**SİSTEM ANALİZİ VE TASARIMI Dersi**

**Dönem Projesi**

Siber Güvenlik

Yapay Zeka İle Nükleer Tesislerin Siber Güvenliğinin Sağlanması

***Mehmet Hallaç*** (**221120221060**)

***Zehra Cıroğlu*** (**221120221010**)

***Eren Savaşır*** (**221120221052**)

***Abdülsamet Ünverdi*** (**221120221050**)

***Tunay Yıldız*** (**221120221031**)

Konumuz içerisinde siber güvenliğin, nükleer tesisler açısından önemine, işleyişindeki zorluklara ve bu zorluklara karşı yapay zekayı nasıl kullanacağımız üzerine çalıştık. Siber güvenlik ve nükleer tesisler hakkında bazı temel bilgiler ve makine öğrenmesi, makine öğrenmesi algoritmalarını açıklayarak konunun anlaşılacağı bir temel hazırladıktan sonra projeyi geliştirme sürecimizi, hangi algoritmayı neden kullandığımızı, bunların başarı oranlarını detaylıca açıklayacağız.

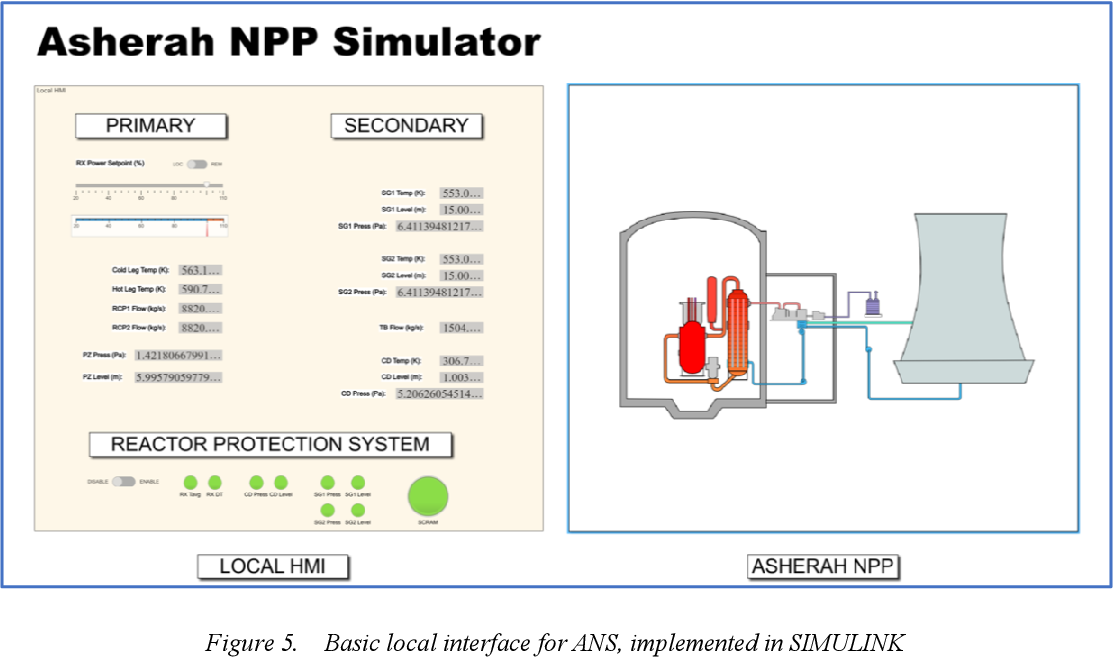
Konumuzu spesifik olarak seçme sebebimiz, nükleer tesislerin siber güvenliğinin sağlanmasının diğer tesislere ve kurumlara oranla daha zor olması. Bu zorluğun sebebi nükleer tesislerin çalışması esnasında peneterasyon testlerin yapılması işleyişi bozabileceği için oldukça riskli, fakat tesisin işleyişini durdurmak da oldukça fazla maddi kayba yol açacağı için bu tesisler yılda 1-2 kez kesinti yapıp testler gerçekleştiriyorlardı. Bu da güncel güvenlik zaafiyetlerine karşı tesisin savunmasız olmasına yol açmakta.

Bu sorunu çözmek için nükleer tesislere yapılabilecek olan başlıca 3 saldırı tipini tespit etmek üzerine bir çalışma yürüttük. Bunun için çeşitli makine öğrenmesi algoritmalarını kullanarak yapay zeka eğitmeyi amaçladık. Bu yapay zeka tesisin ağını anlık olarak izleyerek oluşacak anomalileri tespit edecek, bunların saldırı olup olmadığını ve saldırı ise saldırı tipini bize verecek. IT ve ICS ataklarını çok yüksek oranlarda doğru tespit eden yapay zekamız NPP saldırılarında %70lik bir başarı oranı yakalamıştır. Bu oranın artırılması elimizdeki verilerin çeşitliliği ve fazlalığı ile mümkündür. Fakat Sivrihisar şartlarında internet hızımızın düşük olması maalesef kısıtlı süre içerisinde daha fazla veri işlemimizi mümkün kılmadığı için sistemimizin şimdilik başarı oranı bu kadardır. İlerleyen bölümlerde grafiklerle daha detaylı şekilde başarı oranlarımızı ele alıp değerlendireceğiz.

