A problem that interests me is the detection of depression using information from music streaming apps such as Spotify, Youtube Music, Apple Music, etc.

This can be done by creating an AI model that can understand the type and genre of music that a person listens to most frequently and classify the user as potentially having depression or not.

In terms of data, firstly, we would need a list of all the songs that are currently available on this app, along with their genres. Second, we would need to classify the songs into different categories such as sad, depressing, heartbreak, happiness, disco, rock, etc. There are already certain AI models that perform this task; however, if something cannot be found for this task in particular, we would need the lyrics to the songs, and then we can create a model to assign the categories. In order to make a decision for the users, we would also need to keep track of the music that they listen to and which songs they prefer to listen to the most.

If there aren't any other models to find the labels for the songs, an LSTM or GRU model would be good for the job. LSTMs have a memory line, also called a "cell state," that can hold on to information for longer. Hence, it would be appropriate to use LSTMs since they would need to remember the lyrics. Using an LSTM model, we can assign labels to the songs. Once we have the labels for each song, we would need a neural network classification model to classify the music preference of the user as depressed or not depressed.

To train the neural network to tell if a person is depressed or not, we would need to know if the person is actually depressed or not. Then, we can use different metrics, like accuracy, precision, recall, and the F1 score, to judge how well the model works. In this particular case, reducing the False Negatives is really important since falsely classifying a depressed person is much more damaging than falsely classifying a healthy person as depressed. Hence, in such a case, we can consider using recall as the main metric.