**Industry Classification**

Problem:

You can think of the job industry as the category or general field in which you work. On a job application, "industry" refers to a broad category under which a number of job titles can fall. For example, sales is an industry; job titles under this category can include sales associate, sales manager, manufacturing sales rep, pharmaceutical sales and so on

Solution:

The problem is supervised text classification problem, and our goal is to investigate which supervised machine learning methods are best suited to solve it. Given a new job title that comes in, we want to assign it to one of 4 industry categories.

The classifier makes the assumption that each new complaint is assigned to one and only one category. This is multi-class text classification problem.

**1.Which techniques you have used while cleaning the data if you have cleaned it?**

- Remove Duplicates

-Text preprocessing techniques

1- removing stop words (by adding some to default library and exclude “it”)

2- neglect words less than 2 letters

3- remove text noise

4- remove words that has digits in it

5- lemmatization and stemming

5- convert all to lowercase letters

**2.Why have you chosen this classifier? (E.g. I used Multinomial Naive Bayes because it is easy to interpret with text data and there are more than two outcomes).**

**I used linear svc**

The linear kernel is good when there is **a lot of features**. That's because mapping the data to a higher dimensional space does not really improve the performance.

Less parameters to optimize

It generalizes better

And controls overfitting

**3.How do you deal with (Imbalance learning)?**

I tried several approaches

1. Over Sampling
2. SMOTE
3. Weighted cost
4. Tried different evaluation metrics

And finally I choose to combine oversampling with weighted cost

**4. How can you extend the model to have better performance?**

1. Get more data to train with
2. Try different models may be (deep learning)
3. Algorithm tuning
4. Try to add more features

**5.How do you evaluate your model? (i.e. accuracy, F1 score, Recall)**

F1 score and precision