Towards A Skills Taxonomy

Mohsen Sayyadiharikandeh
School of Informatics and Computing
Indiana University
Bloomington, Indiana 47405
msayyadi@indiana.edu

Raquel L. Hill
School of Informatics and Computing
Indiana University
Bloomington, Indiana 47405
ralhill@indiana.edu

ABSTRACT

When evaluating job applications, recruiters and employers try to determine whether the information that is provided by a job seeker is accurate and whether it describes an individual that possesses sufficient skills. These questions are related to the hidden skills and skills resolution problems. In this paper, we argue that using a skills taxonomy to identify and resolve unknown relationships between text that describe an applicant and job descriptions is the best way for addressing these problems. Unfortunately, no comprehensive, publicly available taxonomy exists. To this end, this work proposes an automated process for creating a skills taxonomy.

Effective and efficient methods for bootstrapping a taxonomy are critical to any process that names and characterizes the properties and interrelationships of entities. To this end, we present three potential methods for bootstrapping and extending our skills taxonomy. We propose a hybrid scheme that combines the beneficial features of those methods. Our hybrid approach seeds the bootstrapping process with publicly available resources and identifies new skill terms and corresponding entity relationships. In this paper, we focus specifically on using Wikipedia as our corpus and exploiting its structure to populate the taxonomy. We begin by constructing a relationship graph of possible skill terms from Wikipedia. We then use a data mining methodology to identify skill terms. Our results are promising, and we are able to achieve a 98% classification rate.

Keywords

skill, ontology, bootstrapping, job application

1. INTRODUCTION

Research data show that recruiters and employers screen applicants based on age, class, personality, and other characteristics that may not be gleaned from a resume. In an ideal situation, one might imagine that employers hire the most skilled applicant, but sociological research shows this is not the case [15, 12, 19, 21, 17, 8]. On the other hand, research also shows that job applicants lack the tools to express and present their skills. In addition, constant changes in technology limit recruiters ability to specify a skill set that meets job requirements [7]. Both issues contribute to the hidden skills problem, which concerns representing, identifying and measuring skills that haven’t been explicitly articulated in the job description or resume. For example a job description may state that the applicant should know PHP, and an applicant’s resume lists Symphony as a skill. Symphony is a PHP web application framework, but the person that is evaluating the resume may not be able to recognize the relationship between PHP and Symphony. Therefore the applicant has a hidden skill, PHP, which has not been explicitly articulated by the applicant, and may not be detected by the evaluator. In addition to the hidden skills problem, there is the skills resolution problem. The skills resolution problem involves using online social media and online blogs to identify and confirm the skills of potential applicants. Software like hiringSolved [14] scrapes social media to build a potential skill cloud for applicants. The effectiveness of such software depends in part on the software’s ability to map text taken from social media to specific skills.

This research seeks to address the hidden skills and skills resolution problems by proposing a framework for developing a skills taxonomy that specifies a hierarchy of job skills and relationships between the specified skills. The proposed framework uses the web as its language corpus, and uses a hybrid model to extract relationship information from the corpus. It extends prior work by using the structure of web text to define rules for extracting information and to specify the relationships between skill terms. We present three potential methods for bootstrapping and extending our skills taxonomy and propose a hybrid method that combines the beneficial features of those methods. Our hybrid approach seeds the bootstrapping process with publicly available resources. In this paper, we focus specifically on using Wikipedia as our corpus and exploiting its structure to populate the taxonomy. We begin by constructing a relationship graph of possible skill terms related to compressing domain from Wikipedia. We then use a data mining methodology to identify skill terms. Experiments show promising results, and we are able to achieve a 98% classification rate.

The remainder of this paper is organized as follows. In section 2 we explain prior efforts for creating a skills taxon-
ony. Section 3 describes a motivating example of the skills resolution problem and how a skill taxonomy can be used to resolve a job seeker’s skills from online data. Section 4 describes three different approaches for bootstrapping a taxonomy, and our hybrid approach. We also present the results of our initial experiments. We conclude the paper and discuss future work in Section 5.

2. RELATED WORKS

The problem of creating a dataset or list of skills has been tackled before [4, 2, 10], but the results cannot be accessed freely for scientific projects. In addition, these datasets may not specify the relationship between skill terms, thereby limiting their use for solving the hidden skills and skills resolution problems.

