğitim süreleri ve performansları kıyaslanmıştır. CBA ve du-CBA algoritmaları uygulanırken pyArc [31] modülü kullanılmış, ilişkilerin çıkarılması için CBA içerisinde varsayılan olarak belirnemiş Apriori [32] algoritması tercih edilmiştir. pyArc modülü için gerekli destek ve güven değerleri literatürde varsayılan değerler olan sırasıyla 0.2 ve 0.5 olarak belirlenmiştir [33-35]. Yapılan deneyler, 11.Nesil i5-1135G7 işlemciye ve 16 GB DDR3 belleğe sahip bir bilgisayarda gerçekleştirilmiştir.

CBA ve du-CBA algoritmalarının kıyası için UCI veri havuzundan car evaluation [36], bank marketing [37], mushroom [38], nursery [39] ve adult [40] veri setleri seçilmiştir. UCI veri havuzundan alınan bu beş veri seti ile CBA ve du-CBA algoritmaları kullanılarak ayrı ayrı modeller eğitilmiştir. Veri setinden daha anlamlı birliktelik kuralları elde edebilmek ve oluşacak modelin etkinliğini arttırmak için kategorik veriler gerekmektedir. Çalışma kapsamında kullanılan veri setleri kategorik verilerden oluştuğu ve veri seti içinde eksik (Null) veri bulunmadığı için tercih edilmiştir. Tablo 1'de bu veri setlerinin özellikleri gösterilmektedir.

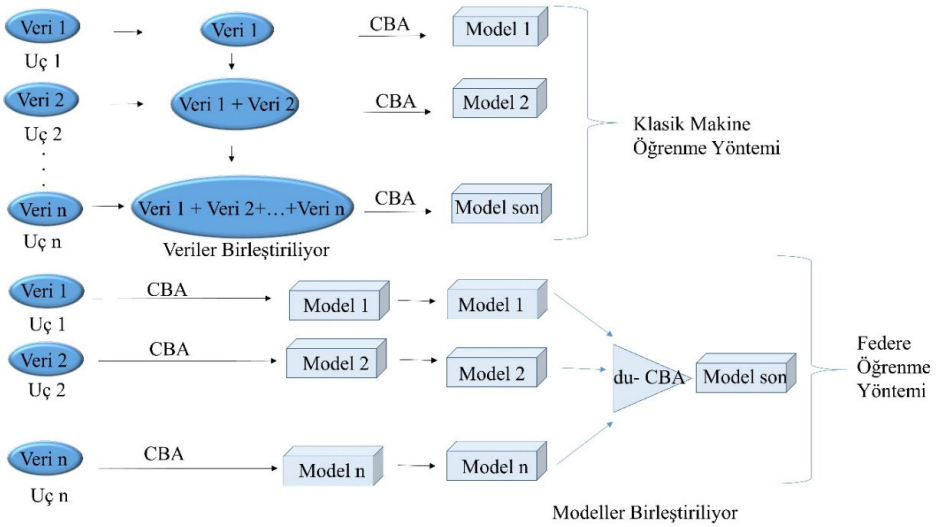
Tablo 1. UCI Veri Seti Özellikleri (UCI Dataset Properties)

Veri seti	Öznitelik Sayısı	Kayıt Sayısı	Sınıf
Car Evaluation	6	1728	4
Bank Marketing	16	4521	2
Mushroom	22	8124	2
Nursery	8	12960	4
Adult	14	32561	2