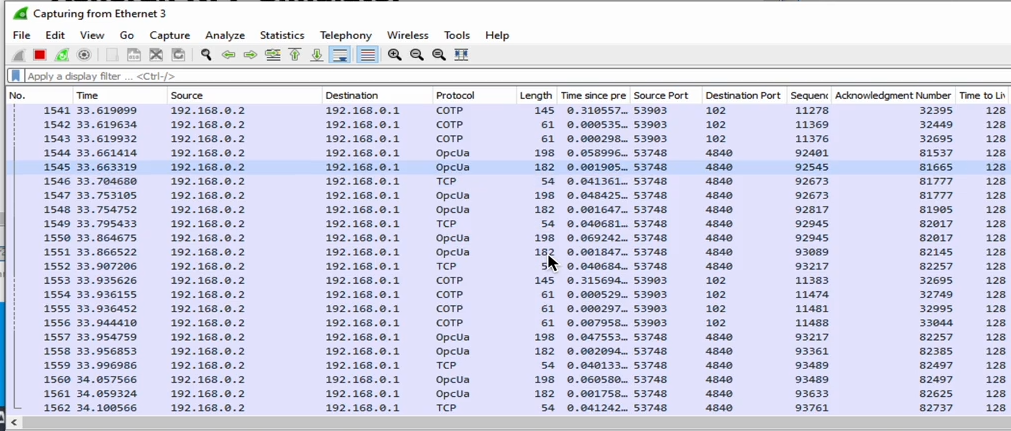
Projemizde başlıca 3 saldırı tipinin tespiti üzerinde durduk. Bunlar IT, ICS ve NPP hack saldırıları. Bu saldırılar içerisinde de en tehlikeli saldırı tipimiz NPP saldırılarıdır. Saldırı tiplerini açmak gerekirse IT saldırları HMI’dan PLC’ye erişimi engelliyorlar. Bu saldırı tipini DDOS saldırılarına benzetebiliriz. Ağ içinde sahte paketler ile bir yoğunluk yaratarak iletişimin hızını yavaşlatıyor. Bu saldırının tespiti için Cycle time dediğimiz bloklar arası iletişim hızı verisi üzerine odaklandık ve yapay zekamızı bu veri ile besledik. Cycle Timedaki dalgalanmalar, yavaşlamalara bakarak saldırımızın IT saldırısı olduğunu anlıyoruz diyebiliriz basitçe. ICS saldırıları ise çalışan PLC’yi kapatmak için bir paketin taklit ederek makinelere aç, kapat komutlarını veriyor. Bu saldırı tipinin tespiti için ise PLC’lerin anlık durum verisi üzerinde durmamız gerekmekte. Bu saldırının tespiti için ayrıca s7comn veri tipindeki girdiler saldırı göstergesi olarak kullanılabilirdi. Bu veri tipi ağ dışından bir veri geldiğini gösterir. Fakat nükleer tesisler intranet ile çalıştığı için ağ içerisinde bu tipte veriler olmayacaktır, simülasyonumuzda olabilir çünkü sanal makineyi ana makinemize bağlıyoruz. Kısaca bu veri tipini bir saldırı göstergesi olarak yapay zekaya tanıtmadık çünkü kolaya kaçmak olurdu ve eğittiğimiz yapay zekada gerçekçi olmayan bir yüksek başarı oranı elde edilirdi. Bunun yerine şunu kullandık; açık bir PLC’nin kapanması kapalı PLC’nin açılması gibi devredeki değişiklikler incelenerek anomali tespit edilebilir. NPP saldırıları ise en tehlikeli olan saldırı tipidir. Stuxnet saldırısı NPP saldırısına bir örnektir. Stuxnet saldırısına kısaca değinmek gerekirse; İran’ın Natanz kentindeki uranyum zenginleştirme tesisine yapılmış bir NPP saldırısıdır. Saldırı için yazılan solucan Siemens’in ürettiği PLC’leri hedef alarak santrifüjleri konrol eden sisteme zarar vermeyi hedefliyordu. Santifrüjlerin dönme hızını artırarak tesiste fiziksel hasara yol açtı. Ayrıca bunu yaparken operatörlere her şey normalmiş izlenimi verdi. NPP saldırılarının tespiti için ise yapay zekaya önce normal verileri daha sonra saldırı yapılmış verileri vererek anomaliyi tespit etmesini sağlamayı amaçlıyoruz. LSTM dediğimiz makine öğrenmesi algoritması ile bunu yapacağız.

