Hi, my name is Sheen. This is the I B M Advanced Data Science Capstone. The dataset for this project is obtained from Kaggle, New York City current job posting. I will be using this dataset to build a content based recommender system for jobs. The use case is as the name suggests, recommend relevant jobs to individuals who satisfies certain requirements which will be shown in my Jupiter notebook. The goal of this project is to recommend jobs to someone according to one's interests, skill sets and education, etc.

In order to get there, I have done the following work. Let's begin with architectural choices. I used Apache Spark accompanied by Pandas. I originally planned to use spark data frame all the way through. Unfortunately, I was not able to complete everything I wanted to achieve. Also, I used up my capacity unit hours on I B M cloud. As a result, I had to switch to pandas data frame instead. The good news is that I achieved what I wanted to do with the data in a short time span thanks to the small dataset itself. In the near future, I will try to do it with spark again.

Next, let's talk about data quality assessment, data preprocessing and feature engineering. I started out by simply read the C S V file as a spark data frame. I immediately noticed that there are missing values. After few attempts, I realized that I cannot drop rows or columns with missing values because it will reduce the dataset drastically. For this reason, imputing missing values is preferred here. However, I can only impute the part-time/full-time indicator column since I can easily see which job is part-time given the salary level. As for other missing values including minimum qualification requirements and preferred skills, I cannot impute them since each job gives different information. From here, I start to inspect each column separately and I found wrong data type issues. For example, the salary value should be integer rather than strings. I converted the value into correct data type. Next I combined multiple columns with string values into one and extract keywords from the combined column. I then used the keywords to one-hot encode the dataset.

Now we are ready to talk about model training. There are two models used in this project. One is based on machine learning and the other is based on deep learning. For the machine learning model, I used T f I d Vectorizer and Cosine Similarity Matrix, because I need to weigh a keyword in the combined column and assign the importance to that keyword based on the number of times it appears in the column. And I need to compare the similarity between different jobs. For the deep learning model, I used keras sequential model, because it is the simplest model I know and it allows us to create models layer- by-layer for most problems. For more details, you can look them up in my Jupiter notebook. The model performance indicator for both machine learning and deep learning is accuracy, since I want to know how relevant or similar among jobs in this project. After model evaluation step, I looked for ways to improve the model performance. Here’s what I found or suspected, removing punctuations and digits will likely to improve performance because it is difficult to extract keywords when having different types of punctuations. In addition, digits in keyword provides no information for our model. In fact, we lost information simply by extract keywords. In addition, increasing the number of epochs seemingly led to better performance.

That is all. Thank you.