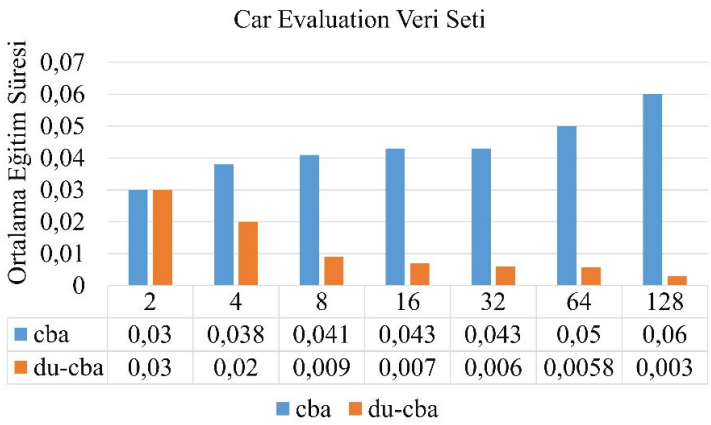
Şekil 2'de benzetim ortamında iki farklı mimari yöntem ile gerçekleştirilen model eğitimi gösterilmiştir. İlk model eğitiminde klasik öğrenme yöntemi kullanılmıştır, yani istemcilerden sunucuya verilerin aktarılmasıyla ve bu verilerden model eğitilmesiyle elde edilmiştir. Bu yöntemde sunucuya her uçtan veri gelmekte, gelen veriler sunucuda toplanmaktadır. Toplanan veriler ile eğitim yapılarak model oluşmaktadır. İkinci model ise federe öğrenme mimarisi olan, istemcilerde model eğitilip sunucuda modellerin birleştirilmesi ile nihai modelin eğitilmesidir. Klasik öğrenme ile model eğitimini gerçekleştirmek için CBA algoritması, federe öğrenme mimarisini benzetimde gerçekleştirmek için ise geliştirilen du-CBA algoritması kullanılmıştır. Bunun için öncelikle kullanılan veri setleri eğitim ve test setine ayrılmıştır. Eğitim setinde bulunan veriler sanki farklı uçlardan geliyormuş gibi varsayılmış ve uç sayısı kadar parçaya neredeyse eşit sayıda veri içerecek şekilde rastgele ayrılmıştır. Örneğin uç sayısı 2 kabul edildiğinde, eğitim seti neredeyse eşit iki parçaya rastgele ayrılmıştır. Buna göre; CBA algoritması test edilirken bu 2 parça birleştirilerek model eğitilmiş ve ardından test gerçekleştirilmiştir. Önerilen du-CBA algoritmasında ise bu 2 parçadan ayrı ayrı 2 model eğitilmiştir. Veriden bağımsız olarak uçlardan sunucuya sadece modellerin gönderildiği simüle edilmiştir. Modeller etiketli kurallardan oluşmaktadır. Her bir uçtan sunucuya tek başına anlamsız verilerden oluşan etiketli kurallar yani modeller gönderilmiş ve bu modeller birleştirilmiştir. Aslında modellerin birleştirilmesi modeller ile birlikte gelen kuralların birleştirilmesi anlamına gelmektedir. Sunucuda birleştirme işlemi ardından yeni bir model elde edilmiştir. Bu model nihai modeldir. Elde edilen nihai model ile testler gerçekleştirilmiştir. Şekil 2'de uçta çalışan cihaz sayısı bilinmediği için uç sayısı n olarak ifade edilmiştir. Farklı uç sayıları ile çalışmayı gözlemlemek için bahsedilen işlemler uç sayısı 4, 8, 16, 32, 64 ve 128 olarak varsayıldığı durumlar için de tekrarlanmıştır. Test sonuçları paylaşılmış ve kıyaslamalar gerçekleştirilmiştir.

Benzetim ortamında yapılan testler 5 kez tekrar edilerek çalıştırılmıştır. Geliştirilen algoritma içerisinde veri seti rastgele bölündüğü için her test sonucunda farklı değerler ortaya çıkmıştır. Testler sonucunda elde edilen değerlerin ortalaması alınmıştır. Her bir veri seti ve varsayılan her bir uç sayısı için doğruluk (accuracy), kesinlik (precision), duyarlılık (recall), F1 ölçütü değerlerinin ortalama sonucu ve bu değerlere göre hesaplanan standart sapma sonucu tablo üzerinde gösterilmiştir. Modellerin eğitim sürelerinin ortalama değerleri ise şekil üzerinde gösterilmiştir.

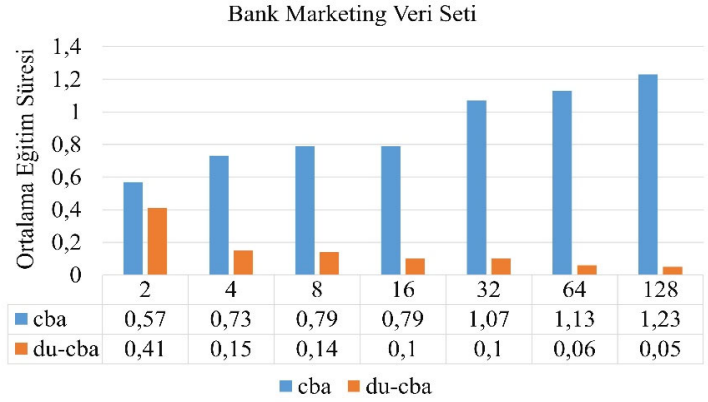
Şekil 3, Şekil 4, Şekil 5, Şekil 6 ve Şekil 7'de uç sayısı 2, 4, 8, 16, 32, 64 ve 128 olarak varsayılan yedi durum için, beş farklı veri seti ile CBA ve du-CBA algoritmalarının eğitim sürelerinin kıyaslamasını gösteren grafikler yer almaktadır. 7 farklı uç sayısı kullanılarak; car evaluation, bank marketing, mushroom, nursery ve adult veri setleri için ortalama eğitim süresi gösterilmektedir. Şekillerde uç sayısı yatay eksende, eğitim zamanı dikey eksende gösterilmektedir. Şekil üzerinde mavi çubuk klasik yöntem ile modelin eğitim süresini ifade



