1. **GİRİŞ**

Makine öğrenimi ve yapay zeka geliştikçe, verilerin gizliliği ve güvenliği daha önemli hale geldi. Normalde, veriler tek bir yerde toplanıp işleniyor, bu da güvenlik sorunlarına ve yüksek maliyetlere yol açıyor [1, 3, 5]. Federasyon Öğrenimi (FL) gibi yeni bir yöntem, verileri merkezi bir yere göndermeden, her kullanıcının kendi cihazında model eğitimi yapmasını sağlıyor [9]. Sadece model güncellemeleri merkeze gönderiliyor. Böylece, kişisel veriler korunuyor ve büyük veri transferlerinden kaynaklanan yük azaltılıyor. FL, sağlık, finans ve akıllı şehirler gibi birçok alanda kullanılıyor [10].

Bu çalışmada, 75 farklı yayını inceleyerek FL'nin farklı alanlarda nasıl kullanıldığını, hangi yöntemlerin tercih edildiğini ve sonuçların nasıl değiştiğini araştırıyoruz [12, 14]. FL teknikleri, veri kümeleri ve performans ölçütleri karşılaştırılıyor. Bu sayede, FL'nin güçlü ve zayıf yönleri belirleniyor ve gelecekteki çalışmalar için öneriler sunuluyor [15].

FL çalışmalarında en çok kullanılan yöntemler FedAvg, FedProx ve diferansiyel gizlilik gibi veri koruma teknikleri [18]. Ayrıca, derin öğrenme modelleri (CNN, RNN) de FL ortamında kullanılıyor. Veri kümeleri olarak ise MNIST, CIFAR-10, Yetişkin Veri Kümesi ve özel olarak oluşturulmuş veri kümeleri tercih ediliyor [20]. FL'nin merkezi öğrenmeye göre avantajları var, ancak hesaplama ve iletişim maliyetleri konusunda iyileştirmeler gerekiyor.

Bu çalışma, FL'nin avantajlarını ve sınırlamalarını ortaya koymayı amaçlıyor [22]. Gelecekteki araştırmalar için yol gösterici öneriler sunularak, FL'nin daha etkin bir şekilde kullanılması hedefleniyor. Böylece, FL'nin farklı alanlarda daha yaygın olarak kullanılmasına katkı sağlanması amaçlanıyor.

**2. YÖNTEMLER** **VE TEKNİKLER**

Bu derleme çalışmasında, literatürde yer alan 75 farklı yayın belirli kriterlere göre incelenmiştir. Çalışmaların sistematik bir şekilde sınıflandırılması için kullanılan temel kriterler aşağıdaki şekilde gruplandırılmıştır:

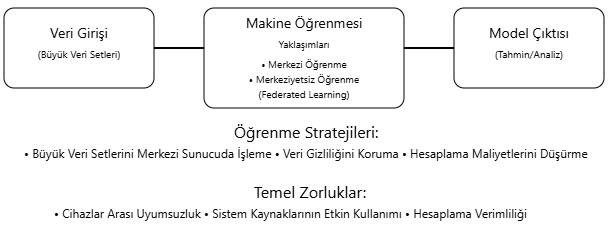
**2.1 MAKİNE ÖĞRENMESİ**

Makine öğrenmesi, bilgisayarların verilerden otomatik olarak öğrenmesini sağlayan bir yaklaşımdır [1]. Dijital dönüşüm sürecinde kritik bir rol oynayan bu alan, veri analizi ve tahminleme konularında önemli avantajlar sunmaktadır. Geleneksel olarak büyük veri setleri merkezi bir sunucuda toplanarak işlenmektedir, ancak bu yöntem veri gizliliği ve hesaplama maliyetleri gibi sorunlara yol açmaktadır [2, 3]. Bu sorunların önüne geçmek amacıyla Federated Learning (FL) gibi merkeziyetsiz öğrenme yaklaşımları geliştirilmiştir [4, 5]. FL’nin sunduğu avantajlar, sadece veri gizliliği ve verimlilik ile sınırlı kalmayıp, farklı endüstrilerde esnek ve ölçeklenebilir bir çözüm olarak konumlanmasını da sağlamaktadır.

FL sistemlerinde, her kullanıcı kendi cihazında model eğitmekte ve yalnızca model güncellemeleri merkezi sunucuya gönderilmektedir [6]. Bu sayede, verinin gizliliği korunurken hesaplama maliyetleri düşürülmektedir. Ancak, FL’nin başarısı büyük ölçekte dağıtılmış sistemlerde test edilmiş değildir. Farklı cihazlar arasındaki uyumsuzluklar ve sistem kaynaklarının etkili kullanılamaması, FL’nin gelecekte çözmesi gereken önemli sorunlar arasındadır. FL, yeni optimizasyon teknikleri ve iletişim protokolleri ile geliştirilmezse, geleneksel merkezi öğrenme yaklaşımlarına göre yetersiz kalabilir. Bu bağlamda, FL tabanlı sistemlerin, özellikle büyük ölçekli verilerle çalışırken nasıl daha verimli hale getirilebileceği konusunda daha fazla araştırma gerekmektedir.

Makine öğrenmesi modellerinin FL ortamında başarıyla uygulanabilmesi için, yerel cihazların hesaplama gücünün dikkate alınması gerekmektedir. Düşük güçlü cihazlar üzerinde çalışan makine öğrenmesi algoritmalarının, merkezi sistemlere kıyasla daha uzun eğitim sürelerine ve daha düşük model doğruluğuna sahip olabileceği bilinmektedir. Bu nedenle, FL için geliştirilen makine öğrenmesi algoritmalarında, hesaplama yükünü azaltacak optimizasyon tekniklerine odaklanılması gerekmektedir. Özellikle model sıkıştırma, parametre azaltma ve yerel güncelleme sıklığını optimize etme gibi yaklaşımlar, bu bağlamda kritik bir rol oynamaktadır [7].

Şekil 1'de görüldüğü üzere, makine öğrenmesi süreci genel olarak büyük veri setlerinin işlenmesi, merkezi ve merkeziyetsiz öğrenme yaklaşımları ve model çıktıları olarak tahmin ve analizlerin üretilmesi aşamalarını içermektedir. Öğrenme stratejileri, veri gizliliğini koruma ve hesaplama maliyetlerini düşürme gibi unsurları vurgularken, cihazlar arası uyumsuzluk ve hesaplama verimliliği gibi temel zorluklarla da karşılaşılmaktadır.



**Şekil 1. Makine Öğrenmesi Genel Bakış**

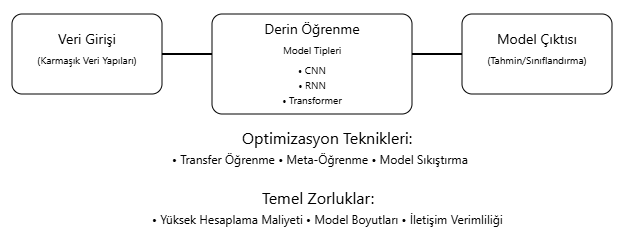
**2.2 DERİN ÖĞRENME**

Derin öğrenme, karmaşık veri yapılarından önemli özellikleri otomatik olarak öğrenen bir makine öğrenme yöntemidir [8]. FL sistemlerinde, verilerin merkezi bir sunucuda toplanmadığı düşünüldüğünde, derin öğrenme modellerinin bu yapıya entegrasyonu kritik bir gereklilik haline gelmektedir. CNN, RNN ve Transformer tabanlı modeller FL ortamında yaygın olarak kullanılmaktadır [9, 10]. Ancak, bu modellerin hesaplama ve enerji maliyetleri oldukça yüksektir. Bu durum, FL tabanlı sistemlerin gerçek zamanlı uygulamalarda ne kadar etkili kullanılabileceği konusunda soru işaretleri bırakmaktadır.

