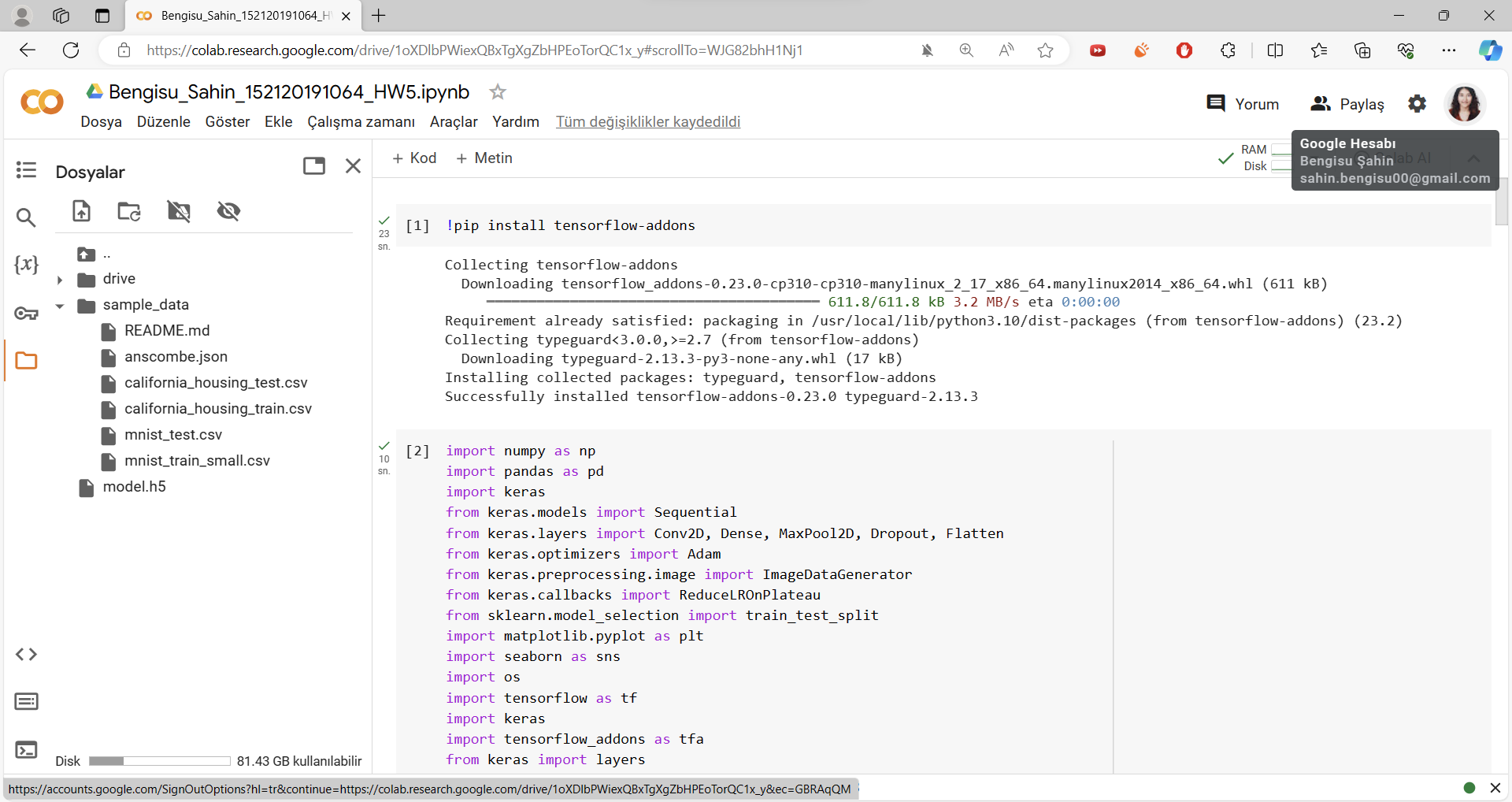
**Bengisu Şahin**

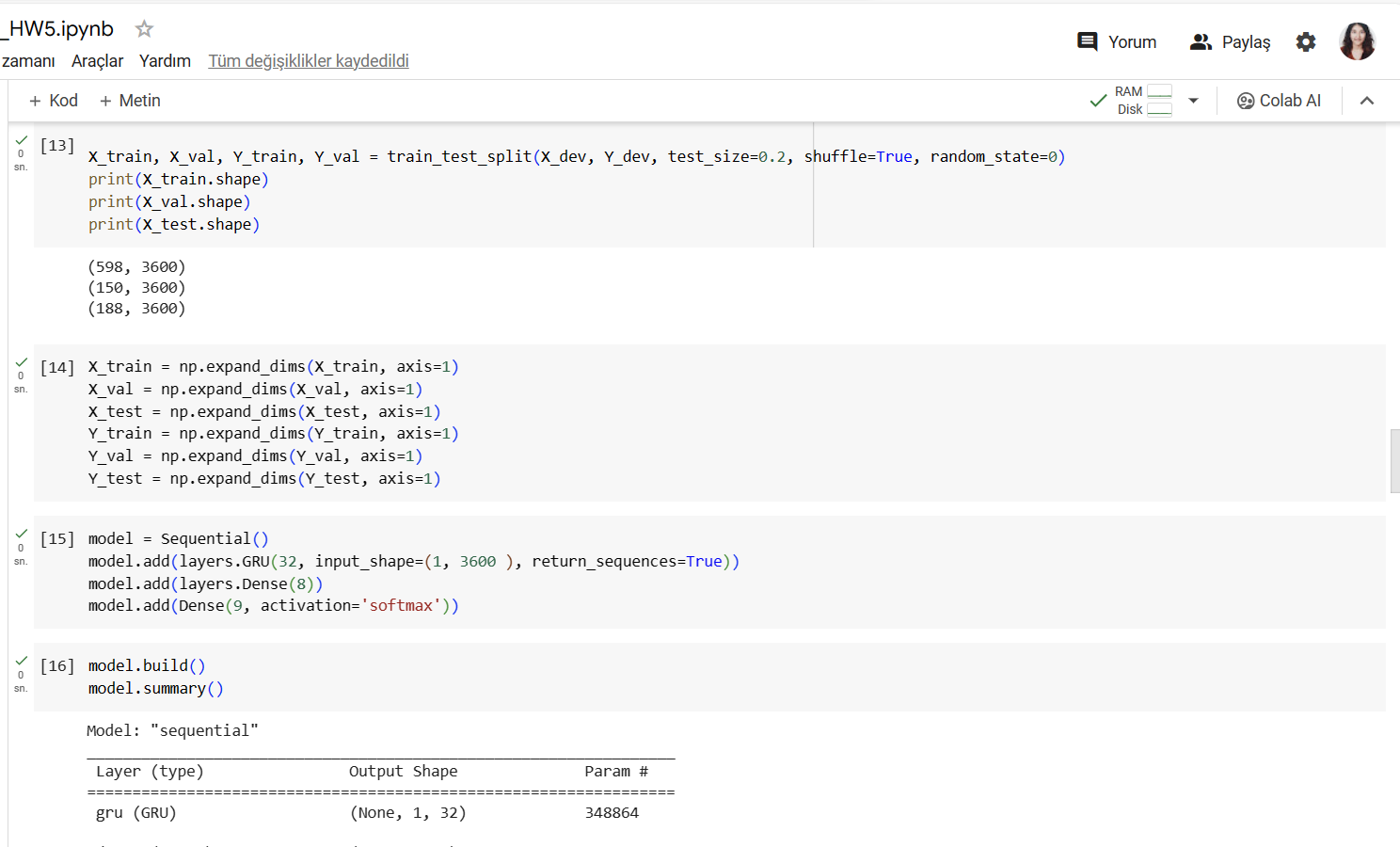
**152120191064**

**INTRODUCTION to DEEP LEARNING HW5 