Şekil 2. Benzetim ortamında kıyaslanan klasik makine öğrnemesi yöntemi ve federe öğrenme yönteminin şekil ile gösterimi (Figure display of classical machine learning method and federated learning method compared in simulation environment)

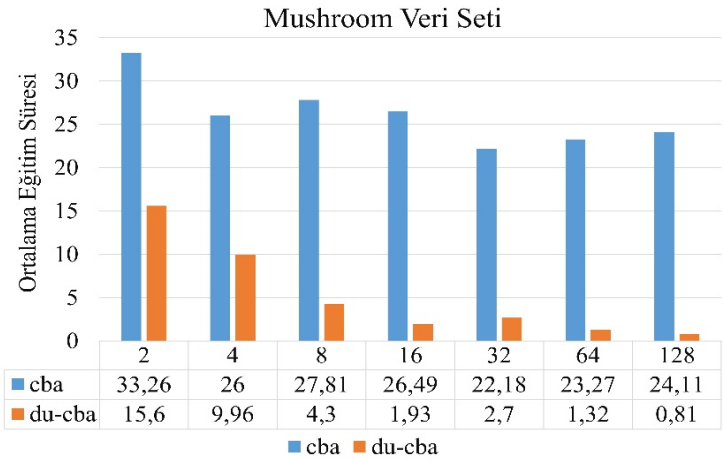


Şekil 3. Car Evalution veri seti için CBA ve du-CBA algoritmalarının 2, 4, 8, 16, 32, 64 ve 128 olarak belirlenen yedi farklı uç sayıları için tek tek modellerin eğitim sürelerinin karşılaştırılması. (Comparison of the training times of the CBA and du-CBA algorithms for the Car Evalution dataset for the seven different extreme numbers determined as 2, 4, 8, 16, 32, 64 and 128.)

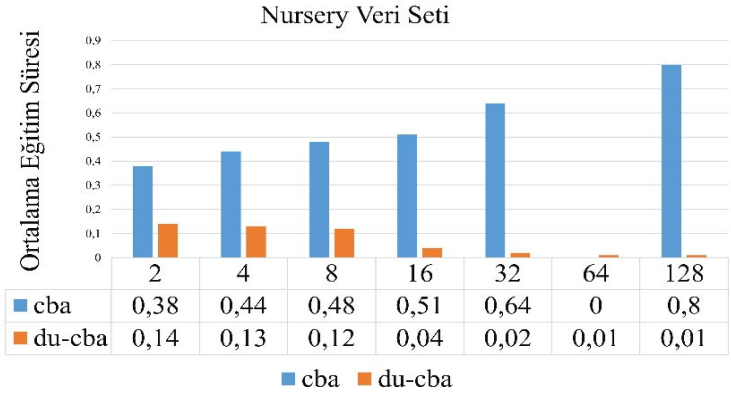


Şekil 4. Bank Marketing veri seti için CBA ve du-CBA algoritmalarının 2, 4, 8, 16, 32, 64 ve 128 olarak belirlenen yedi farklı uç sayıları için tek tek modellerin eğitim sürelerinin karşılaştırılması. (Comparison of the training times of the CBA and du-CBA algorithms for the Bank Marketting dataset for the seven different extreme numbers determined as 2, 4, 8, 16, 32, 64 and 128.)

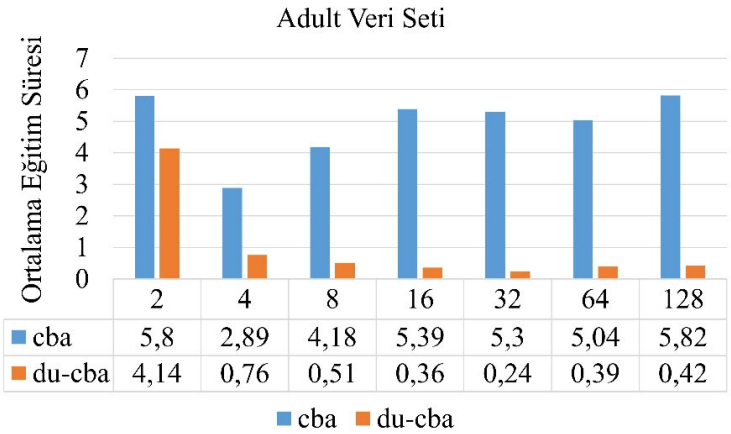
etmektedir. 2 uçtan oluşan bir sistemde, CBA ile model eğitimi için önce birinci uçtan gelen veri ile model eğitilmiştir. Daha sonra ikinci uçtan gelen veri, birinci uçtan gelen veri ile birleştirilip tekrar model eğitilmiştir. Her uçtan gelen verinin, mevcut veriler ile birleştirilerek