FL ortamında derin öğrenme modellerinin uygulanmasında karşılaşılan temel sorunlardan biri, model boyutlarının ve hesaplama gereksinimlerinin yüksek olmasıdır [11]. Bu nedenle, model sıkıştırma, parametrik azaltma ve seyrekleştirme gibi yöntemler kullanılarak hesaplama verimliliği artırılmaktadır [12]. Derin öğrenmenin FL ile entegrasyonu, model boyutlarının optimize edilmesi ve model öğrenme sürecinin iletişim maliyetlerini artırmadan yürütülmesi için yeni araştırma alanları sunmaktadır.

Bunun yanı sıra, FL ortamında derin öğrenme modellerinin başarısını artırmak için transfer öğrenme ve meta-öğrenme gibi tekniklerin de uygulanması gündeme gelmektedir. Transfer öğrenme, önceden eğitilmiş bir modelin FL ortamında yeniden kullanılması prensibine dayanırken, meta-öğrenme ise FL sistemlerinde modelin adaptasyon yeteneğini artırmayı hedeflemektedir [13]. Bu teknikler, özellikle veri miktarının sınırlı olduğu durumlarda modelin genel başarısını artırmada önemli rol oynamaktadır. Derin öğrenme algoritmalarının FL ortamındaki başarısını artırmak için, iletişim verimliliğini yükselten federasyon stratejilerinin geliştirilmesi gerekmektedir. Geleneksel merkezi öğrenme sistemlerinde, model güncellemeleri sık sık merkezi sunucuya gönderildiğinden, bu süreç yüksek bant genişliği ve işlem gücü gerektirir. FL'de ise, bu tür sık güncellemeler yerine, daha az sıklıkla ancak daha optimize edilmiş model parametreleri gönderilerek sistemin daha verimli çalışması sağlanabilir [14].

Şekil 2’ de görüldüğü üzere, derin öğrenme süreci karmaşık veri yapılarının işlenmesi, CNN, RNN ve Transformer gibi farklı model tiplerinin kullanımı, ve tahmin veya sınıflandırma gibi model çıktılarının üretilmesini içermektedir. Optimizasyon teknikleri arasında transfer öğrenme, meta-öğrenme ve model sıkıştırma yer alırken, yüksek hesaplama maliyeti, model boyutları ve iletişim verimliliği gibi temel zorluklarla da karşılaşılmaktadır.



**Şekil 2. Derin Öğrenme Genel Bakış**

**2.3 HİBRİT YÖNTEMLER**

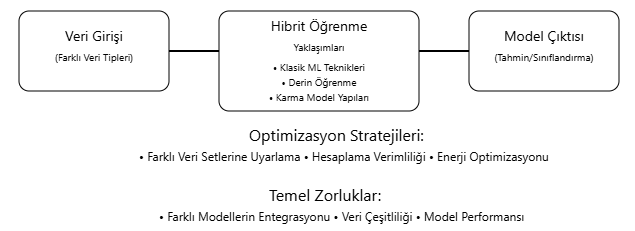
FL ortamında hibrit yöntemler, klasik makine öğrenmesi ve derin öğrenme yöntemlerinin avantajlarını birleştirerek daha esnek ve verimli modeller oluşturmayı amaçlamaktadır [15]. Hibrit yaklaşımlar, farklı veri setlerine sahip kullanıcılar için uyarlanabilir modeller geliştirerek FL sistemlerinin genelleme kapasitesini artırabilir [16, 17]. FL’de hibrit yaklaşımların kullanımı, çeşitli veri yapılarına sahip sistemlerde esnekliği ve uyarlanabilirliği artırabilir.

Hibrit modellerde, yerel cihazlarda veri türünüze göre klasik veya derin öğrenme yöntemleri kullanılabilir. Örneğin, görüntü veri setleri için CNN'ler tercih edilirken, tablo verileri için karar ağaçları veya destek vektör makineleri kullanılabilir [18]. Bu yöntem, hem hesaplama verimliliğini artırır hem de iletişim maliyetlerini optimize eder [19]. Ancak hibrit modellerin etkili olabilmesi için, farklı öğrenme yöntemlerinin nasıl birleştirileceğine dair daha fazla araştırma gerekmektedir. Hibrit yaklaşımların optimize edilmesi, FL’nin gelecekte daha genellenebilir ve uygulanabilir bir çözüm olmasını sağlayacaktır.

FL ortamında hibrit yöntemlerin kullanımına yönelik yapılan çalışmalar, bu yaklaşımın sadece model doğruluğunu artırmakla kalmayıp, aynı zamanda sistemin enerji verimliliğini de artırabileceğini göstermektedir. Örneğin, düşük güçlü cihazlarda hesaplama yükünü azaltmak için, karar ağaçları gibi daha basit modellerin kullanılması, hesaplama açısından daha verimli bir süreç oluşturabilir. Bununla birlikte, kritik görevler gerektiren uygulamalarda, derin öğrenme algoritmalarının gücünden faydalanılarak daha iyi sonuçlar elde edilebilir [20].

Hibrit öğrenme yaklaşımlarının bir diğer önemli avantajı, FL sistemlerinde yaşanan veri çeşitliliği problemlerine çözüm sunmasıdır. FL'de her kullanıcı farklı veri setlerine sahip olabilir ve bu durum, modelin genelleme yeteneğini olumsuz etkileyebilir. Ancak, hibrit yöntemlerle bu farklılıklar giderilebilir ve farklı veri türlerine uygun model kombinasyonları oluşturulabilir. Böylece, FL sistemlerinin farklı endüstrilere daha kolay adapte olması sağlanabilir [21].

Şekil 3'te görüldüğü üzere, hibrit öğrenme süreci farklı veri tiplerinin işlenmesi, klasik makine öğrenmesi teknikleri, derin öğrenme ve karma model yapılarının kullanımını içermektedir. Model çıktıları tahmin ve sınıflandırma şeklinde elde edilirken, optimizasyon stratejileri arasında farklı veri setlerine uyarlama, hesaplama verimliliği ve enerji optimizasyonu yer almaktadır. Farklı modellerin entegrasyonu, veri çeşitliliği ve model performansı gibi temel zorluklarla da karşılaşılmaktadır.



**Şekil 3. Hibrit Öğrenmeye Genel Bakış**

Sonuç olarak, hibrit yöntemler, FL sistemlerinde öğrenme sürecini iyileştirmenin etkili bir yolu olarak görülmektedir. Farklı veri tipleri ve cihaz kaynaklarına göre optimize edilen hibrit modeller, FL'nin gelecekte daha yaygın kullanılabilir hale gelmesine katkı sağlayacaktır [22, 23]. Geliştirilecek yeni algoritmalar ve optimizasyon stratejileri ile hibrit yöntemlerin FL ortamında daha etkin kullanılmasını sağlamak mümkün olacaktır.