Yapay zekayı eğitmek için öncelikle verilere ihtiyacımız var. Bu verileri sağlamak için Siemens’in PLCsim ve TİAportal gibi programlarını kullandık. Bunlar dışında MATlab Simulink’in de oldukça yardımı dokundu. Öncelikle 3 farklı alanda veriler topladık, bunlar; Ağ trafigi verileri, PLC verileri ve Process verileri. Bu verilerin normal halleirni ve saldırı altındaki hallerini kaydedip çeşitli makine öğrenmesi algoritmalarını deneyerek en verimli çalışanı bulmak için çalıştık. Ağ trafiği ve PLC datalarını, IT ve ICS atakları için kullandık ve bunlar için bir kaç farklı makine öğrenmesi algoritması denedik. Bunlara daha sonra detaylı değineceğiz. Process verilerini de NPP saldırılarını tespit etmek için kullanacağız, bu saldırı tipinin tespiti diğerlerine göre daha zor çünkü ayrıca odaklanacağımız bir veri parçası yok. Bu sebeple normal process ve saldırı altındaki process verilerini LSTM modeliyle eğiteceğimiz yapay zekaya verip süreç içindeki anomaliyi kendisinin anlamasını sağlamayı planladık. Yapay zekamız, diğer saldırı tiplerine kıyasla daha düşük bir başarı elde etmiş olsa da %70 oranla saldrıları tespit etmeyi başardı.

IT saldırıları için gerekli olan Cycle Time (Bloklar arası iletişim süresi) verisini TiaPortal programı bize vermediği için, kendimiz bir logic kodlayıp, PLC’ye indirip bunu simüle etmemiz ve simülasyonu izleyip cycle time verisini almak için bir Python scripti yazmamız gerekti. Bu ve yazdığımız başka yardımcı programları da ödev dosyamızın içerisne yükledik. Matlab Simulink’e Asherah NPP simulator ile bir nükleer güç santrali simülatörü indirip kurduk. Bu simülasyon sayesinde verileri elde edeceğiz. Bu süreç içerisinde internet hızı bize oldukça zorluk çıkarsa da en verimli şekilde sorunu çözmeyi başardık.

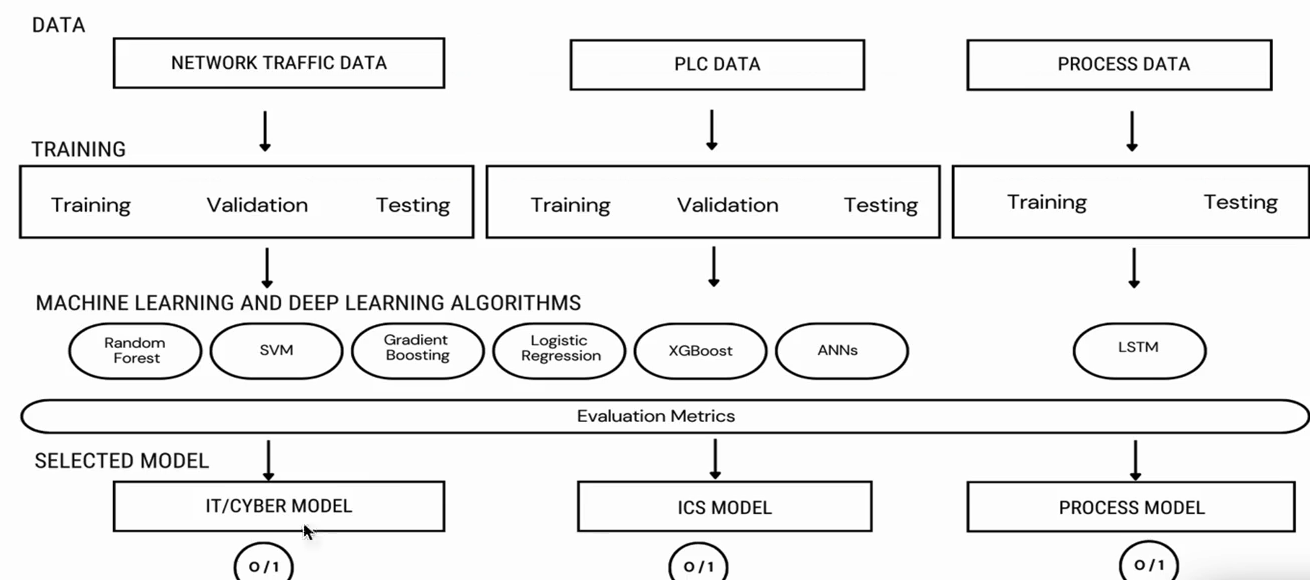


Bu simulasyondan aldığımız verileri PLC’ye verip PLC’deki verileri de Python’da yazıdğımız script aracılığı ile çekiyoruz. Ayrıca simülasyonlar çalışırken ağı Wireshark programı ile izleyerek ağ trafiği verilerini de elde ediyoruz. Verileri Python ile çekerken OpCua Server kullandığımız için ağ içinde o verileri de izliyordu wireshark, bunu filtrelemek için OpCua serverin kullandığı portu bulup filtreleyerek Wireshark’ın izleme alanı dışında tuttuk.



Normal verileri topladıktan sonra saldırı anındaki ağ verilerine de ihtiyacımız var ve saldıracağımız sistem bir nükleer tesis olduğu için hazır bir tool bulmak imkansız. Bu sebeple Python’da kendi saldırı toollarımızı yazdık. Bu toollar da ayrıca ödev dosyası içinde mevcut.

Normal ve saldırı altındaki verilerimizi aldık ve kaydettik. Sıra yapay zekayı eğitmeye geldi. Ağ trafiği ve PLC verileri için bir kaç farklı makine öğrenmesi modeli kullanacağız ve en doğru sonucu veren algoritmayı detaylı kıyaslama sonuçları ile elde edeceğiz.