It’s yours skills is a project that gets updated regularly, and its aim is to cover skills across industries and functions [4]. The project website offers an API for the developers but it is not freely available. Skill-project’s objective is to build a comprehensive and organized database of all human skills [2]. The developers claim to have built the largest, multilingual skills database by contribution of a skillful community. However, there is no API available for developers to benefit from this project, and the manual process for generating the lists is time consuming.

Bastian et al proposed a large-scale topic extraction and inference method for inferring skills from the LinkedIn profile of users [10]. They assume that “Skills” are a data-driven feature on LinkedIn, which means that the LinkedIn recommender engine can use the data of the users and propose skills for them. However, LinkedIn also allows members to specify their skills and endorse (verify) the skills of others. LinkedIn offers an API to developers but skills are not part of the attributes of basic profile data [1] and cannot be scraped. Though there are instances of skill data sets, as previously mentioned, they are not publicly available or they do not contain comprehensive relationship information.

3. MOTIVATING EXAMPLE: SKILLS RESOLUTION PROBLEM

In this section we present a simple example of the skill resolution problem and discuss how a skill taxonomy can be used to resolve a job seeker’s skills from online data. Section 4 describes three different approaches for bootstrapping a taxonomy, and our hybrid approach. We also present the results of our initial experiments. We conclude the paper and discuss future work in Section 5.

Figure 1: Keyword Cloud using posts authored by user

Figure 2: Tag Cloud using frequent tags in posts authored by user

Figure 3: Skill Cloud using tags from users’ post in Stack Overflow that also appeared in the skills list. Figure 3 shows that the list of irrelevant words are filtered out.

However, the weakness of this method is that it discards possibly relevant words that have no exact match in the skills data set. Such terms may allow us to infer additional skills by further analysis. For example, tags like 'Array' or 'List' which indicate that the user has some knowledge of 'data structures', can be used infer such knowledge. In order for an automatic agent to be able to do this inference task we would need to have a robust taxonomy that contains different entities with embedded relationships between them. Contextual domain information may be used to ease the design of such a taxonomy, but populating the instances for each entity would be time consuming task. In this paper we explore three potential approaches for bootstrapping a skill taxonomy from a initial set of skills. We also propose a
4. METHODOLOGY

Based on prior works we propose to develop a skills taxonomy and a method capable of automatically bootstrapping the taxonomy. Bootstrapping is a process for learning relationship rules that alternates between learning rules or rule accuracy from sets of instances of included entities and finding instances using sets of rules. We studied different approaches for determining relationships between skills, and discovered several that in combination could help with creating and bootstrapping our skills taxonomy. We present these works below and propose a hybrid approach that leverages the benefits of these prior works.

4.1 Word2Vec

Word2Vec is a tool that creates a language model that enables a user to measure the semantic similarity between a queried word and words found in the input corpus [18]. The resulting language model contains continuous vector representations of words, and these vectors can be used to measure word similarity. This is a nice feature which can be employed in our project for bootstrapping an initial data set of skills. Another useful feature of this tool is that you can use context to filter out irrelevant terms.

Word2Vec can use a corpus like Wikipedia to create a language model, or any text corpus can be used to train a language model. We conducted preliminary experiments using both Wikipedia(text8 corpus contains 100 megabytes of wikipedia content) and our own corpus that contained job descriptions scraped from famous job advertisement websites. Initial experiments using Word2Vec produced some noisy results that contained terms that did not fit our context. For example, our experiment assumed a computer science, programming languages context. When querying python with Wikipedia as our corpus, we get a list of words containing monty, moby, dracula, and circus. Table 1 shows the results of the top 10 similar words for 'python'.

We also created our own corpus using job descriptions collected by using the Bing search engine API [6] to scrape Indeed.com website. Table 2 shows the results. While the terms seem to be more related to computer science, all terms cannot be considered as skills. To increase the overall accuracy of these results, we propose to use the structure of web text, such as Wikipedia articles, to specify semantic relationships between terms, thereby enabling us to filter irrelevant terms.

4.2 Exploiting Wikipedia Structure

The research community in the domain of information retrieval and semantic web often rely on Wikipedia as an external source of knowledge. Additionally, researchers use the content and the embedded structure of the articles to enhance the performance of tasks like query expansion [9], named entity recognition [16, 20]. Mining the wiki can enhance our bootstrapping process by providing a hierarchy of semantic relationships between terms.