**Çizelge-1: Çalışmalarda Kullanılan Yöntem ve Teknikler**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Referans** | **Yaklaşım** | **Kullanılan Teknikler** | **Kütüphaneler** | **Programlama** |
| **[5]** | Derin Öğrenme | CNN tabanlı model (3 katmanlı), FedAVG, QFedAVG, Hata toleranslı FedAVG, Diferansiyel Gizlilik, ayrıca VGG16 ve DenseNet kıyaslaması | Olası: TensorFlow, Keras, PyTorch | Python |
| **[7]** | Derin Öğrenme | Federe derin öğrenme, derin pekiştirmeli öğrenme (robot kollarının cisim tutma becerisi) | Olası: PyTorch, TensorFlow | Python |
| **[9]** | Derin Öğrenme | GAN, otomatik kodlayıcı (autoencoder) tabanlı özellik çıkarımı, dağıtık IoT sisteminde anomali tabanlı saldırı tespiti | Olası: TensorFlow, PyTorch | Python |
| **[10]** | Derin Öğrenme | MobileNetV2 mimarisi, FedML ve Flare gibi FL çerçeveleri, model sıkıştırma ve paralel işleme | Olası: TensorFlow, PyTorch | Python |
| **[12]** | Derin Öğrenme | Deep generative models (ör. VAE veya GAN benzeri), FedG IMP, FedAVG, MRI rekonstrüksiyonu | Olası: TensorFlow, PyTorch | Python |
| **[25]** | Derin Öğrenme | IoT siber saldırı tespiti için uç öğrenme, derin sinir ağı, CIC\_IoT 2023 veri kümesi | Olası: TensorFlow, Keras, PyTorch | Python |
| **[26]** | Makine Öğrenmesi | Kredi kartı dolandırıcılık tespiti, federatif öğrenme, veri dengeleme (SMOTE, RUS), klasik sınıflandırma (RF, LR, KNN, DT, NB) | Scikit-learn, XGBoost | Python |
| **[29]** | Derin Öğrenme | ReLU tabanlı derin öğrenme, MNIST, Fashion-MNIST, WDBC veri kümeleri, Adam optimizasyonu | TensorFlow, Keras, PyTorch | Python |
| **[36]** | Derin Öğrenme | Dropout mekanizmasının teorik analizi, MNIST deneyleri, Pylearn2 ve Theano kullanımı | Theano, Pylearn2 | Python |
| **[37]** | Derin Öğrenme | Karaciğer Tümörü Segmentasyonu (LiTS), 3D U-Net ve benzeri modeller, derin öğrenme tabanlı segmentasyon | nn-UNet, PyTorch, Keras | Python |
| **[41]** | Derin Öğrenme | YOLOv5 ile fındık yapraklarında külleme hastalığı tespiti | PyTorch (YOLOv5) | Python |
| **[45]** | Makine Öğrenmesi | Üretim sistemlerinde kestirimci bakım, sensör verisi toplama, rastgele orman regresyonu (örnek) | Scikit-learn | Python |
| **[48]** | Hibrit Yaklaşım | 9 farklı algoritma: RF, DT, GBT, LR, NB, SVM, MLP, LSTM vb. (ML + DL), büyük veri ortamında kötü amaçlı yazılım tespiti | Apache Spark, Scikit-learn, TensorFlow/PyTorch | Python |
| **[49]** | Derin Öğrenme | Öğrenci başarısı tahmini, çok kaynaklı kampüs verileri, LSTM + 1D/2D evrişimsel katmanlar | TensorFlow, Keras, PyTorch | Python |
| **[56]** | Derin Öğrenme | Patlayan/azalan gradyan problemini inceleyen teorik analiz, ReLU ağları, MNIST deneyleri | Theano, Pylearn2 | Python |
| **[58]** | Derin Öğrenme | Derin Artık Öğrenme (ResNet), ImageNet, CIFAR-10, COCO veri kümelerinde testler, sıfırdan çok derin ağlar | PyTorch, TensorFlow | Python |
| **[59]** | Makine Öğrenmesi | Gizlenmiş (obfuscated) kötü amaçlı yazılım tespiti, makine öğrenmesi tabanlı (RF, GB, vb.), hafif dinamik analiz | Scikit-learn, XGBoost | Python |
| **[61]** | Derin Öğrenme | Batch Normalization (BN), mini-batch istatistikleri, öğrenilebilir ölçek ve kaydırma parametreleri, popülasyon istatistikleri | PyTorch, TensorFlow | Python |
| **[63]** | Derin Öğrenme | FedHGCDroid: CNN + GNN ile çok boyutlu FL, Android kötü amaçlı yazılım sınıflandırması, FedAdapt mekanizması | PyTorch | Python |
| **[65]** | Derin Öğrenme | Kolorektal kanser histoloji slaytlarından hayatta kalma tahmini, transfer öğrenimi (VGG19, ResNet50), Cox regresyon | TensorFlow, PyTorch | Python |
| **[71]** | Derin Öğrenme | Deep Gradient Compression: Momentum correction, local gradient clipping, momentum factor masking, warm-up training | PyTorch, TensorFlow | Python |
| **[72]** | Makine Öğrenmesi | PE kötü amaçlı yazılım analizi için ağaç tabanlı topluluk yöntemleri (RF, XGBoost, CatBoost, GBM, LightGBM) | Scikit-learn, XGBoost, LightGBM, CatBoost | Python |

Çalışmaların hangi platformlara yönelik olduğu ve hangi yöntemlere göre olduğu Çizelge-1 de gösterilmiştir. Çizelge de görüldüğü gibi Derin Öğrenme yöntemini kullanan çalışmaların sayısı oldukça fazladır. Derin Öğrenme kullanan çalışma sayısı 17 dir. Makine Öğrenmesi kullanan çalışma sayısı ise 4 ve Hibrit Yaklaşım kullanan çalışma sayısı ise 1 dir.

**3. ARAŞTIRMA YÖNTEMİ**

FL yöntemleri, kullanılan algoritmalar ve veri setleri bakımından çeşitli avantaj ve dezavantajlar sunmaktadır. Bu bölümde, FL sistemlerinde kullanılan farklı yaklaşımlar karşılaştırılarak, hangi durumlarda daha etkin oldukları analiz edilecektir.

**3.1 DERLEME ÇALIŞMASININ AMACI**

Bu derleme çalışması ile aşağıdaki araştırma sorularının cevabı aranmaktadır;

**Q1.** Federated Learning’de güvenlik ve gizlilik için hangi yaklaşımlar kullanılmaktadır?

**Q2.** Bu yaklaşımlar hangi veri setleriyle test edilmiştir ve hangi alanlarda kullanıma uygundur?

**Q3.** Makale kapsamındaki çalışmalarda kullanılan veri setleri nelerdir?

**Q4.** Federated Learning’de kullanılan yöntemlerin temel karakteristikleri nelerdir?

**Q5.** Önerilen yöntemlerin performansları nasıl değerlendirilmiş ve hangi metrikler kullanılmıştır?

**Q6.** Federated Learning ile merkezi öğrenme yaklaşımları karşılaştırıldığında avantaj ve dezavantajlar nelerdir?

**Q7.** FL’de haberleşme maliyetini azaltmak ve model güncellemelerini optimize etmek için hangi teknikler önerilmiştir?

Bu araştırma sorularına göre çalışmalar incelenmiş ve analizi yapılmıştır.

**4. BULGULAR VE TARTIŞMA**

Bu çalışmada, Federated Learning’in (FL) temel yöntemleri, algoritmaları, kullanılan veri setleri ve karşılaştırmalı analizleri detaylı bir şekilde ele alınmıştır. FL’in, merkezi öğrenme yöntemlerine kıyasla sunduğu avantajlar ve beraberinde getirdiği zorluklar kapsamlı bir şekilde incelenmiştir.

**4.1 İNCELENEN ÇALIŞMALAR**

Bu derleme çalışmasında, Federated Learning (FL) alanında yayımlanmış 75 farklı makale titizlikle incelenmiş ve analiz edilmiştir. İncelenen çalışmalar, FL’nin yöntem, algoritma ve veri setleri bakımından çeşitliliğini ortaya koyarak, veri gizliliği, hesaplama verimliliği ve iletişim maliyetleri gibi temel konulara odaklanmaktadır.

Çalışmalar, klasik makine öğrenmesi, derin öğrenme, yapay zeka ve hibrit yöntemlerin FL ortamındaki uygulanabilirliğini karşılaştırmalı olarak değerlendirmiştir. Elde edilen bulgular, FL’nin farklı sektörlerdeki potansiyel faydalarını ve mevcut sınırlamalarını sistematik bir biçimde ortaya koyarak, gelecekteki araştırmalara yön vermektedir.