Şekil 5. Mushroom veri seti için CBA ve du-CBA algoritmalarının 2, 4, 8, 16, 32, 64 ve 128 olarak belirlenen yedi farklı uç sayıları için tek tek modellerin eğitim sürelerinin karşılaştırılması. (Comparison of the training times of the CBA and du-CBA algorithms for the Mushroom dataset for the seven different extreme numbers determined as 2, 4, 8, 16, 32, 64 and 128.)



Şekil 6. Nursery veri seti için CBA ve du-CBA algoritmalarının 2, 4, 8, 16, 32, 64 ve 128 olarak belirlenen yedi farklı uç sayıları için tek tek modellerin eğitim sürelerinin karşılaştırılması. (Comparison of the training times of the CBA and du-CBA algorithms for the Nursery dataset for the seven different extreme numbers determined as 2, 4, 8, 16, 32, 64 and 128.)



Şekil 7. Adult veri seti için CBA ve du-CBA algoritmalarının 2, 4, 8, 16, 32, 64 ve 128 olarak belirlenen yedi farklı uç sayıları için tek tek modellerin eğitim sürelerinin karşılaştırılması. (Comparison of the training times of the CBA and du-CBA algorithms for the Adult dataset for the seven different extreme numbers determined as 2, 4, 8, 16, 32, 64 and 128.)

model eğitilmesi işlemi son uç sayısına kadar devam etmektedir. Grafikte modelin eğitimi için harcanan zaman gösterilmektedir. Şekil üzerinde turuncu çubuk ise federe öğrenme yöntemi ile modelin eğitim süresini ifade etmektedir. du-CBA algoritması kullanılarak ilk

uçtan son uca kadar yerelde modeller eğitilmiş ve birleştirilmiştir. Grafik, modellerin eğitimi ve birleştirilme işlemi için harcanan zamanı göstermektedir. Grafik sonuçlarına göre tüm durumlar için federe öğrenme yönteminin klasik öğrenme yönteminden daha hızlı çalışma zamanına sahip olduğu sonucuna varılmıştır.

CBA ve du-CBA algoritmaları kullanılarak eğitilen bütün modellerin ortalama değerleri ve standart sapma değerleri tablolar halinde paylaşılmıştır. Bu değerler algoritmaların performanslarının değerlendirilmesinde ölçüt olarak kullanılmıştır.

Tablo 2'de CBA ve du-CBA algoritmalarının beş farklı veri seti üzerindeki başarı ölçütleri gösterilmiştir. Test edilen tüm durumlar için algoritmaların ortalama doğruluk (average accuracy), ortalama kesinlik (average precision), ortalama duyarlılık (average recall) ve ortalama F1 ölçütü değerleri karşılaştırılmış ve sonuçları paylaşılmıştır. Elde edilen sonuçlara göre iki yöntemin birbirlerine göre başarı ölçütleri bakımından belirgin bir üstünlüğü gözlenmemiştir.

Tablo 3'te CBA ve du-CBA algoritmaları ile eğitilen her bir modelin, eğitilmesinden elde edilen performans ölçütlerine ait değerlerin, standart sapmaları yer almaktadır. İstatistikte standart sapma değerinin küçük olması verilerin ortalama değere daha yakın şekilde dağıldığı anlamına gelmektedir. Tablo 3'e göre standart sapma değerlerinin düşük olması CBA ve du-CBA ile elde edilen her bir karşılaştırma ölçütü değerinin ortalama değere daha yakın şekilde dağılım gösterdiği anlamına gelmektedir. Bu istenilen bir durumdur.

