**JPMorgan Tech fest–2017**

**Vivek Mishra- Technical Architect**

**Rajesh Rajamanickam – Senior Programmer**

**Algorithm:**

**Step 1**: **Feature Extraction**

Sub Algorithm: Parse the resume using open source text parser to filter the job profile and resume into different categories.

Categories: **Education background, working experience, skill set, requirement to job/job nature**

**Step 2**: **Semantic Analyzers**

Sub-Algorithm:

Word2vec – [Loading Data](https://deeplearning4j.org/word2vec#loading-data)

[Tokenizing the Data](https://deeplearning4j.org/word2vec#tokenizing-the-data)

[Training the Model](https://deeplearning4j.org/word2vec#training-the-model)

Evaluating the Model, Using Word2vec

O/P:

Double cosSim = vec.similarity("Programmer", "Software Enginner");

//output: 0.7704452276229858

**Step 3: Vector skill analyzer using Word2Vec**

Use Step 1 algorithm to find the similarity in Skills

Find the average of probability in order to get the probability of multiple skills

**Step 4: Converting category into numeric data job skill and job desc**

|  |  |  |
| --- | --- | --- |
| **Category** | **Numeric data** | **Job desc** |
| **Education level** | **4** | **5** |
| **Education** | **1** | **0** |
| **Industry Experience** | **0** | **0** |
| **Position Name** | **0.87** | **0.87** |
| **Experience** | **60** | **70** |
| **Skill set** | **0.71** | **0.71** |
| **Salary range** | **0.5** | **0.3** |
| **Work location** | **200** | **200** |
| **Work time** | **1** | **0** |

Use Euclidean distance or cosine similarity to find the probability score

Used for recommendation of engine.

**Step 5: Using numeric data building a supervised model**

Use Logistic regression or Decision tree to whether candidate will be hired or not.