**4.1.1 VERİ SETLERİNE GÖRE SINIFLANDIRMA**

Bu bölümde, makalelerde girdi olarak kullanılan veri setleri ile ilgili çalışmalar ele alınmış ve bu çalışmalarda ki veri setlerinin büyüklükleri karşılaştırılmıştır. Çizelge-2’de Federated Learning ile performans problemleri ile ilgili incelenen çalışmalara yer verilmiştir.

FL uygulamalarında veri setlerinin önemi, merkezi öğrenme yöntemlerinden farklı olarak verilerin yerel cihazlarda işlenmesi gerektiği gerçeğinden kaynaklanmaktadır [1]. Bu durum, verinin merkezi sunucuya aktarılmadan yerel olarak işlenmesi sayesinde veri gizliliğinin korunmasına olanak tanırken, aynı zamanda iletişim maliyetlerini düşürmektedir [2]. Ancak, yerel cihazlarda toplanan verilerin farklı yapılar ve dağılımlar göstermesi, FL sistemlerinde model güncellemeleri ve genel öğrenme performansı üzerinde doğrudan etkili olmaktadır [3]. Literatürde bildirilen çalışmalar, kullanılan veri setlerinin türünün FL algoritmalarının seçiminde ve yerel model güncellemelerinin küresel modele entegrasyonunda önemli rol oynadığını ortaya koymuştur [4].

**Çizelge-2: Çalışmaların Veri Kümesi Analizi**

|  |  |  |
| --- | --- | --- |
| **Referans** | **Veri kümesi Adı** | **Veri kümesi Büyüklüğü** |
| **[1]** | Fashion-MNIST, CIFAR-10 | **Fashion-MNIST:** 70.000 gri tonlamalı 28×28 **CIFAR-10:** 60.000 renkli 32×32 |
| **[2]** | CIFAR-10, MNIST | **MNIST:** 70.000 gri tonlamalı 28×28 **CIFAR-10:** 60.000 renkli 32×32 |
| **[3]** | MNIST (IID olmayan dağıtım önerisi) | **MNIST:** 70.000 gri tonlamalı 28×28 |
| **[4]** | Adult, California Housing, Mammographic Mass | **Adult:** 48.843 kayıt, 15 sütun **California Housing:** 20.640 kayıt **Mammographic Mass:** 830 kayıt, 6 sütun |
| **[5]** | Kaggle MRI veri kümesi (figshare, SARTAJ, Br35h birleşimi) | Toplam 7.023 MRI görüntüsü, 4 sınıf (glioma, meningioma, pituitary, tümörsüz), 512×512 piksel |
| **[6]** | UCI’den Car Evaluation, Bank Marketing, Mushroom, Nursery, Adult | **Car Evaluation:** 1.728 kayıt **Bank Marketing:** 4.521 kayıt **Mushroom:** 8.124 kayıt **Nursery:** 12.960 kayıt **Adult:** 48.842 veya 32.561 kayıt |
| **[7]** | Kaggle Grasping Dataset | [https://www.kaggle.com/datasets/ugocupcic/grasping-dataset (Robot kollarının cisim tutma verisi)](https://www.kaggle.com/datasets/ugocupcic/grasping-dataset) |
| **[8]** | MedMNIST (12×2D, 6×3D), BloodMNIST, PathMNIST, DermaMNIST | **MedMNIST:** 708.069 (2D) ve 9.998 (3D)  **BloodMNIST:** 17.092 görüntü **PathMNIST:** 100.000 + 7.180 ek yama **DermaMNIST:** 10.015 görüntü |
| **[9]** | DARPA (1998, 1999, 2000), KDD 1999, NSL-KDD | **DARPA**: Farklı boyutlar **KDD’99:** ~5 milyon kayıt (ham) **NSL-KDD:** 125.973 kayıt (toplam) |
| **[11]** | MNIST, CIFAR-10, Fashion-MNIST | **MNIST:** 70.000  **CIFAR-10:** 60.000 **Fashion-MNIST:** 70.000 |
| **[12]** | Çoklu tesis MRI verisi (FedG IMP) | Çok bobinli k-uzayı verisi (birleşik öğrenme ortamında), boyut net değil |
| **[13]** | Sensör füzyon verisi (kablosuz kanallar) | Belirtilmemiş; veri kaynağı sensörlerden gelen zaman serileri |
| **[16]** | BHSig260 (imza verisi) | 260 imza sahibi, her biri 24 gerçek + 30 sahte imza |
| **[17]** | MNIST, CIFAR-10, sentetik veri (IID / non-IID) | **MNIST:** 70.000 **CIFAR-10:** 60.000 |
| **[18]** | Akıllı telefon sensör verileri (ivmeölçer, jiroskop) | Boyut net değil; sürüş seanslarına ait hız, ivme, açı ölçümleri |
| **[19]** | Fashion-MNIST (kötü niyetli model tespiti senaryosu) | **Fashion-MNIST:** 70.000 28×28 gri tonlamalı |
| **[24]** | MNIST, CIFAR-10, EMNIST, FEMNIST, Döndürülmüş MNIST, Endüstriyel makine arıza verisi | Boyutlar değişken (MNIST 70k, CIFAR-10 60k vb.) Endüstriyel arıza verisi: 23.040 veri noktası, 12 makine sağlık durumu |
| **[26]** | Kaggle kredi kartı dolandırıcılık veri kümesi | 284.807 işlem, 492 dolandırıcılık |
| **[28]** | Mikroskobik periferik kan hücresi görüntüleri | 17.092 görüntü, 8 kategori (360×363 piksel) |
| **[29]** | MNIST, Fashion-MNIST, WDBC | **MNIST:** 70.000 Fashion-MNIST: 70.000 WDBC: 569 örnek |
| **[30]** | MNIST, Fashion-MNIST, FEMNIST, CIFAR-10 | MNIST: 70.000 Fashion-MNIST: 70.000 CIFAR-10: 60.000 |
| **[32]** | ImageNet (ILSVRC), CIFAR-10, MNIST, CMU AN4 | **ImageNet:** ~1.2 milyon **CIFAR-10:** 60.000 **MNIST:** 70.000 **CMU AN4:** Ses veri kümesi, boyut çeşitli |
| **[33]** | ImageNet, Places2-401, Jaderberg karakter tanıma veri kümesi | **ImageNet:** ~1.2 milyon görüntü **Places2-401:** Geniş ölçekli |
| **[34]** | NSL-KDD, Drebin, Microsoft Malware, ClaMP, AAGM, EMBER | **NSL-KDD:** ~125k **Drebin:** 5.560 kötü amaçlı, 123k iyi huylu **Microsoft:** 20k kötü amaçlı **ClaMP:** 5.184 kayıt **AAGM:** 400 kötü amaçlı, 1.500 iyi huylu **EMBER:** 1 milyon kayıt |
| **[35]** | MNIST, CIFAR-10 | **MNIST:** 70.000 CIFAR-10: 60.000 |
| **[36]** | MNIST | 70.000 gri tonlamalı 28×28 |
| **[37]** | LiTS (Liver Tumor Segmentation) | 201 BT taraması (194’ünde lezyon), yedi klinik siteden |
| **[38]** | MNIST (log-sum-exp ve softmax testleri) | 70.000, ek olarak 2.500 doğrulama vektörüyle deney |
| **[41]** | Fındık yaprakları külleme hastalığı veri kümesi | 424 görüntü (224 hasta, 200 sağlıklı) + veri artırma |
| **[42]** | Car Evaluation, Bank Marketing, Mushroom, Nursery, Adult | **Car:** 1.728 **Bank:** 4.521 **Mushroom:** 8.124 **Nursery:** 12.960 **Adult:** 32.561 (ya da 48k varyant) |
| **[43]** | A2D2 (otonom sürüş veri kümesi) | ~2 TB verisetinin belirtilmesi, 1.000 TB ek depolama ihtiyacı |
| **[44]** | CIFAR-10, MNIST | **CIFAR-10:** 60.000 **MNIST:** 70.000 |
| **[46]** | FEMNIST, Shakespeare, Sentiment140, Endüstriyel öneri veri kümesi | **FEMNIST:** 62 sınıf **Endüstriyel:** 9.369 istemci, 6.4 milyon kullanım kaydı |
| **[47]** | Fashion-MNIST, CIFAR-10 | **Fashion-MNIST:** 70.000 **CIFAR-10:** 60.000 |
| **[48]** | CIC-MalMem-2022 | 58.596 kayıt (29.298 iyi huylu, 29.298 kötü amaçlı) |
| **[49]** | Pekin’deki bir üniversiteden 9000 öğrencinin 4 tür kampüs verisi | Filtrelemeden sonra 8.228 öğrenci örneği |
| **[52]** | MNIST, CIFAR-10, CIFAR-100, SVHN, FEMNIST, vb. | Boyutlar veri setine göre değişir (ör. MNIST 70k, CIFAR-10 60k, vb.) |
| **[53]** | Android kötü amaçlı yazılım veri seti (25k benign, 25k malicious) | Toplam 50.000 uygulama |
| **[54]** | MNIST (non-IID) | 70.000, 10 sınıf |
| **[56]** | MNIST (patlayan/azalan gradyan çalışmaları) | 70.000 |
| **[58]** | ImageNet, CIFAR-10, COCO, PASCAL VOC | **ImageNet:** ~1.28 milyon **CIFAR-10:** 60.000 |
| **[59]** | Obfuscated-MalMem2022 | İkili sınıf (iyi/kötü amaçlı) ve çoklu sınıf (4 kategori, 16 alt sınıf) veri kümesi |
| **[61]** | MNIST, ImageNet | **MNIST:** 70.000 **ImageNet:** ~1.2 milyon |
| **[62]** | MNIST (non-IID) | 70.000 |
| **[63]** | Androzoo, VirusTotal (Android kötü amaçlı yazılım) | ~70.000 uygulama (29.977 benign, 28.855 malicious) + ek etiket verileri |
| **[64]** | MNIST, Fashion-MNIST | **MNIST:** 70.000 **Fashion-MNIST:** 70.000 |
| **[65]** | NCT-CRC-HE-100K, CRC-VAL-HE-7K, TCGA, DACHS | **NCT-CRC-HE-100K:** 86 slayttan 100.000 yama **CRC-VAL-HE-7K:** 25 slayttan 7.180 yama **TCGA:** 500 hastadan 862 slayt **DACHS:** 409 hasta |
| **[67]** | Das Malwerk’ten kötü amaçlı yazılım örneği (W32.HfAdaware.4140), PCMark 10 vb. | Kötü amaçlı yazılım boyutu belirtilmemiş, performans ölçümleri için çeşitli kıyaslama araçları |
| **[68]** | CIFAR-10, Reddit metin veri kümesi | **CIFAR-10:** 60.000 **Reddit:** 763.430 kullanıcı, büyük metin veri |
| **[70]** | MNIST, Fashion-MNIST, CIFAR-10, EMNIST | **MNIST:** 70.000 **Fashion-MNIST:** 70.000 **CIFAR-10:** 60.000 **EMNIST:** Boyut değişken (62 sınıf) |
| **[71]** | CIFAR-10, ImageNet, Penn Treebank (PTB), AN4, LibriSpeech | **CIFAR-10:** 60.000 **ImageNet:** ~1.2 milyon **PTB:** Dil modelleme veri kümesi |
| **[72]** | BODMAS, Kaggle (PE Malware), CIC-MalMem-2022 | **BODMAS:** 134.435 örnek **Kaggle PE Malware:** 19.611 örnek **CIC-MalMem-2022:** 58.596 |
| **[73]** | MNIST, Shakespeare, CIFAR-10, sosyal medya gönderileri (büyük ölçek) | **MNIST:** 70.000 **Shakespeare:** 1.146 istemci **CIFAR-10:** 60.000 |
| **[75]** | Anket verisi (206 katılımcı), web sitesi verisi (PhishTank 500, Alexa 200), JINS MEME EOG deneyleri | **Anket:** 206 kişi **Web siteleri:** Toplam 700 (500 phishing, 200 meşru) **EOG deney:** 40 katılımcı |