Ağ Trafigi verileri ve PLC verileri için kullanacagimiz modeller: Random Forest, SVM, Gradient Boosting, Logic Regression, XGBoost, ANNs

Process Data icin ise LSTM modelini kullanacagiz.

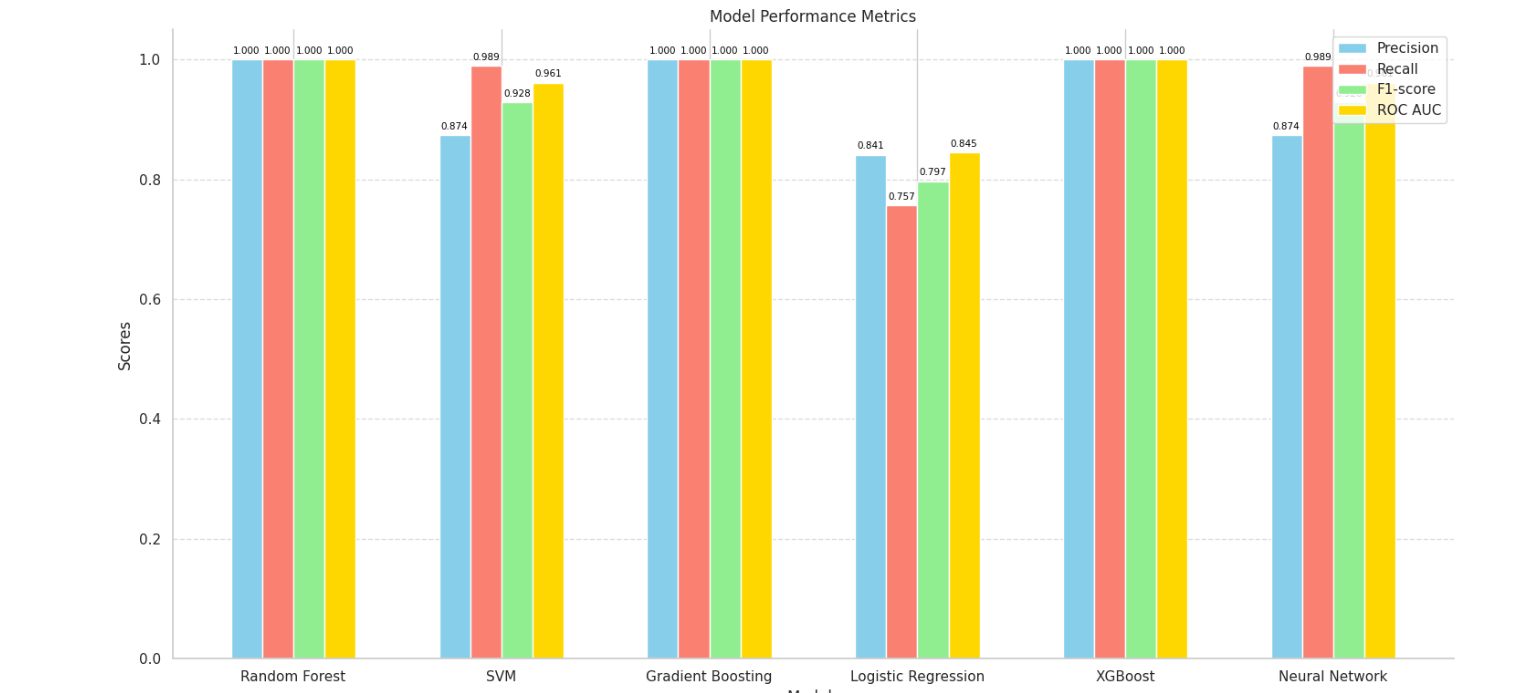
Modellerimize kısaca değinmek gerekirse:

* **Random Forest:** Karar ağaçlarının bir topluluğudur. Her ağaç rastgele bir alt veri kümesiyle eğitilir ve sonuçlar çoğunluk oyu ile belirlenir. Bu yöntem aşırı uyum (overfitting) riskini azaltır.
* **SVM (Support Vector Machine):** Veriyi sınıflandırmak için en iyi ayrım çizgisini (hiper düzlemi) bulur. Margin dediğimiz bu çizgiye en yakın veriler karar sınırını belirler.
* **Gradient Boosting:** Karar ağaçlarının sıralı olarak eğitildiği bir tekniktir. Her yeni ağaç, önceki ağaçların hatalarını düzeltmek için eklenir.
* **Logistic Regression:** İkili sınıflandırma problemleri için kullanılan basit ve etkili bir yöntemdir. Olasılıkları kullanarak veriyi sınıflandırır.
* **XGBoost:** Gradient Boosting'in optimize edilmiş bir versiyonudur. Daha hızlı ve daha performanslıdır.
* **ANNs (Artificial Neural Networks):** Yapay sinir ağları, birbirine bağlı nöronlar kullanarak veriyi işler ve öğrenir. Özellikle karmaşık ve büyük veri setlerinde etkilidir.
* **LSTM (Long Short-Term Memory):** Zaman serisi verileri ve sekans verileriyle çalışmak için tasarlanmış bir RNN (Recurrent Neural Network) türüdür. Uzun vadeli bağımlılıkları öğrenmede başarılıdır.