Report**

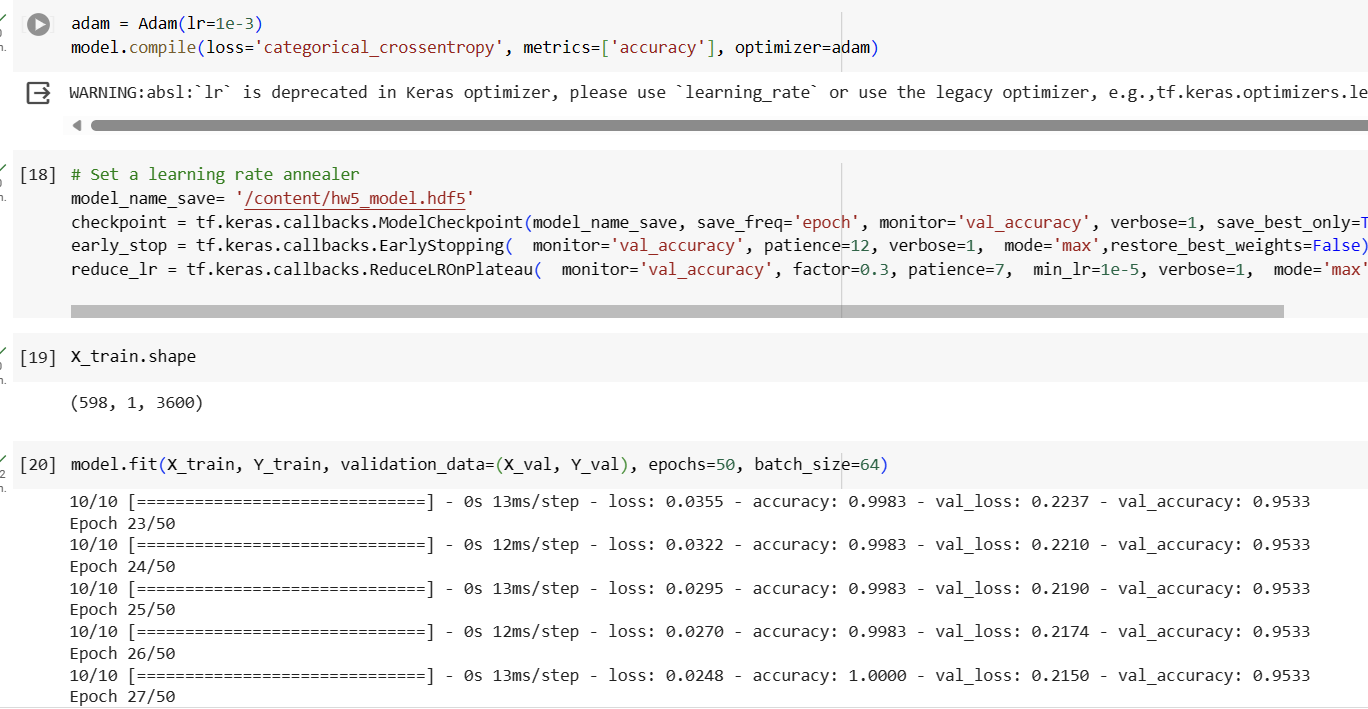
Gear Fault Datasetinin LSTM, GRU, CNN1D ya da Dense NN sınıflandırıcılarını kullanarak sınıflandırılması, model oluşturulması ve oluşturulan modelin testinin yapılması istendi. Classifier olarak GRU ve Dense classifierlarını kullanıldı ve modele layerlar eklendi.