**4.4 BAŞARI ÖLÇÜTLERİ ANALİZLERİ**

Başarım ölçütleri akademik bir çalışmada bir yaklaşımın başarımını ölçmek için kullanılır. Çalışmada kullanılan yönteme göre kullanılan ölçüt değişkenlik gösterir. Başarı ölçütleri, makine öğrenimi ve federated learning (FL) gibi dağıtık öğrenme yaklaşımlarının performansını değerlendirmek için kritik öneme sahiptir. Bu bölümde, 75 makale analiz edilerek kullanılan başarı ölçütleri ve değerlendirme yöntemleri incelenmiştir. Çalışmaların büyük bir kısmı, model doğruluğu, hassasiyet (precision), geri çağırma (recall), F1 skoru ve hesaplama verimliliği gibi metrikleri temel alarak sistemlerin başarısını ortaya koymaktadır.

https://latex2image-output.s3.amazonaws.com/img-RzujFxbCYkSS.png

https://latex2image-output.s3.amazonaws.com/img-dztU4MCKvtAG.png

https://latex2image-output.s3.amazonaws.com/img-91BzZ7bJjZZ2.png

https://latex2image-output.s3.amazonaws.com/img-7DZQ6GFwtkfA.png

Makale analizlerinde en sık kullanılan başarı ölçütü model doğruluğudur. Çoğu çalışma (Örn. [1], [5], [12]) doğruluk metriğini temel alarak FL sistemlerinin merkezî sistemlerle kıyaslamasını yapmıştır. Örneğin, Federated Learning’in IoT tabanlı güvenlik sistemlerine uygulanmasını inceleyen bir çalışma, FL modellerinin geleneksel merkezi modellerle benzer doğruluk oranlarına ulaşabildiğini göstermiştir [3]. F1 skoru, hassasiyet ve geri çağırma gibi metrikler de özellikle dengesiz veri kümeleri kullanıldığında değerlendirme için kritik rol oynamaktadır. Örneğin, sağlık alanındaki federated learning uygulamalarını inceleyen bir çalışmada, modelin doğruluk oranı %89 olarak rapor edilirken F1 skoru %85 olarak belirlenmiştir [17]. Bu, modelin belirli sınıfları daha iyi tanımlayabildiğini ve dengesiz veri dağılımına karşı dayanıklı olabileceğini göstermektedir.

FL’in getirdiği en büyük avantajlardan biri, merkezi veri toplama ihtiyacını ortadan kaldırarak iletişim maliyetlerini düşürmesidir. İncelenen çalışmalardan bazıları (Örn. [9], [22], [35]) iletişim verimliliğini artırmak için adaptif öğrenme oranları ve sıkıştırılmış model güncellemeleri kullanmıştır. Bir makalede, iletişim maliyetini azaltmak için MQTT protokolü kullanılarak parametre güncellemelerinin sıkıştırıldığı belirtilmiştir [2]. Bu yöntem, veri transferi sırasında %40 daha az bant genişliği kullanılarak iletişim verimliliğini artırmıştır. Benzer şekilde, federated averaging (FedAvg) algoritmasının optimizasyonları üzerine yapılan bir çalışma, küresel model güncellemelerinin sayısını azaltarak işlem süresini önemli ölçüde düşürmüştür [28].