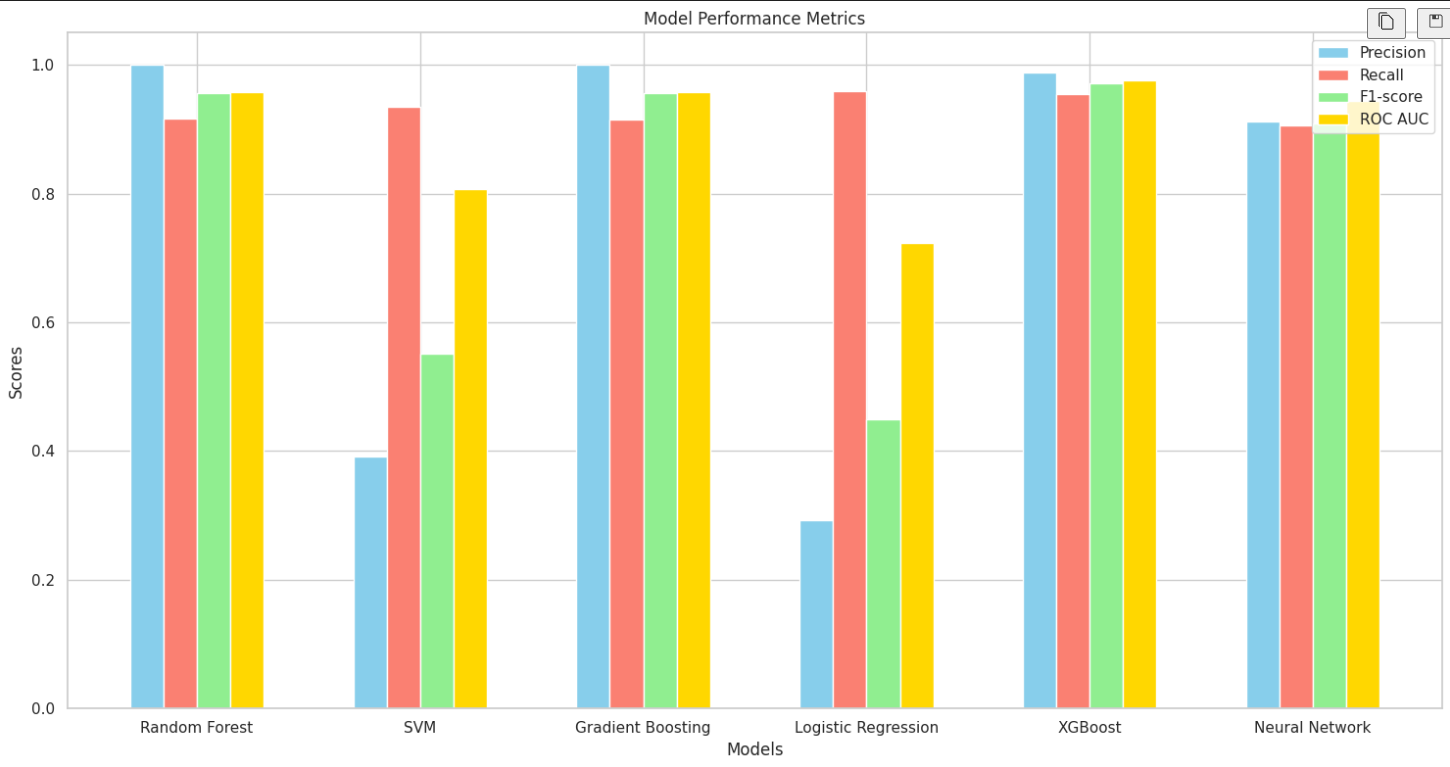
Bu modellerin eğitimine başlamadan önce eliimizdeki verileri preprocessing’den geçirmemiz gerekiyor. Bu süreçte kısaca verilerimizi numerik hale getirip normal ve abnormal olarak sınıflandıracağız. Ağ trafiği verilerini IP adresine, protokole ve kayanğa göre sınıflandırdık. Örnek vermek gerekirse IP adreslerini numerik hale getirmek için 0 ve 1 durumunu güvenilir ve güvenilmez ipler olarak düşündük. 0 durumuna PLC’lerimizin IP adreslerini girdik, 1ler ise güvenilmez IP’ler bunlar else kod bloğu içerisinde. Yani 0’a tanımlı IPler hariç bütün IPler güvenilmez ve abnormal. Ayrıca kolaylık olması açısından PLC IP’lerini statik yapıp sisteme girdik, dinamik olsa idi IP’ler değişeceği için güvenilir IPler de değişecekti. Bu durumu aşmak için doğrudan IP tanımlamak yerine PLC’lerin IP’sini çekip Python’da .format dediğimiz komutla 0 durumu içine atamak kullanılabilir. Zamanın kısıtlı olması ve projenin demo olması dolayısıyla şimdilik kullandığımız yöntemde bu kolay yolu tercih ettik. Ayrıca nükleer tesisler intranet kullandığı için arp spoofing saldırısına maruz kalma ihtimali çok çok düşük. Bu sebeple yapay zekamızı arp spoofing saldırılarını tespit etmeye yönelik eğitmekle zaman harcamadık ve arp verilerini güvenli olarak etiketledik. Eğitmek isteseydik bunu şu yolla yapabilirdik; Arp spoofing saldırılarında saldırgan cihaz kurbana modem modeme ise kurbanmış gibi gözükür ve veri akışınnda modemle kurban cihaz arasına konumlanır. Bunu da Mac adreslerini kopyalayarak yapar. Yani belli aralıklarla Arp –a komutu çalıştırılıp arp tablomuz incelenebilir ve değişen Mac adresi varsa abnormal olarak etiketlenerek Arp saldırısının tespiti sağlanabilirdi.