Tablo 2. Beş farklı veri kümesi için CBA ve du-CBA algoritmaları ile eğitilen modellerin performans ölçülerinin ortalama değerleri
(Average values of performance measures of models trained with CBA and du-CBA algorithms for five different datasets)

		CBA				du-CBA			
Uç Sayısı	Veri Seti	Ortalama Doğruluk	Ortalama Kesinlik	Ortalama Duyarlılık	Ortalama F1	Ortalama Doğruluk	Ortalama Kesinlik	Ortalama Duyarlılık	Ortalama F1
2	Car Evaluation	0,8	0,78	0,8	0,78	0,8	0,76	0,81	0,79
	Bank Marketing	0,88	0,8	0,86	0,84	0,88	0,78	0,88	0,83
	Mushroom	0,98	0,99	0,99	0,99	0,98	0,99	0,99	0,99
	Nursery	0,66	0,52	0,67	0,56	0,66	0,52	0,67	0,56
	Adult	0,77	0,75	0,78	0,74	0,77	0,76	0,77	0,73
4	Car Evaluation	0,79	0,76	0,8	0,77	0,79	0,73	0,79	0,76
	Bank Marketing	0,88	0,78	0,88	0,83	0,88	0,78	0,88	0,83
	Mushroom	0,98	0,98	0,98	0,98	0,98	0,98	0,98	0,98
	Nursery	0,65	0,6	0,66	0,6	0,67	0,63	0,67	0,64
	Adult	0,77	0,76	0,78	0,74	0,77	0,79	0,78	0,71
8	Car Evaluation	0,79	0,78	0,8	0,78	0,79	0,72	0,79	0,74
	Bank Marketing	0,87	0,77	0,88	0,8	0,87	0,82	0,88	0,8
	Mushroom	0,99	0,99	0,99	0,99	0,99	0,99	0,99	0,99
	Nursery	0,66	0,6	0,67	0,6	0,7	0,7	0,71	0,7
	Adult	0,77	0,75	0,78	0,75	0,77	0,79	0,77	0,7
16	Car Evaluation	0,79	0,78	0,79	0,78	0,72	0,6	0,72	0,63
	Bank Marketing	0,87	0,77	0,87	0,82	0,87	0,77	0,87	0,82
	Mushroom	0,98	0,99	0,99	0,99	0,98	0,99	0,99	0,99
	Nursery	0,66	0,6	0,66	0,6	0,71	0,69	0,71	0,7
	Adult	0,78	0,76	0,78	0,75	0,77	0,79	0,78	0,71
32	Car Evaluation	0,79	0,77	0,8	0,78	0,71	0,6	0,71	0,61
	Bank Marketing	0,87	0,79	0,88	0,83	0,87	0,79	0,88	0,83
	Mushroom	0,98	0,99	0,99	0,99	0,95	0,95	0,95	0,95
	Nursery	0,66	0,6	0,66	0,6	0,71	0,71	0,72	0,72
	Adult	0,78	0,76	0,74	0,72	0,76	0,79	0,77	0,68
64	Car Evaluation	0,79	0,77	0,8	0,78	0,7	0,6	0,71	0,61
	Bank Marketing	0,88	0,77	0,88	0,83	0,88	0,77	0,88	0,83
	Mushroom	0,98	0,99	0,99	0,99	0,95	0,95	0,95	0,95
	Nursery	0,66	0,61	0,66	0,61	0,71	0,7	0,72	0,71
	Adult	0,78	0,76	0,78	0,75	0,76	0,79	0,77	0,67
128	Car Evaluation	0,8	0,77	0,81	0,79	0,73	0,69	0,73	0,68
	Bank Marketing	0,88	0,78	0,89	0,83	0,88	0,78	0,89	0,83
	Mushroom	0,98	0,99	0,99	0,99	0,91	0,91	0,91	0,91
	Nursery	0,65	0,6	0,65	0,6	0,75	0,74	0,75	0,75
	Adult	0,78	0,76	0,78	0,75	0,75	0,78	0,76	0,66

Tablo 3. CBA ve du-CBA algoritmalarının beş farklı veri seti ile eğitildiği modellerin performans ölçütlerine ait standart sapma değerleri
(Standard deviation values of performance criteria of CBA and du-CBA algorithms for five different data sets)