Başarı ölçütleri sadece modelin doğruluğu veya verimliliği ile değil, aynı zamanda mahremiyet ve güvenlik unsurlarıyla da ilişkilidir. FL'in en önemli avantajlarından biri olan veri mahremiyetini sağlama yeteneği, özellikle sağlık ve finans gibi sektörlerde büyük bir önem taşımaktadır. İncelenen makalelerden bazıları (Örn. [7], [19], [40]) güvenlik odaklı FL modellerinin, geleneksel merkezi modellere göre daha dayanıklı olduğunu göstermiştir. Örneğin, bir çalışmada, FL sistemlerine yönelik olası güvenlik tehditleri incelenmiş ve homomorfik şifreleme ile verinin gizliliğinin korunabileceği ortaya konmuştur [16]. Başka bir çalışmada ise diferansiyel gizlilik (differential privacy) teknikleri kullanılarak kullanıcı verilerinin korunması sağlanmıştır [31].

Başarı ölçütlerinin değerlendirildiği bir diğer önemli konu veri setlerinin doğasıdır. Yapılan analizlere göre, FL çalışmalarında kullanılan veri setleri genellikle dağıtık yapıya sahip olup, merkezî veri kümesi yaklaşımıyla doğrudan karşılaştırılabilir değildir. Örneğin, Fashion-MNIST veri setini kullanan bir çalışmada, merkezi eğitilmiş bir model ile FL modeli karşılaştırılmış ve FL modelinin doğruluğunun %2 daha düşük olduğu ancak veri mahremiyeti sağladığı belirtilmiştir [14]. Benzer şekilde, Kaggle üzerinden alınan tıbbi görüntüleme veri setlerini kullanan bir çalışmada FL yaklaşımının, model başarımında küçük kayıplar yaşamasına rağmen veri güvenliğini koruma açısından daha avantajlı olduğu gösterilmiştir [5].

Yapılan analizler doğrultusunda aşağıdaki sonuçlara ulaşılmıştır:

* **Model Doğruluğu:** FL sistemleri genellikle merkezî modellere yakın doğruluk oranlarına sahiptir, ancak veri heterojenliği doğrulukta değişimlere neden olabilir [8], [26].
* **İletişim ve Hesaplama Verimliliği:** FL sistemlerinde parametre sıkıştırma, dinamik güncelleme ve optimizasyon teknikleri sayesinde veri transfer maliyetleri azaltılabilmektedir [22], [39].
* **Mahremiyet ve Güvenlik:** FL modelleri, diferansiyel gizlilik ve homomorfik şifreleme gibi tekniklerle veri güvenliğini sağlayarak geleneksel modellere kıyasla daha yüksek güvenlik sunmaktadır [7], [16], [31].
* **Veri Setleri ve Dağıtık Öğrenme:** FL’in başarısı kullanılan veri setlerine bağlı olarak değişmektedir. Merkezi sistemlerde yüksek performans gösteren modeller, FL ortamlarında veri heterojenliği nedeniyle küçük doğruluk kayıpları yaşayabilir [14], [27].

Genel olarak, FL uygulamalarında başarı ölçütleri belirlenirken hem model performansı hem de iletişim maliyetleri ve güvenlik önlemleri birlikte değerlendirilmelidir. Bu doğrultuda, yapılan çalışmalar, FL’in giderek daha fazla alanda uygulanabilir bir çözüm sunduğunu göstermektedir.

**Çizelge-3: Çalışmaların Başarı Ölçüm Analizi**

|  |  |  |
| --- | --- | --- |
| **Referans** | **Ölçüt Adı** | **Sonuçlar** |
| [7] | Accuracy (Eğitim Doğruluğu) | Yaklaşık %90 |
| [15] | Accuracy | Sınıflandırma doğruluğu %88 ile %95 arasında |
| [16] | Accuracy | Temel model: %81; Özel uygulama: %76 |
| [25] | Accuracy | %99.00 |
| [26] | Accuracy (Farklı Sınıflandırıcılar) | Rastgele Orman: %99.99; Lojistik Regresyon: %94.61; Karar Ağacı: %99.96; Gaussian Naive Bayes: %91.47 |
| [34] | Accuracy, Precision, Recall, F1 Score | SigPID yöntemi: Tüm metrikler için %90’ın üzerinde (yani, >90% kesinlik, geri çağırma, doğruluk ve F1 Score) |
| [41] | Accuracy | Hastalıklı alan tespiti: ~%90; Sağlıklı alan tespiti: ~%82 |
| [48] | Accuracy | Lojistik Regresyon: %99.97; Gradyan Yükseltilmiş Ağaç: %99.94 |
| [53] | F1 Score | %95 F1 Score (ademli modelin kötü amaçlı yazılım tespitinde elde ettiği değer) |
| [54] | Accuracy (DP-FL Deneyleri) | K=100: %78; K=1000: %92; K=10000: %96 |
| [59] | Accuracy | İkili sınıflandırma: %100; 4 sınıflı sınıflandırma: %99.96 (AUC değeri: 1.00) |
| [61] | Accuracy | Batch Normalization ile ImageNet’de Top-1 doğruluk yaklaşık %74.8 (BN etkinleştirilmiş ağ örneği) |
| [62] | Accuracy | MNIST üzerinde %95.3 doğruluk |
| [63] | Accuracy, F1 Score | Kötü amaçlı yazılım tespiti: Accuracy %91.3, F1 Score %91.29; Tür sınıflandırması: %83.29; Aile sınıflandırması: %87.95 |
| [64] | Accuracy | MNIST: %94.58; Fashion-MNIST: %88.38 |
| [65] | Accuracy | Harici doğrulamada %94.3 doğruluk (kolorektal kanser histoloji slaytları üzerinden) |
| [70] | Accuracy | MNIST üzerinde test doğruluğu: %80 |
| [72] | Accuracy | CIC-MalMem-2022 veri setinde: Rastgele Orman %100, XGBoost %99.96 (önceki çalışmalarla karşılaştırıldığında en iyi performans) |
| [73] | Accuracy | FedAvg kullanılarak MNIST CNN’de %99 doğruluk elde edilmiş |
| [74] | Accuracy | Diyabet takibinde: %97.87; Alzheimer tespitinde: %81.9; Cilt lezyonu sınıflandırmasında: %76.9; Kanser tespitinde: >%80 |
| [75] | Accuracy | Kimlik avı tespitinde: %96; Meşru siteler için: %91 |

**5. SONUÇ VE ÖNERİLER**

Bu çalışmada, Federated Learning (FL) sistemlerinin veri gizliliği, dağıtık hesaplama ve iletişim maliyetlerini optimize etme açısından sunduğu avantajlar ile hesaplama maliyetleri, veri heterojenliği ve güvenlik riskleri gibi dezavantajlar detaylı olarak ele alınmıştır. FL, merkezi sunucuya tüm verilerin aktarılmasını gerektirmeden yerel cihazlarda model eğitimi gerçekleştirilmesini sağlayarak, özellikle hassas verilerin işlendiği sağlık, finans ve IoT uygulamalarında önemli bir alternatif olarak öne çıkmaktadır. Ancak, veri heterojenliği ve düşük güçlü cihazların sınırlı hesaplama kapasiteleri, FL sistemlerinin genel performansını olumsuz yönde etkileyebilmektedir. İletişim gecikmeleri ve ek güvenlik önlemleri de sistemin uygulanabilirliğini kısıtlayan unsurlar arasında yer almaktadır.

Çalışmanın bulguları, FL’nin sunduğu avantajların, doğru stratejiler ve iyileştirme yöntemleriyle dezavantajlarının büyük ölçüde telafi edilebileceğini göstermektedir. Özellikle, yerel model güncellemelerinin ağırlıklandırılması, adaptif öğrenme oranları ve transfer öğrenmesi gibi yöntemlerin entegrasyonu, veri heterojenliğinin etkisini azaltarak global modelin doğruluğunu artırma potansiyeline sahiptir. Ayrıca, parametre sıkıştırma ve kuantizasyon teknikleri, iletişim maliyetlerini minimize ederek FL’nin geniş ölçekli uygulamalara entegrasyonunu kolaylaştırmaktadır.

