Gerçek Zamanlı Büyük Veri Analitiği ile Anomali Tespiti

Spark ve Kafka Entegrasyonu Projesi

1. Veri Seti ve Ön İşleme

1.1 Veri Seti Hakkında

Bu projede RT_IOT2022.csv veri seti kullanılmıştır. Bu veri seti, IoT ağ trafiği verilerini içermektedir ve aşağıdaki özelliklere sahiptir:

- Toplam kayıt sayısı: [veri setinizdeki satır sayısı]

- Özellik sayısı: [veri setinizdeki sütun sayısı]

- Sınıf dağılımı: Normal ve anormal trafik örnekleri

Veri setinde bulunan önemli özellikler:

- flow_duration: Akış süresi

- fwd_pkts_tot: İleri yönlü paket toplamı

- bwd_pkts_tot: Geri yönlü paket toplamı

- Attack_type: Saldırı türü (hedef değişken)

1.2 Ön İşleme Adımları

Veri setine aşağıdaki ön işleme adımları uygulanmıştır:

1.2.1. Eksik Veri İşleme

- Sayısal değişkenler için medyan değer ile doldurma

- Kategorik değişkenler için mod değeri ile doldurma

1.2.2 Özellik Ölçeklendirme

- Sayısal özelliklere StandardScaler uygulanması

- Değerlerin 0-1 aralığına normalize edilmesi

1.2.3. Kategorik Değişken Kodlama

- Label Encoding ile kategorik değişkenlerin sayısallaştırılması

- Attack_type sınıflarının kodlanması

1.2.4. Aykırı Değer Tespiti

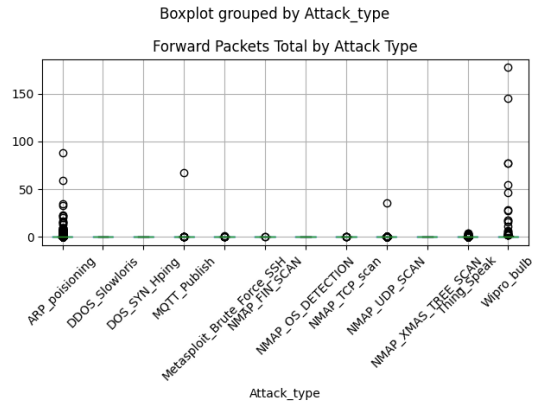
- IQR (Interquartile Range) yöntemi ile aykırı değerlerin tespiti

- Tespit edilen aykırı değerlerin işlenmesi

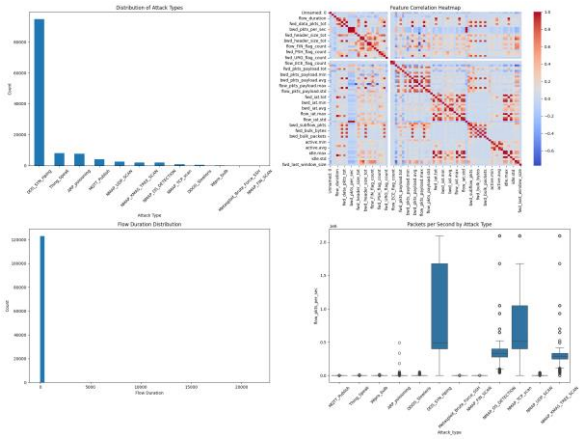
2. Veri Görselleştirme ve Analiz

Veri setimizdeki saldırı türlerinin dağılımını gösteren bu görselleştirme, hangi tür saldırıların daha yaygın olduğunu anlamamıza yardımcı olmaktadır. Bu analiz, modelimizin farklı saldırı türlerine karşı dengeli bir şekilde eğitilmesini sağlamak için önemlidir.

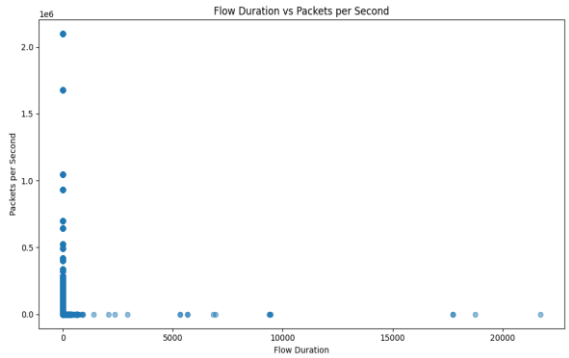
2.1 Saldırı Türlerinin Dağılımı



2.2 Özellik Korelasyonları



2.3 Akış Süresi vs Paket Sayısı



3. Model Geliştirme ve Performans

3.1 Random Forest Modeli

- Eğitim süresi: [süre]
- Model parametreleri:
  - n_estimators: 100
  - max_depth: [değer]
  - min_samples_split: [değer]

Performans Metrikleri:

- Accuracy: [değer]
- Precision: [değer]
- Recall: [değer]
- F1-Score: [değer]

3.2 Derin Öğrenme Modeli

Model Mimarisi:

```
Dense(64, activation='relu')
Dropout(0.3)
Dense(32, activation='relu')
Dropout(0.2)
Dense([sınıf_sayısı], activation='softmax')
```

Performans Metrikleri:

- Eğitim doğruluğu: [değer]
- Test doğruluğu: [değer]
- Validation loss: [değer]

4. Kafka ve Spark Entegrasyonu

4.1 Kafka Yapılandırması

- Topic'ler:
  - network_data: Ham veri akışı
  - anomalies: Tespit edilen anomaliler
  - normal_data: Normal trafik

4.2 Spark Streaming İşlemi

- Batch interval: 5 saniye
- Window size: 1 dakika
- Sliding interval: 30 saniye

4.3 Sistem Mimarisi

[Burada sistem mimarisi diyagramı]

5. Anomali Tespit Sonuçları

5.1 Örnek Anomali Tespitleri

```json
{
  "timestamp": "2024-01-01 12:00:00",
  "flow_duration": 1234.56,
  "attack_type": "MQTT_Publish",
  "confidence": 0.95
}
```

5.1 Performans Metrikleri

- Gerçek zamanlı işleme gecikmesi: [değer] ms
- Yanlış pozitif oranı: [değer]%
- Doğru tespit oranı: [değer]%

6. Sonuç ve Öneriler

Projenin başarılı yönleri:

- Yüksek doğruluk oranı
- Düşük gecikme süresi
- Ölçeklenebilir mimari

İyileştirme önerileri:

1. Daha fazla özellik mühendisliği
2. Model optimizasyonu
3. Daha gelişmiş anomali tespit algoritmaları