Plc verileri için ise, çalışma durumunun true’dan false, false’dan true olmasına göre ve pump speed gibi bazı PLC’lerin kontrolündeki verilerin kısa zaman içerisindeki ani değişimlerini baz alarak abnormal olarak etiketleme yaptık.

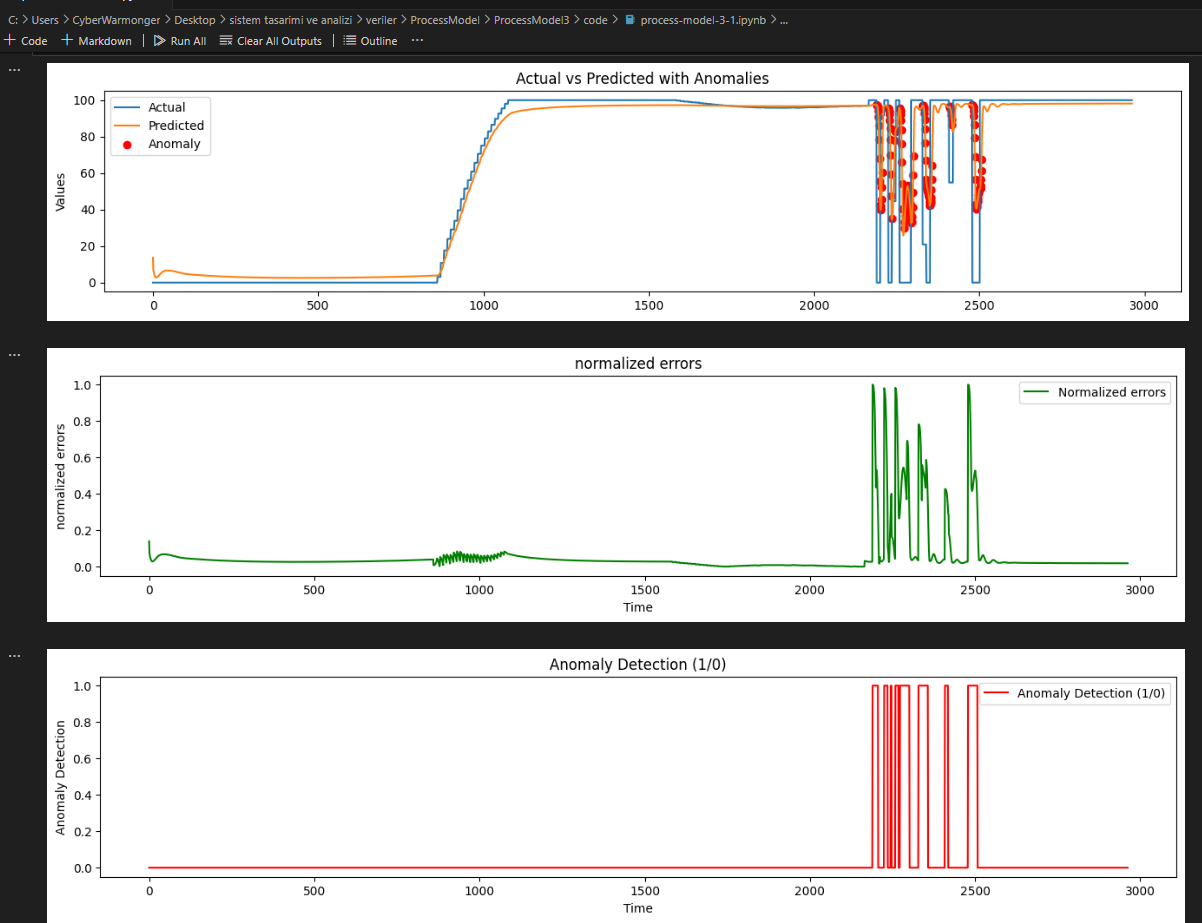
Process verilerimiz için ise herhangi bir ekstra etiketleme kullanmadık. Önce normal daha sonra saldırı yapılmış verileri etiketleyip modelimize okuttuk. Sistemdeki anomalilileri tespit etmesini sağladık.