		CBA				du-CBA			
Uç Sayısı	Veri Seti	Doğruluk Standart Sapma	Kesinlik Standart Sapma	Duyarlılık Standart Sapma	F1 Standart Sapma	Doğruluk Standart Sapma	Kesinlik Standart Sapma	Duyarlılık Standart Sapma	F1 Standart Sapma
	Car Evaluation	0,015	0,025	0,018	0,023	0,016	0,035	0,029	0,028
	Bank Marketing	0,008	0,048	0,05	0,025	0,008	0,034	0,008	0,015
2	Mushroom	0,004	0	0	0	0,004	0	0	0
	Nursery	0,005	0,029	0,005	0,005	0,005	0,029	0,005	0,005
	Adult	0,005	0,005	0,007	0,008	0,005	0,02	0,004	0,01
	Car Evaluation	0,016	0,017	0,019	0,015	0,016	0,023	0,02	0,025
	Bank Marketing	0,007	0,017	0,01	0,01	0,007	0,017	0,01	0,01
4	Mushroom	0	0,004	0,004	0,004	0	0,004	0,004	0,004
	Nursery	0,008	0	0,007	0	0,02	0,04	0,02	0,05
	Adult	0,004	0,005	0,004	0,004	0,005	0,01	0,01	0,01
	Car Evaluation	0,024	0,033	0,03	0,03	0,013	0,023	0,011	0,011
	Bank Marketing	0,011	0,022	0,013	0,05	0,011	0,95	0,013	0,05
8	Mushroom	0,99	0,99	0,99	0,99	0,99	0,99	0,99	0,99
	Nursery	0,01	0	0,01	0	0,004	0,008	0,004	0,005
	Adult	0,004	0,004	0,005	0	0,005	0,01	0,004	0,01
	Car Evaluation	0,79	0,78	0,79	0,78	0,72	0,6	0,72	0,63
	Bank Marketing	0,08	0,01	0,008	0,01	0,08	0,01	0,008	0,01
16	Mushroom	0,98	0,99	0,99	0,99	0,98	0,99	0,99	0,99
	Nursery	0,01	0	0,01	0	0	0,005	0,004	0,005
	Adult	0,008	0,008	0,008	0,008	0,005	0,01	0,007	0,01
	Car Evaluation	0,79	0,77	0,8	0,78	0,71	0,6	0,71	0,61
	Bank Marketing	0,008	0,02	0,01	0,01	0,008	0,02	0,01	0,01
32	Mushroom	0,98	0,99	0,99	0,99	0,95	0,95	0,95	0,95
	Nursery	0	0	0,005	0	0,008	0,01	0,01	0,05
	Adult	0,008	0,008	0,07	0,05	0,008	0,01	0	0,008
	Car Evaluation	0,79	0,77	0,8	0,78	0,7	0,6	0,71	0,61
	Bank Marketing	0,01	0,02	0,01	0,02	0,01	0,02	0,01	0,02
64	Mushroom	0,98	0,99	0,99	0,99	0,95	0,95	0,95	0,95
	Nursery	0,01	0,02	0,01	0,02	0,01	0,01	0,01	0,01
	Adult	0,004	0,004	0,005	0,007	0,008	0,008	0,007	0,01
	Car Evaluation	0,8	0,77	0,81	0,79	0,73	0,69	0,73	0,68
	Bank Marketing	0,88	0,78	0,89	0,83	0,88	0,78	0,89	0,83
128	Mushroom	0,98	0,99	0,99	0,99	0,91	0,91	0,91	0,91
	Nursery	0,65	0,6	0,65	0,6	0,75	0,74	0,75	0,75
	Adult	0,004	0,007	0,005	0,005	0,004	0,03	0,005	0,008

6. Sonuçlar (Conclusions)

Bu çalışmasında, mobil uygulamalar, nesnelerin interneti, akıllı cihazlar gibi istemciler ve sunucuların birlikte çalıştığı sistemlerde, uçlardan veri çıkarmadan kollektif bir CBA modeli oluşturulması hedeflenmiştir. Modelin eğitiminde federe öğrenme mimarisi kullanılmıştır. Bu mimari ile uçlardan verinin değil sadece yerel verilerden eğitilmiş CBA modellerinin sunucuya gönderilmesi ile veri mahremiyeti problemi çözülmektedir. Bu özelliği ile sunucuda veri toplayarak model eğiten klasik öğrenmelerden ayrılmaktadır. Birçok uçtan gelen ve bütün veriyi temsil eden modellerin sunucuda birleştirilmesi ile yeni bir model oluşturulmakta ve bu model sunucudan uçlara tekrar gönderilmektedir. Bu sayede kollektif bir öğrenme de sağlanmaktadır.