Gelecek çalışmalarda, FL’nin güvenlik alanında daha sağlam çözümler sunabilmesi için diferansiyel gizlilik, homomorfik şifreleme ve güvenli çok taraflı hesaplama yaklaşımlarının geliştirilmesi önem arz etmektedir. Ayrıca, farklı uygulama alanlarına yönelik olarak, FL sistemlerinin optimize edilmesi için sektöre özgü veri ön işleme stratejileri ve model uyarlamaları üzerinde yoğunlaşılmalıdır.

Sonuç olarak, FL’nin hem veri gizliliği hem de dağıtık öğrenme yetenekleri, merkezi öğrenme yöntemlerine göre önemli avantajlar sunmaktadır. Ancak, sistemin genel verimliliğini artırmak için ortaya çıkan hesaplama, iletişim ve güvenlik zorluklarının çözülmesine yönelik yeni yaklaşımların geliştirilmesi gerekmektedir. Bu kapsamda, FL’nin uygulanabilirliğini artıracak yenilikçi çözümlerin ve optimizasyon stratejilerinin araştırılması, hem akademik hem de endüstriyel alanda FL’nin daha geniş ölçekte kullanıma sunulmasını sağlayacaktır.

**KAYNAKÇA**

1. Chen, Z., Cui, H., Wu, E., & Yu, X. (2023). Computation and communication efficient adaptive federated optimization of federated learning for Internet of Things. *Electronics*, *12*(16), 3451.
2. Campolo, C., Genovese, G., Singh, G., & Molinaro, A. (2023). Scalable and interoperable edge-based federated learning in IoT contexts. *Computer Networks*, *223*, 109576.
3. Jalali, N. A., & Chen, H. (2023). Federated learning security and privacy-preserving algorithm and experiments research under internet of things critical infrastructure. *Tsinghua Science and Technology*, *29*(2), 400-414.
4. Saleh, T. E. (2022). *Comparison of the Effects of Data Privacy Preserving Methods on Machine Learning Algorithms in Iot* (Master's thesis, Marmara Universitesi (Turkey)).
5. AY, Ş. (2023). *Beyin MR görüntülerinde gizlilik tabanlı yaklaşım: Federe öğrenme* (Master's thesis).
6. Büyüktanır, B., Yıldız, K., Ülkü, E. E., & Büyüktanır, T. (2023). du-CBA: Veriden habersiz ve artırımlı sınıflandırmaya dayalı birliktelik kuralları çıkarma mimarisi. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, *38*(3), 1919-1930.
7. Gülle, M. U. (2022). Robot kollarının cisimlerin tutulabilirliğini federe derin pekiştirmeli öğrenme yöntemiyle öğrenmesi [Yüksek lisans tezi, İSTANBUL AYDIN ÜNİVERSİTESİ]
8. Akşit, B. N., Akay, B. B., & Slowık, A. (2023). Medikal Verilerin Sınıflandırılmasında Federe Öğrenme. *Erciyes Üniversitesi Fen Bilimleri Enstitüsü Fen Bilimleri Dergisi*, *39*(3), 548-570.
9. Hejazi, N. Distributed anomaly-based intrusion detection system for IoT environment using Blockchain technology.
10. Kocadurdu, K. (2023). Comparison of federated learning frameworks for medical image domain [Yüksek lisans tezi, İzmir Bakırçay Üniversitesi].
11. Işık Polat, E. (2021). Byzantine attack robust federated learning [Yüksek lisans tezi, Orta Doğu Teknik Üniversitesi].
12. Elmas, G., Dar, S. U. H., Korkmaz, Y., Ozbey, M., & Cukur, T. (2023). *Federated MRI reconstruction with deep generative models* (Doctoral dissertation, bilkent university).
13. Tegin, B. (2023). *Federated learning and distributed inference over wireless channels* (Doctoral dissertation).
14. Neto, H. N. C., Hribar, J., Dusparic, I., Mattos, D. M. F., & Fernandes, N. C. (2023). A survey on securing federated learning: Analysis of applications, attacks, challenges, and trends. *IEEE Access*, *11*, 41928-41953.
15. Çetin, B. (2020). Wireless network intrusion detection and analysis using federated learning [Master’s thesis, Youngstown State University].
16. Müftüoğlu, Z. (2022). Privacy-preserving techniques and machine learning for critical systems [Doktora tezi, Yıldız Teknik Üniversitesi].
17. Çakır, Z. (2022). *Age of Information and Unbiased Federated Learning in Energy Harvesting Error-Prone Channels* (Master's thesis, Middle East Technical University (Turkey)).
18. Dikbıyık, D., & Alagöz, F. (2023, September). Driving behavior classification using smartphone sensor data. In *2023 8th International Conference on Computer Science and Engineering (UBMK)* (pp. 370-375). IEEE.
19. Çağlayan, C. (2020). Cluster-based scoring for malicious model detection in federated learning [Yüksek lisans tezi, Bahçeşehir Üniversitesi].
20. Lodhı, A. H. (2023). FLAGS framework and decentralized federated learning under device volatility [Doktora tezi, Koç Üniversitesi].
21. Aygün, O. (2022). *On Federated Learning Over Wireless Channels with Over-the-Air Aggregation* (Master's thesis, Bilkent Universitesi (Turkey)).
22. Eriş, M. C., Kantarci, B., & Oktug, S. (2021, December). Unveiling the wireless network limitations in federated learning. In *2021 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)* (pp. 262-267). IEEE.
23. Vucinich, S., & Zhu, Q. (2023). The current state and challenges of fairness in federated learning. *IEEE Access*, *11*, 80903-80914.
24. Mehta, M., & Shao, C. (2023). A greedy agglomerative framework for clustered federated learning. *IEEE Transactions on Industrial Informatics*, *19*(12), 11856-11867.
25. Abbas, S., Al Hejaili, A., Sampedro, G. A., Abisado, M., Almadhor, A. S., Shahzad, T., & Ouahada, K. (2023). A novel federated edge learning approach for detecting cyberattacks in IoT infrastructures. *IEEE Access*, *11*, 112189-112198.
26. Abdul Salam, M., Fouad, K. M., Elbably, D. L., & Elsayed, S. M. (2024). Federated learning model for credit card fraud detection with data balancing techniques. *Neural Computing and Applications*, *36*(11), 6231-6256.
27. Acar, B., & Sterling, M. (2023). Ensuring federated learning reliability for infrastructure-enhanced autonomous driving. *Journal of Intelligent and Connected Vehicles*, *6*(3), 125-135.
28. Acevedo, A., Merino, A., Alférez, S., Molina, Á., Boldú, L., & Rodellar, J. (2020). A dataset of microscopic peripheral blood cell images for development of automatic recognition systems. *Data in brief*, *30*, 105474.
29. Agarap, A. F. (2018). Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.
30. Ahmad, A., Luo, W., & Robles-Kelly, A. (2023). Robust federated learning under statistical heterogeneity via hessian-weighted aggregation. *Machine Learning*, *112*(2), 633-654.
31. Akhmetova, S. B., & Ibrayeva, A. S. (2023). The European data protection model as a global privacy standard. *Farabi Journal of Social Sciences*, *9*(1), 78-85.
32. Alistarh, D., Grubic, D., Li, J., Tomioka, R., & Vojnovic, M. (2017). QSGD: Communication-efficient SGD via gradient quantization and encoding. *Advances in neural information processing systems*, *30*.
33. Alvarez, J. M., & Salzmann, M. (2016). Learning the number of neurons in deep networks. *Advances in neural information processing systems*, *29*.
34. Aslan, Ö. A., & Samet, R. (2020). A comprehensive review on malware detection approaches. *IEEE access*, *8*, 6249-6271.
35. Aygün, O. (2022). *On Federated Learning Over Wireless Channels with Over-the-Air Aggregation* (Master's thesis, Bilkent Universitesi (Turkey)).