Makine Öğrenmesi Modellerinin IT saldırılarının tespitindeki başarı oranları:

Plc verilerini kullanarak ICS saldırılarının tespitindeki başarı oranları:



LSTM Modelinin Anomalileri Tespiti:

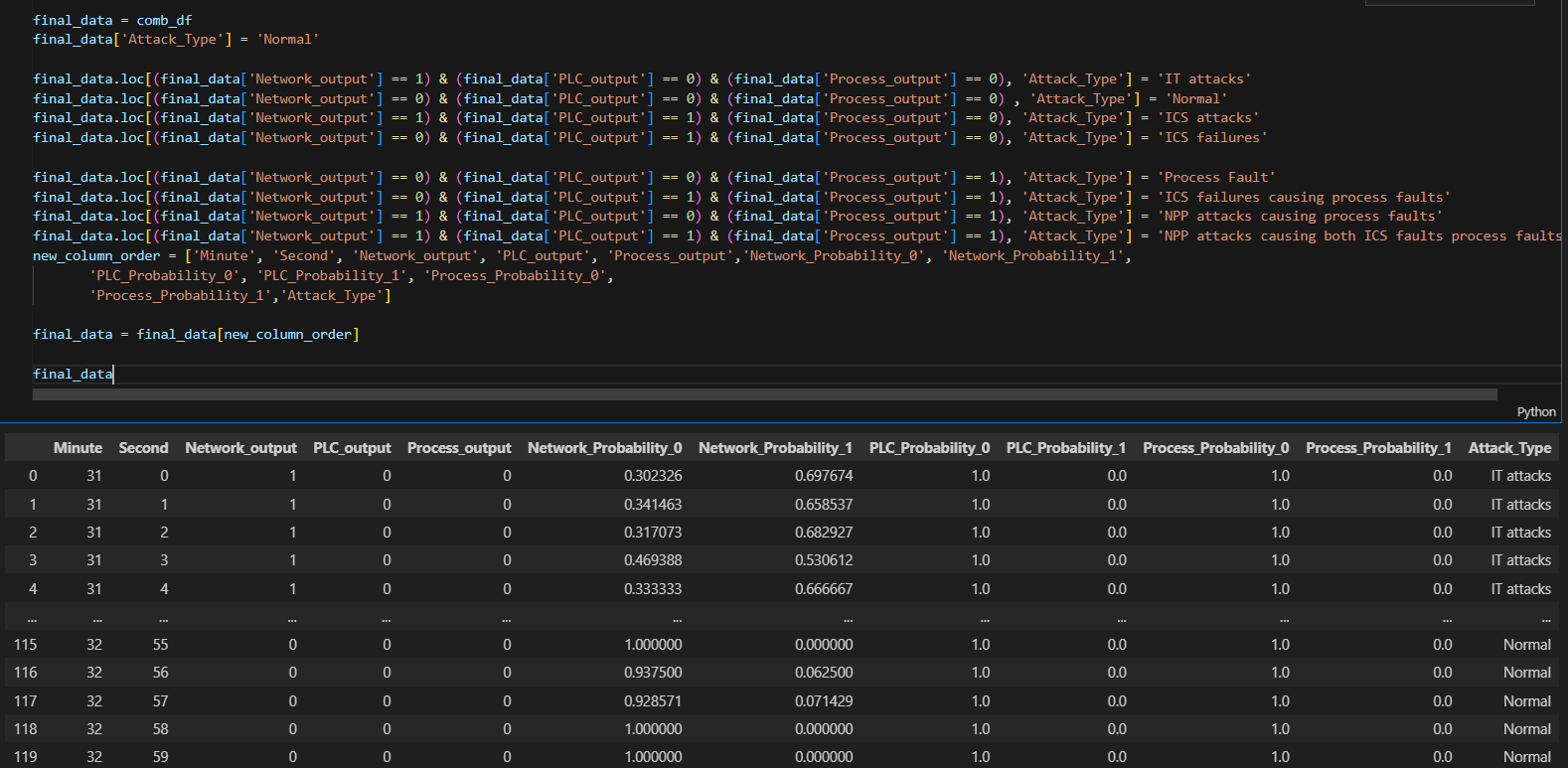


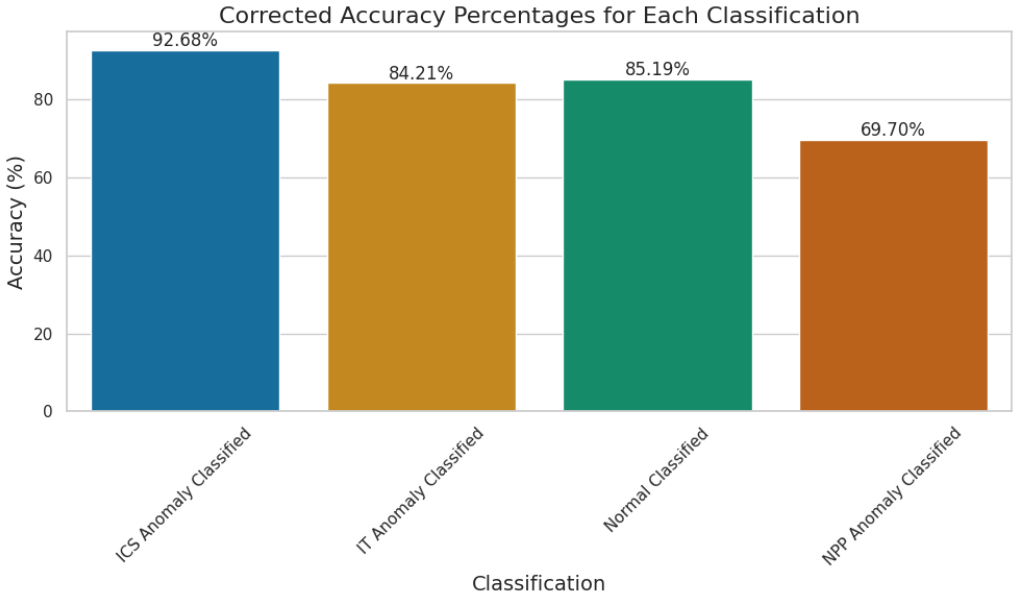
Ayrıca modellerın Confussion Matrixle anaizleri yani hangi modelin ne kadar yanlış tahminde bulunduğunun verileri ödev dosyamız içinde mevcut. Raporu fazla uzatmamak için buraya koymuyorum.