Çalışmada ilişkisel sınıflandırma algoritmalarından biri olan CBA üzerinde çalışılmış ve federe öğrenme mimarisi için yeni bir algoritma oluşturulmuştur. du-CBA olarak adlandırılan bu algoritma ile eğitilen model öğrenmeye devam edip güncel kaldığı için artırımlı öğrenme yöntemine de örnektir. Federe öğrenme ile klasik öğrenme mimarilerini karşılaştırıp başarılarını ölçmek için çalışma kapsamında bir benzetim ortamı oluşturulmuştur. Benzetim ortamında iki farklı şekilde model eğitimi gerçekleştirilmiş ve elde edilen iki model kıyaslanmıştır. Yapılan testler, ayrı ayrı her bir veri seti için du-CBA ile model eğitiminde geçen süresinin yaklaşık olarak %70 oranında azaldığını, böylece eğitim süresinde büyük bir tasarrufun sağlandığını göstermektedir. Yine yapılan testler sonucunda elde edilen başarı ölçütlerine göre, bütün veriler ile eğitilen modelin, modellerin birleştirilmesine dayanan eğitime belirgin bir üstünlüğü gözlemlenmemektedir. Yapılan çalışmada, uçlardan bütün verinin sunuculara transferi gerekliliği ortadan kaldırılmıştır. Böylece uçlar ile sunucu arasında oluşan internet trafiği önemli ölçüde azaltılmaktadır. Sonuçlar geliştirilen algoritmanın başarıya ulaştığını ortaya koymaktadır. Bu bilgilere göre, literatüre mevcut temel yöntemin geliştirilmesi, performans olarak iyileştirilmesi ve daha önce uygulanmayan bir alana uygulanması kapsamında katlı sağlanmıştır.

Geliştirilen algoritmada birliktelik kuralları çıkartmak için Apriori algoritması kullanılmıştır. İlerleyen çalışmalarda diğer birliktelik kuralları algoritmaları kullanılarak duCBA'nın performansı değerlendirilecektir. Federe öğrenmesi mimarisi kapsamında veri güvenliğini sağlamak adına homomorfik şifreleme yöntemi kullanılarak çalışma yapılması hedeflenmektedir.

Ayrıca federe öğrenme yöntemi gerçek hayatta uygulandığında, kullanılan sistemlerin tam koordine edilememesinden kaynaklı bağlantı sorunları, modellerin güncelleme eksiklikleri, eğitim sürelerinin ve sürümlerinin farklı olması bu teknolojinin eksikliklerini ve zorluklarını göstermektedir. İlerleyen çalışmalarda federe öğrenmesi uygulanmasında meydana gelen bu sorunların çözümü için yeni algoritma ve yaklaşımların geliştirilmesi hedeflenmektedir.

Kaynaklar (References)

