Entity Extraction From Job Ads, CV and Skill Gap Analysis

|  |  |  |
| --- | --- | --- |
| *Naman Tyagi (2018055)* | *Sanchit Trivedi (2018091)* | *Shashwat Aggarwal (2018097)* |

*Indraprastha Institute of Information Technology, Delhi*

**Abstract**

*Given a job advertisement, we often would like to know whether or not we are a fit candidate for the given job ad. Getting useful insight into the required skills for the job is important to improve chances of a candidate getting selected for the job. For this project, we are limited to software jobs only. This requires semantic information about jobs such as MySQL and Database are closely related; however, Java and JavaScript are entirely different languages. Furthermore, even more, important than hard skills are the soft skills required for any job. For this project, three soft skills have been shortlisted - leadership, teamwork and problem-solving. These broadly give us three different categories of soft skills that might be present in the job advertisement. Lastly, the candidate can view the skills that they are missing concerning those required in the job advertisement. We have used Skill2Vec, which is a pre-trained neural network architecture and implemented it further to extract the required entities based on custom phrase matching and skill gap analysis. Our model reported almost 70% accuracy for soft skills extraction and 63.42% for skill gaps. [*[*GitHub*](https://github.com/shashwataggarwal/entity-extraction-ip)*] [*[*Demo*](http://bit.ly/entity-extraction-ip)*]*

**Introduction**

With leaps in technology, humans rely on the internet for most of their information through sources, such as websites, social media and web portals. This advancement in internet technology has also had an impact on recruiting potential employees for an organisation and on candidates looking for a job. Online Job Portals provide a lot of options but with the increased competition since this information is publicly available. In this paper, we aim to identify the skill gaps between a Job description and a prospective candidate. It allows the candidate to apply for Jobs which matches the most with his/her skillset. Also, It will enable companies to filter out candidates from the multitude of applications it receives.

To apply for any job, every candidate already has a CV/Resume which can provide the details needed to match a candidate to a given job profile. Comparing the two, we can provide the candidate with an insight into what they might need to improve on or add to their skill set to enhance their chances of getting hired. In the end, the candidate can see if they are fit for a given job via the ad they present or given their CV, what jobs are suitable for them. Certain positions also require a set of soft skills to perform better in a given role. Through our system IP, we aim to identify the skill gaps between a candidate’s skill set and job requirements. To provide proof of the model and reduce the dataset, we are focusing only on software development jobs.

We have labelled the job descriptions with three soft skills, namely leadership, teamwork and problem solving that is required for the job. Each job ad can have multiple labels. Leadership roles need candidates to lead teams or work independently with high onus. Teamwork requires candidates to work in teams and groups. We came across a lot of jobs requiring users to work in scrum groups. Finally, problem-solving requires candidates to be sharp at solving challenging problems and come up with inventive solutions.

**Dataset**

* **Job Advertisements**

Three hundred job advertisements were scraped from www.glassdoor.com™ using Software Developer as the search keyword using Selenium and BeautifulSoup in Python. The results were stored in a CSV file. The Company Name, Job Title, Job location, Job Description and Remuneration were extracted. The job description contains the entire job description as a string.

* **Resumes**

A pre-prepared, publicly available dataset was used from Kaggle.com ([link](https://www.kaggle.com/avanisiddhapura27/resume-dataset)) which contains the skills for each candidate.

**Methodology**



* **Skillset Phrase Matcher**

We approached skill extraction by first defining the technical skills. This was done by initialising a dictionary of skills containing 8000 significant skills in the software and technical domain.

A phrase matcher processed these skills, and a spacy model was built on these shortlisted skills. A similar process was carried out for soft skills as well.

* **Preprocessing and Extracting Skills from Job Description**

The job description is in an inconsistent format. Some contain a company description, and some don’t. Furthermore, there is no clear demarcation between the various sections of the job description. This job description was pre-processed to remove stopwords and then lemmatised using the Spacy library. The skills required for the job description were extracted using the skill phrase matcher by parsing the preprocessed job description.