We think that Wikipedia is a convenient knowledge source, because it is a large corpus containing millions of articles about named entities. Wikipedia is perhaps the biggest knowledge base in computing that can be mined in a more structured way than search engine results. In addition, it has more coverage than well-known ontologies like word net [22]. Our interest is to analyze the content of Wikipedia and its internal structure in order to bootstrap our skills taxonomy. We propose to use the Wikipedia API [5], a web service that provides convenient access to wiki features, data, and meta-data. The Wikipedia API enables us to retrieve the corresponding Wikipedia entry for each candidate word and extract categories of the associated article or the content of...
it. For instance, we use the first sentence of the entry (using simple lexico-syntactic pattern) to build a hierarchy of concepts and identify the relationship between terms. We use this relationship information to create a graph of terms with the links within the graph indicating a relationship between two adjacent terms.

Relationship between terms in Wikipedia can be derived in many other ways, including using html hyperlinks within articles as indicators of relationships. This approach produces a lot of noise in the graph. This noise should be filtered out because links can have different semantic meanings far beyond the expected semantic relationship in our taxonomy. In addition to links, the Wikipedia "category" tag may be used to indicate a relationship between terms. Using the category tag may still produce a large graph in terms of branching factor and multiple inheritance relations, which creates obstacles in searching and mining problems [22]. In our project we decided to use this tag for creating graph of wikipedia.

We used the Wikipedia API to start scraping from one article with the title 'Python (programming language)'. Using category relationships, we scraped parent categories and their subcategories of each category in the path with the length equal to 2 from starter node (python article). The resulted undirected graph had 1359 nodes and 1581 edges. Two human annotators manually annotated the nodes in the graph, identifying each node with a binary value of either 1 or 0 with 1 indicating a new skill term. We used the annotated data set as a training set for machine learning algorithms that would classify terms as related or not related to IT skills. Approximately 80 percent of the nodes are not related to skills and 20 percent have some information related to skills. We used Gephi to visualize the graph and used Force Atlas algorithm to change the layout of the graph. Figure 4 shows the resulting Python-related graph. Nodes shown in green were labeled as skills.

![Figure 4: Resulted Wikipedia Graph starting by python article](image)

As we can see most of the desired nodes exist in the dense sub-graph. We defined some structure based features, and fit some models using different feature sets. So for this binary classification problem we used logistic regression, and decision tree to do the initial experiments. Note that we started by some minimal features (degree, avg degree of neighbors and exist-in-k-core) and incrementally added other features until the point we got satisfying results. We used cross validation with 10 folds and randomly chose 80 percent of the data set for the training set and 20 percent for the test set. As common in machine learning we found the fit model on training set and tested on the test set. Since 80 percent of the dataset has one label the maximum likelihood estimator (MLE) can predict with 80 percent accuracy by predicting every node to non-relevant. However, we consider MLE as a baseline and try to compare other algorithm to it. Table 4 shows accuracy, recall and F1-score of predictions. We can reframe the problem here as a retrieval task. So far we have indeed treated the problem as a binary classification task, assuming that ‘being related to IT skill’ was the ‘True’ label, and ‘not related to IT skills’ the ‘False’ one. We can instead consider each as the set of “relevant” examples, and compute accuracy and recall scores accordingly. Accuracy shows the performance of classifier in average meaning that it calculates fraction of right predictions. Precision is the fraction of positive predicted samples that are truly related to IT skills over number of positive predictions.

\[
\text{Precision} = \frac{\text{relevant samples} \cap \text{Retrieved samples}}{\text{Retrieved samples}}
\]

Recall shows the fraction of the samples that are relevant to IT skills and are successfully retrieved.

\[
\text{Recall} = \frac{\text{relevant samples} \cap \text{Retrieved samples}}{\text{Relevant samples}}
\]

F1-score is defined as following:

\[
F1 - \text{score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]

Weights of features in the fit model by logistic regression and most important decision tree rules shows that noOfSkillsInNeighbour is the most important feature for prediction task.

The proposed automatic method for using Wikipedia’s structure to identify relevant skills is scalable, and more fiscally efficient than prior approaches that used Amazon’s Mechanical Turks to find the associated articles [10]. However,
We propose a machine learning based method in this level which learns through experimental exploration. Again we propose a hybrid approach to build a skills taxonomy. The best way to combine the outputs of three methods can be learned through experimental exploration. As we extend our relationship graph by searching for additional terms within Wikipedia or other related corpuses on the web, more noise cannot be considered as skills.