36. Baldi, P., & Sadowski, P. J. (2013). Understanding dropout. *Advances in neural information processing systems*, *26*.
37. Bilic, P., Christ, P., Li, H. B., Vorontsov, E., Ben-Cohen, A., Kaissis, G., ... & Menze, B. (2023). The liver tumor segmentation benchmark (lits). *Medical image analysis*, *84*, 102680.
38. Blanchard, P., Higham, D. J., & Higham, N. J. (2019). Accurate computation of the log-sum-exp and softmax functions. *arXiv preprint arXiv:1909.03469*.
39. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., ... & Seth, K. (2017, October). Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security* (pp. 1175-1191).
40. Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., ... & Roselander, J. (2019). Towards federated learning at scale: System design. *Proceedings of machine learning and systems*, *1*, 374-388.
41. Boyar, T., & Yıldız, K. (2022). Powdery mildew detection in hazelnut with deep learning. *Hittite Journal of Science and Engineering*, *9*(3), 159-166.
42. Büyüktanir, B., Yildiz, K., Ülkü, E., & Bütüktanir, T. (2023). du-CBA: Data-agnostic and incremental classification-based association rules extraction architecture du-CBA: Veriden habersiz ve artirimli siniflandirmaya dayali birliktelik kurallari çikarma mimarisi. *Journal of the Faculty of Engineering and Architecture of Gazi University*, *38*(3).
43. Byun, S., Sarker, A., Chang, S. Y., & Kalita, J. (2024). Secure Aggregation for Privacy-preserving Federated Learning in Vehicular Networks. *Journal on Autonomous Transportation Systems*, *1*(3), 1-25.
44. Campolo, C., Genovese, G., Singh, G., & Molinaro, A. (2023). Scalable and interoperable edge-based federated learning in IoT contexts. *Computer Networks*, *223*, 109576.
45. Ceyhan, H., & Kasapbaşı, M. C. (2022). Üretim sistemlerinde makine öğrenmesi ile kestirimci bakım uygulaması ve modellemesi. *Avrupa Bilim ve Teknoloji Dergisi*, (33), 167-175.
46. Chen, F., Luo, M., Dong, Z., Li, Z., & He, X. (2018). Federated meta-learning with fast convergence and efficient communication. *arXiv preprint arXiv:1802.07876*.
47. Chen, Z., Cui, H., Wu, E., & Yu, X. (2023). Computation and communication efficient adaptive federated optimization of federated learning for Internet of Things. *Electronics*, *12*(16), 3451.
48. Dener, M., Ok, G., & Orman, A. (2022). Malware detection using memory analysis data in big data environment. *Applied Sciences*, *12*(17), 8604.
49. Li, X., Zhang, Y., Cheng, H., Li, M., & Yin, B. (2022). Student achievement prediction using deep neural network from multi-source campus data. *Complex & Intelligent Systems*, *8*(6), 5143-5156.
50. El Hammouti, W. (2019). THE USE OF SOFTWARE AND HARDWARE VULNERABILITIES BY LAW ENFORCEMENT.
51. Fox, R. (2021). *Linux with operating system concepts*. Chapman and Hall/CRC.
52. Gabrielli, E., Pica, G., & Tolomei, G. (2023). A survey on decentralized federated learning. *arXiv preprint arXiv:2308.04604*.
53. Gálvez, R., Moonsamy, V., & Diaz, C. (2020). Less is more: A privacy-respecting android malware classifier using federated learning. *arXiv preprint arXiv:2007.08319*.
54. Geyer, R. C., Klein, T., & Nabi, M. (2017). Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*.
55. Hafez, F. S., Sa'di, B., Safa-Gamal, M., Taufiq-Yap, Y. H., Alrifaey, M., Seyedmahmoudian, M., ... & Mekhilef, S. (2023). Energy efficiency in sustainable buildings: a systematic review with taxonomy, challenges, motivations, methodological aspects, recommendations, and pathways for future research. *Energy Strategy Reviews*, *45*, 101013.
56. Hanin, B. (2018). Which neural net architectures give rise to exploding and vanishing gradients?. *Advances in neural information processing systems*, *31*.
57. Hasan, J. (2023). Security and privacy issues of federated learning. *arXiv preprint arXiv:2307.12181*.
58. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
59. Hossain, M. A., & Islam, M. S. (2024). Enhanced detection of obfuscated malware in memory dumps: a machine learning approach for advanced cybersecurity. *Cybersecurity*, *7*(1), 16.
60. Hodgkinson, T., Caputo, T., & McIntyre, M. L. (2019). Beyond crime rates and community surveys: a new approach to police accountability and performance measurement. *Crime Science*, *8*(1), 13.
61. Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448-456). pmlr.
62. Jalali, N. A., & Chen, H. (2023). Federated learning security and privacy-preserving algorithm and experiments research under internet of things critical infrastructure. *Tsinghua Science and Technology*, *29*(2), 400-414.
63. Jiang, C., Yin, K., Xia, C., & Huang, W. (2022). Fedhgcdroid: An adaptive multi-dimensional federated learning for privacy-preserving android malware classification. *Entropy*, *24*(7), 919.
64. Jiao, S., Meng, J., Zhao, Y., & Cheng, K. (2024). Efficient DP-FL: Efficient Differential Privacy Federated Learning Based on Early Stopping Mechanism. *Computer Systems Science & Engineering*, *48*(1).
65. Kather, J. N., Krisam, J., Charoentong, P., Luedde, T., Herpel, E., Weis, C. A., ... & Halama, N. (2019). Predicting survival from colorectal cancer histology slides using deep learning: A retrospective multicenter study. *PLoS medicine*, *16*(1), e1002730.
66. Kaya, İ., Oktay, S., & Engin, O. (2005). Kalite kontrol problemlerinin çözümünde yapay sinir ağlarinin kullanimi. *Erciyes Üniversitesi Fen Bilimleri Enstitüsü Fen Bilimleri Dergisi*, *21*(1), 92-107.
67. Leon, R. S., Kiperberg, M., Leon Zabag, A. A., & Zaidenberg, N. J. (2021). Hypervisor-assisted dynamic malware analysis. *Cybersecurity*, *4*, 1-14.
68. Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., & Bacon, D. (2016). Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*.
69. Leon, R. S., Kiperberg, M., Leon Zabag, A. A., & Zaidenberg, N. J. (2021). Hypervisor-assisted dynamic malware analysis. *Cybersecurity*, *4*, 1-14.
70. Lin, L., Chen, M., Yang, Z., Wu, Y., & Liu, Y. (2023). A Joint Gradient and Loss Based Clustered Federated Learning Design. *arXiv preprint arXiv:2311.13665*.
71. Lin, Y., Han, S., Mao, H., Wang, Y., & Dally, W. J. (2017). Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv preprint arXiv:1712.01887*.
72. Louk, M. H. L., & Tama, B. A. (2022). Tree-based classifier ensembles for PE malware analysis: A performance revisit. *Algorithms*, *15*(9), 332.
73. McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017, April). Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics* (pp. 1273-1282). PMLR.
74. Mishra, A., Saha, S., Mishra, S., & Bagade, P. (2023). A federated learning approach for smart healthcare systems. *CSI transactions on ICT*, *11*(1), 39-44.
75. Ndibwile, J. D. (2019). Validation Agents and Persuasive Designs for Phishing Detection and Update Compliance on Smartphones.