* **Skill2Vec Model**

The Skill2Vec model is an unsupervised pre-trained model which was trained using Word2Vec. This model was trained on a large corpus of 1.4 million job descriptions. This model gives the cosine similarity between 2 words which is the metric that we will use to find thresholds for various further steps. The Skill2Vec model has reported an accuracy of 78% on its dataset. The model can be further improved for the new corpus but requires various preprocessing formats from Skill2Vec which are unavailable.

* **Skill Transform**

We implemented a skill\_transform function which was inspired by Skill2Vec. This is a helper function for the model to incorporate patterns not recognised by the word embeddings. This helps solve inconsistencies in the model by encoding various similarities and differences in the meaning of multiple skills. E.g., OOPs and Object-Oriented Programming are the same.

* **Gap analysis**

The Gap Analysis is performed by converting the list of the candidate’s skill into a feature vector and measuring the cosine similarity between the feature vectors of the candidate and each required skill in the job description. If the score returned is greater than the supplied threshold, the skill is considered to be present in the candidate’s resume. Otherwise, the skill is considered to be absent and reported as a skill gap between the candidate and the job description.

* **Soft Skills**

Each Job description is assigned a multi-class label out of three soft skills viz. a. Viz. Leadership, Problem Solving and Teamwork. The label is assigned by calculating the cosine similarity between the feature vectors of the words similar to the aforementioned soft skills and the Job description. If the score is greater than a threshold value, the job description is assigned a label of the soft skill. The threshold values are decided using Binary Search.

**Results**

To test our model we manually annotated the prepared dataset and assigned multi-class labels of Leadership, Teamwork and Problem Solving. This was then compared with the predictions from our model. The accuracy was calculated using the proportion of the predicted correct labels to the total number of labels for that data point averaged across the number of data points. This value was found to be 69.23%. The precision and recall values for Soft Skills was 73.44% and 64.10% respectively.

For validating the skill gap analysis produced by our model we again manually annotated the dataset with the expected skills gaps. The gaps are annotated as the skills which the candidate does not satisfy from the job skills. The model reported accuracy of 63.42%, precision 67.27% and recall 77.72%

Clearly, there is a scope for improvement. Our model was based on primary knowledge of Word2Vec and NLP methods with which we are able to achieve decent real-world results. However, with more work, this model can be improved to give higher accuracy.

**Limitations**

Technology is constantly evolving and different times demand different skills. Since we are using a dictionary to extract relevant skills the dictionary needs to be constantly updated manually to accommodate the advent of new skills required by the industry.

The model can only classify a job description into three soft skills, whereas there are many more soft skills. Furthermore, by retraining the model on the additional dataset, we can improve the word embeddings for the model. This will help incorporate new skills and trends in the industry. Converting the model into a search and retrieval model based on a score ranking method can prove to be a better approach.

**Future Scope**

The current model uses a dictionary to extract skills from a Job description and is therefore limited by the skills in the Dictionary. This problem can be solved by developing a custom Named Entity Recognition model instead of a dictionary to extract the relevant skills and thereby expanding the scope of the model from tech jobs to all sorts of jobs. The current model is restricted to 3 soft skills to provide a proof of concept. Increasing the scope of the model from 3 soft skills to all the soft skills required in any industry.

**References**

1. Van-Duyet, L., Quan, V. M., & An, D. Q. (2017). *Skill2vec: Machine learning approach for determining the relevant skills from job description.*
2. Honnibal, M., & Montani, I. (2017). *spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing*.
3. Ahmed Awan, M. N., Khan, S., Latif, K., & Khattak, A. M. (2019). *A new approach to information extraction in user-centric E-recruitment systems*. Applied Sciences (Basel, Switzerland), 9(14), 2852.
4. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space.*
5. Kalva, T. R. (n.d.). *Skill finder: ASkill finder: Automated job-resume matching system automated job-resume matching system.*