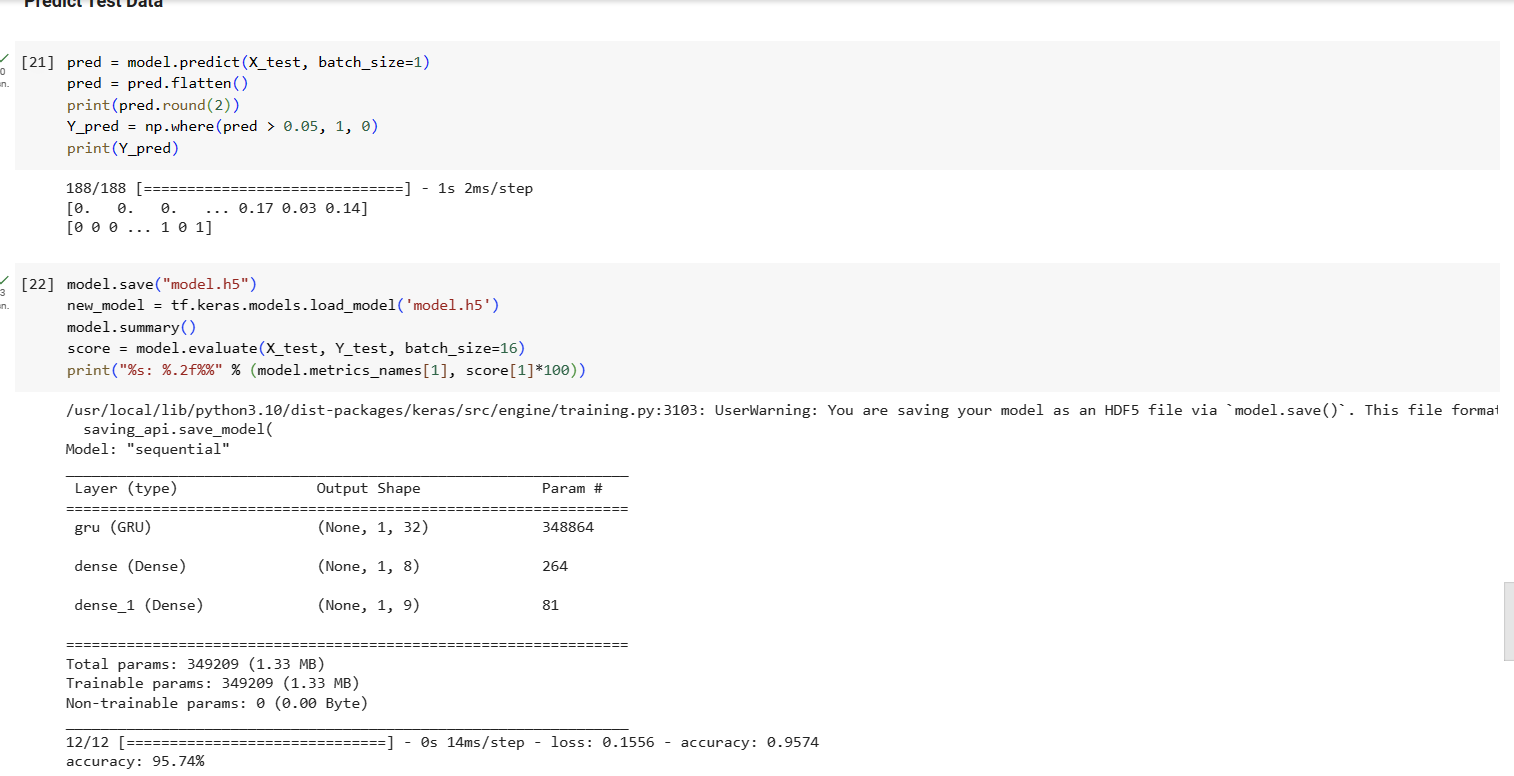
* Gerekli kütüphaneler kuruldu ve uygulamaya import edildi. Sonrasında drive hesabımdan ödev dosyasında verilmiş olan data çekildi.Sonrasında google drive’da bulunan dataset np.load fonkisyonu ile X\_trian değişkenine alındı ve transpose işlemi ile de kodda kullanıma hazır hale getirildi. Types değişkeni altında kaç çeşit sınıflandırmanın olduğu belirlendi ve sonraki adımda da bu sınıflandırmaların label isimleri oluşturuldu ve ekrana basıldı.
* Modelin eğitim adımında kullanılmak üzere Y\_train ve X\_train değişkenleri oluşturuldu.
* Aşağıdaki görselde görülebileceği gibi model oluşturmak için gru ve dense kullanıldı ve katmanlar eklendi. Katmanlar ekledikten sonra modelin özeti ekrana basıldı, bu şekilde modelde bulunan katmanlar, katman özellikleri, parametreler gibi özellikler görülebilir hale geldi.



* Yardımcı kod dosyalarındaki kodlar kullanılarak checkpoint, early stop gibi özellikler ayarlandı. Sonrasında da model.fit fonksiyonu kullanılarak modelin eğitimi gerçekleştirildi. Epoch değeri olarak 50 verildi.



* Ardından metrik ölçümlerinde kullanılmak üzere Y\_pred değişkeni oluşturuluyor, bu değişken oluşturulurken model.predict fonksiyonu kullanılıyor. Öncelikle pred isimli değişkene değer atanıyor ve alınan değerleri sonrasında ölçümlerde kullanmak için belirli adımlardan geçiriyoruz. Flatten yaparak tek boyutlu dizi haline getiriyor. 0 ve 1 olarak ayarlamamız da metrik ölçümlerinde float değerlerde hata vermesi binary değer istemesinden kaynaklıdır.Sonrasında modeli save fonksiyonu ile kaydediyoruz. Ve Modelin test işlemini model.evaluate fonksiyonuna X\_test ve Y\_test değişkenlerini göndererek yapıyoruz ve son olarak da modelin accuracy değerini hesaplayıp ekrana bastırıyoruz. Şekil 7’de de görüldüğü üzere modelin accuracy değeri %95.74.



* Son olarak Y\_test ve Y\_pred değişkenlerinin shapelerinin aynı olmasını sağlıyoruz. accuracy\_score ile accuracy değerini oluşturup ekrana print ile bastırıyoruz, confusion\_matrix ile de confusion matrixini oluşturuyoruz ve plt ile de ekrana grafik olarak bastırıyoruz.