1. Yazici, M. T., Basurra, S., & Gaber, M. M., Edge machine learning: Enabling smart internet of things applications. Big data and cognitive computing, 2 (3), 26, 2018.
2. Merenda, M., Porcaro, C., & Iero, D., Edge machine learning for ai-enabled iot devices: A review. Sensors, 20 (9), 2533, 2020.
3. Murshed, M. S., Murphy, C., Hou, D., Khan, N., Ananthanarayanan, G., & Hussain, F., Machine learning at the network edge: A survey. ACM Computing Surveys (CSUR), 54 (8), 1-37, 2021.
4. Du, M., Wang, K., Chen, Y., Wang, X., & Sun, Y., Big data privacy preserving in multi-access edge computing for heterogeneous Internet of Things. IEEE Communications Magazine, 56 (8), 62-67, 2018.
5. Büyüknacar, Y., Canbay, Y., Federe öğrenme ve veri mahremiyeti, 2021.
6. Priya, S., & Selvakumar, S., PaSOFuAC: Particle Swarm Optimization Based Fuzzy Associative Classifier for Detecting Phishing Websites. Wireless Personal Communications, 1-30, 2022.
7. Çetin, E., & Ortalaş, F., Elektrikli ve Otonom Araçlarda Makine Öğrenmesi Kullanarak Trafik Levhaları Tanıma ve Simülasyon Uygulaması. El-Cezeri, 8 (3), 1081-1092, 2021.
8. Gözüaçık, N., A Virtual Assistant for Predicting Defective Software Module, 29th Signal Processing and Communications Applications Conference (SIU), IEEE, 1-4, June, 2021.
9. Gökdemir, A., Çalhan, A., Deep learning and machine learning based anomaly detection in internet of things environments, Journal of the Faculty of Engineering and Architecture of Gazi University, 37 (4), 1945-1956, 2022.
10. Güngör, E., Sinem, A. K., & Orman, Z., Makine Öğrenmesine Dayalı Mobil İngilizce Öğrenme Uygulaması, Bilgisayar Bilimleri ve Teknolojileri Dergisi, 1 (2), 58-65, 2021.
11. Liu, B., Hsu, W., & Ma, Y., Integrating classification and association rule mining, In Kdd, Vol. 98, pp. 80-86, August, 1998.
12. [12] Li, W., Han, J., & Pei, J., CMAR: Accurate and efficient classification based on multiple class-association rules, In Proceedings 2001 IEEE international conference on data mining, 369-376, November, 2001.
13. Thabtah, F., Cowling, P., & Peng, Y., MCAR: multi-class classification based on association rule, In The 3rd ACS/IEEE International Conference onComputer Systems and Applications, 33, January, 2005.
14. Alwidian, J., Hammo, B., & Obeid, N., Enhanced CBA algorithm based on apriori optimization and statistical ranking measure, In Proceeding of 28th International Business Information Management Association (IBIMA) conference on Vision, 2020, 4291-4306, 2016.
15. Alwidian, J., Hammo, B., & Obeid, N., FCBA: fast classification based on association rules algorithm, International Journal of Computer Science and Network Security (IJCSNS), 16 (12), 117, 2016.
16. Alwidian, J., Hammo, B. H., & Obeid, N., WCBA: Weighted classification based on association rules algorithm for breast cancer disease, Applied Soft Computing, 62, 536-549, 2018.
17. Gepperth, A., & Hammer, B., Incremental learning algorithms and applications, In European symposium on artificial neural networks (ESANN), 2016.
18. Hu, C., Chen, Y., Hu, L., & Peng, X., A novel random forests based class incremental learning method for activity recognition, Pattern Recognition, 78, 277-290, 2018.
19. Tanarat, S., & Kreesuradej, W., Incremental Classification Based on Association Rules Algorithm (ICBA), In Proceedings of the International Conference on Data Science (ICDATA), p. 1, The Steering Committee of the World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2011.
20. Alnababteh, M. H., Alfyoumi, M., Aljumah, A., & Ababneh, J., Associative Classification Based on Incremental Mining (ACIM), International Journal of Computer Theory and Engineering, 6 (2), 135, 2014.
21. Al-Fayoumi, M. A., Enhanced Associative classification based on incremental mining Algorithm (E-ACIM), International Journal of Computer Science Issues (IJCSI), 12 (1), 124, 2015.
22. Tang, C., Li, W., Wang, P., & Wang, L., Online human action recognition based on incremental learning of weighted covariance descriptors, Information Sciences, 467, 219-237, 2018.
23. Ristin, M., Guillaumin, M., Gall, J., & Van Gool, L., Incremental learning of random forests for large-scale image classification, IEEE transactions on pattern analysis and machine intelligence, 38 (3), 490-503, 2015.
24. Wu, C. J., Brooks, D., Chen, K., Chen, D., Choudhury, S., Dukhan, M., ... & Zhang, P., Machine learning at facebook: Understanding inference at the edge, In 2019 IEEE International Symposium on High Performance Computer Architecture (HPCA), 331-344, February, 2019.
25. Jiang, J.C., et al., Federated learning in smart city sensing: Challenges and opportunities.Sensors, 20 (21), 6230, 2020.
26. Liu, Y., et al., A systematic literature review on federated learning: From a model quality perspective. arXiv preprint arXiv:2012.01973, 2020.
27. Yang, Q., et al., Federated learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, 13 (3), 1-207, 2019.
28. Browne, P. R., Sweeting, A. J., & Robertson, S., Modelling the Influence of Task Constraints on Goal Kicking Performance in Australian Rules Football, Sports Medicine-Open, 8 (1), 1-12, 2022.
29. Agrawal, R., Imieliński, T., & Swami, A., Mining association rules between sets of items in large databases, In Proceedings of the 1993 ACM SIGMOD international conference on Management of data, 207-216, June, 1993.
30. Menzies, T., & Hu, Y., Data mining for very busy people, Computer, 36 (11), 22-29, 2003.
31. pyarc 1.1.4. https://pypi.org/project/pyarc/. Yayın tarihi Aralık 9, 2020. Erişim tarihi Aralık 26, 2021.
32. Agrawal, R., & Srikant, R., Fast algorithms for mining association rules, In Proc. 20th int. conf. very large data bases, VLDB, 1215, 487-499, September, 1994.
33. Abdelhamid, N., Multi-label rules for phishing classification, Applied Computing and Informatics, 11 (1), 29-46, 2015.
34. Moh'd Iqbal, A. L., Hadi, W. E., & Alwedyan, J., Detecting Phishing Websites Using Associative Classification, Journal of Information Engineering and Applications, 3, 2013.
35. Thabtah, F., Hadi, W., Abdelhamid, N., & Issa, A., Prediction phase in associative classification mining, International Journal of Software Engineering and Knowledge Engineering, 21 (06), 855-876, 2011.
36. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/car+evaluation. Erişim tarihi Aralık 26, 2021.
37. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/bank+marketing. Erişim tarihi Aralık 26, 2021.
38. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/mushroom. Erişim tarihi Aralık 26, 2021.
39. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/nursery. Erişim tarihi Aralık 26, 2021.
40. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/adult. Erişim tarihi Aralık 26, 2021.