1] Explain how will you build the dataset. If we tell you the requirements?

**=>**

* First, we have to define the problem then accordingly we have to prepare the dataset.
* Let’s assume you want to predict stock price in the future. for that, you have to look for the API’s where we can get the Stock Market data for forecasting.
* If our problem is based on Image Data. Then first we have to know our labels or what we want to predict, then we can use google search or any APIs, if our problem is relevant to the API’s Dataset (eg. **COCO** **Dataset**) and accordingly we will label that dataset.
* If we are working on the text data then we can get that by Web Scraping and then we can preprocess so the noise can be removed.

2] Detail Case study on Pinterest recommendation system? How it works, If you wanted to clone it how will you do it. Mention the process. Which recommendation system do you like and why?

**=>**

**Case Study:**

* Pinterest came up with a unique recommendation system called Pixie. It is a graph-based system used for making personalized recommendations in real-time. Here is the [research paper](https://labs.pinterest.com/user/themes/pin_labs/assets/paper/paper-pixie.pdf).
* Pixie has powers to do recommendation across Pinterest in Related Pins, home feed and Explore, and accounts for about half of all Pins saved.
* It used random walk algorithm to estimate the Personalized PageRank to examines the portion of the graph nearest to nodes. it starts the walk from multiple Pins and find recommendations at the intersection of all of them.

**Pixie also came up with optimization steps:**

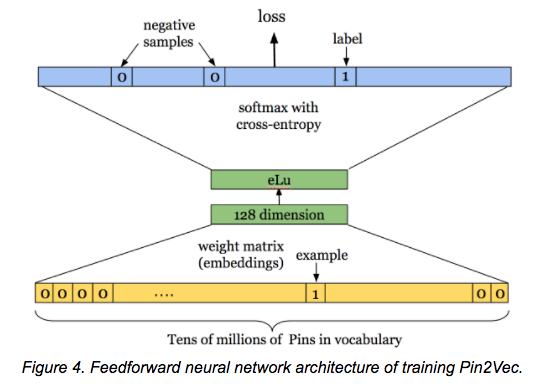
**1. Early Stopping:** In an ideal world, we’d only want to retrieve the top 1,000 most visited nodes, so we wouldn’t need to walk the complete 100,000 steps every time. To accomplish this, we keep walking until the rank 1,000 candidate gets at least 20 visits. From this optimization, we’re able to gain a 2x boost in performance.

**2. Graph Pruning:** The full Pinterest graph has over 100 billion edges, which is way more than we actually use, but we can remove some of those edges to make Pixie suit our needs. To prune the graph, we downscale the effect of popular Pins by implementing a function that provides a cap for the number of neighbors a Pin may have. We can also prune by getting ahead of users who may accidentally save something to the wrong board (which happens to the best of us). If we can identify those edges, we can remove them.

**3. Remove diverse boards (those with Pins from multiple different ideas).**

**How can we clone it?**

I have looked at the resources but I couldn’t find the exact procedure to clone it. However, I have read one [article](https://medium.com/the-graph/applying-deep-learning-to-related-pins-a6fee3c92f5e) on Medium where they mentioned, used a feed-forward neural network with a hidden layer of 128 dimensions and Elu Activation Function. This network is inspired by Word2Vec.



**Which recommendation system I like most:**

I have learned three type of recommendation systems till now and they are Content Based System, Collaborative Filtering Based System (Item-Item, User-User) and Matrix Factorization. In this method I liked Matrix Factorization because of the advantages to overcome with Sparse Matrix Problem (More values are Zeros and Less are One.) and easy to implement with the help of Surprise Model.

3] How can you turn your NLP chatbots or products into Cognitive (Something which can sense the feelings) Products. Describe the process in detail. Take one Product like chatbots/Phone-bot(bot over a call)?

**=>**

First thing, we know that, chatbots working on the basis of Intents. So, if you want from your chatbot to sense the feelings then we can mention the intents like happy, sad, etc. and based on that your chatbot can response.

**Chatbot Talking:**

* Let’s assume, we have a restaurant and we want to create a chatbot to speak with our customers and let them order with their tasting purposes. If customer ask about the “Spicy” foods then our chatbot will give them food list which are spicy that is available at that moment.
* If we are creating chatbot for Music, our chatbot can sense feelings based on the music preferences. If person asking about the “party songs” so as soon as our chatbot will understand the person is happy at that moment.

**Note:** if we want to build an assistant like Google or Alexa then we will go for speech to text algorithm so our chatbot can listen and speak too.