Makine öğremesi modellerimizi kullanmak ve yapay zekayı eğitmek için kaggle platformunu kullandık.

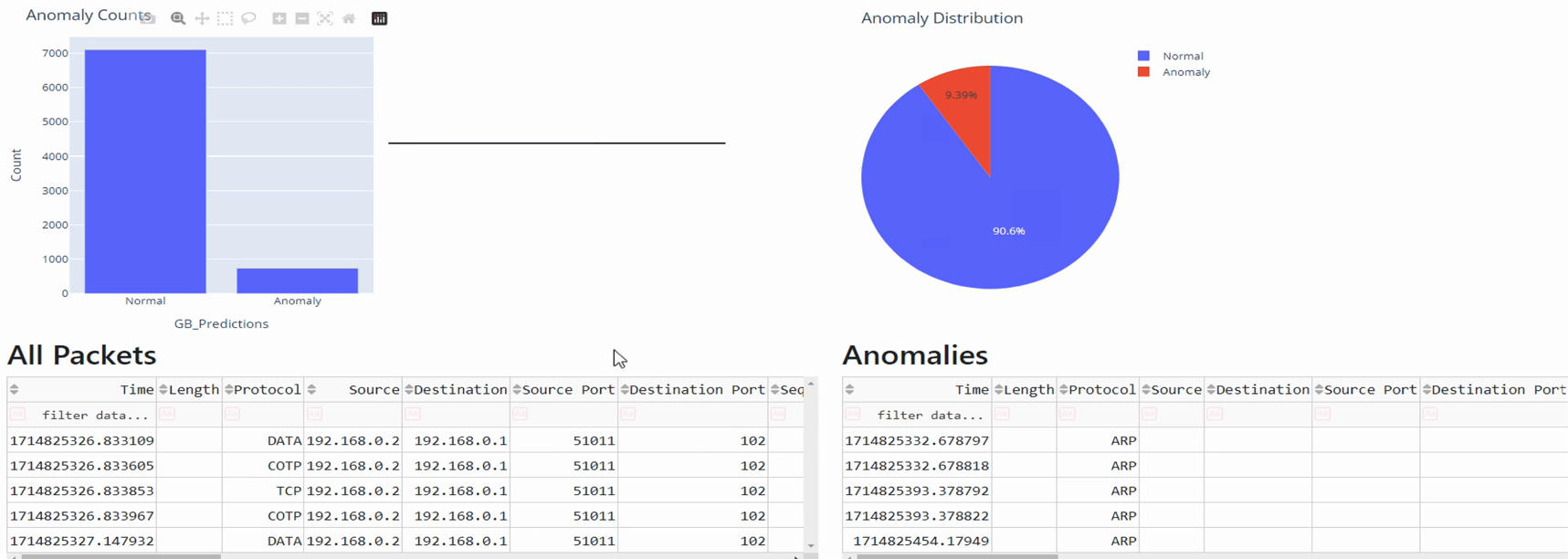
Tek tek bütün saldırı tipleri için yapay zekamızı ayrı ayrı eğittikten sonra, her bir saldırı tipinin çıktısını alarak tekrar bir eğitim deniyoruz. Bunun amacı yapay zekanın anomali olduğunu tespit etmesinin üstüne tespit ettiği anomali, tesisin hangi saldırıya maruz kaldığına işaret ediyor bunu anlayıp ona göre uyarı vermesi. Nükleer tesise yapay zekanın müdahale edecek olması oldukça riskli bir durum olduğu için şu anlık yapacağı en iyi şey operatörleri, sorumluları bilgilendirecek uyarıyı vermesi olacaktır.

Saldırı tipini sınıflandıran kod bloğu.



Saldırı tiplerine göre sınıflandırma yapılmasındaki tahmin/başarı oranları. 

Uygulamanın basit bir Demosu: (Saldırı öncesi ve saldırı anını gösteren ssler ödev dosyasında mevcut.)



KAYNAKÇA:

https://www.semanticscholar.org/paper/DEVELOPMENT-OF-THE-ASHERAH-NUCLEAR-POWER-PLANT-FOR-Silva/215eb3868e595d81fe75496618fae8aebc80674a

https://conferences.iaea.org/event/181/contributions/15642/attachments/8548/11374/cn274\_Full\_Paper\_Rodney\_Busquim\_e\_Silva\_ANS\_Final.pdf