4.3 Co-occurrence on web

Learning an ontology by web mining has been extensively used by researchers [23, 13] Sanches and Moreno proposed a methodology to build automatically an ontology, extracting information from the World Wide Web by the help of search engine from an initial keyword. Their method relies extensively on a publicly available search engine and extracts concepts (based on its relation to the initial one and statistical data about appearance) [23].

KnowItAll [11] is an information extraction system that can extract large numbers of high-quality instances of classes (for example instances of the class City) or relations (like instances of capitalOf(City, Country)). It is seeded with an extensible ontology and a few generic rule templates from which it creates text extraction rules for each class and relation in its ontology. It is a Web-based information extraction system with a novel extract and assess architecture that uses generic lexico-syntactic patterns and Web-scale statistics to extract class and relation instances from unstructured text on the Web. KNOWITALL evaluates the information it extracts using statistics computed by treating the web as a large corpus of text (no structure).

The common components of mentioned information retrieval systems is that they use search engine using some initial terms and try to measure the co-occurrence of other terms in the search result with the seed terms. We employ these ideas for learning a skills taxonomy. We can use search engine operators to focus on certain types of websites that post job advertisements, like Indeed.com and Monster.com. In other words, we can exploit the co-occurrence values of terms in job descriptions for bootstrapping purposes.

4.4 Hybrid approach

Knowing the strengths and weaknesses of different approaches we decided to use a hybrid approach to build a more robust method for building a skills taxonomy. The best way to combine the outputs of three methods can be learned through experimental exploration. Again we propose a machine learning based method in this level which takes input features from other prior levels and outputs labels. During the first phase of our experimental exploration, we will run each method with the initial terms in our taxonomy and collect the results from each of them. Subsequently, we will define meta-features and feed them to the last layer classifier in order to do the final prediction task. Perhaps any method that we use to learn how to combine the outputs of the three previous methods is based on the assumption that a frequent term in a set of domain specific texts with close structure in Wikipedia indicates occurrence of a relevant concept in our taxonomy. The general framework that we propose is shown in Figure 5.

However, this is the macroscopic view of the hybrid method and details of how features from previous layers can collaboratively help a classifier to learn new skills can be understood by further experiments.

5. CONCLUSIONS

Effective and efficient methods for bootstrapping of a taxonomy are critical to any process that names and characterizes the properties and interrelationships of entities. We presented three potential methods for bootstrapping and extending our skills taxonomy and propose a hybrid method that combines the beneficial features of those methods. Our hybrid approach seeds the bootstrapping process with publicly available resources. In this paper, we focus specifically on using Wikipedia as our corpus and exploiting its structure to populate the taxonomy. We begin by constructing a relationship graph of possible skill terms from Wikipedia. We identified a set of features in order to characterize the nodes in the Wikipedia graph. We were able to achieve 98% accuracy with decision tree in finding the related nodes within the graph that originated with one term. These results are promising, but limited. As we extend our relationship graph by searching for additional terms within Wikipedia or other related corpuses on the web, more noise is introduced. We are hopeful that our fully implemented hybrid model will eliminate irrelevant terms from the graph and produce accurate results.

6. ACKNOWLEDGMENTS

This work is funded by the National Science Foundation: 1537768

7. REFERENCES

H. Zargayouna, and O. R. Terrades, editors, 

[10] M. Bastian, M. Hayes, W. Vaughan, S. Shah, 
P. Skomoroch, H. Kim, S. Uryasev, and C. Lloyd. Linkedin 
skills: Large-scale topic extraction and inference. In 
Proceedings of the 8th ACM Conference on Recommender 
Systems, RecSys ’14, pages 1–8, New York, NY, USA, 
2014. ACM.

Popescu, T. Shaked, S. Soderland, D. S. Weld, and 
A. Yates. Web-scale information extraction in knowitall: 
(preliminary results). In Proceedings of the 13th 
International Conference on World Wide Web, WWW ’04, 
pages 100–110, New York, NY, USA, 2004. ACM.

Nottingham, and D. C. R. Council. Half a chance? : a 
report on job discrimination against young blacks in 
Nottingham. London : Commission for Racial Equality in 
association with Nottingham and District Community 
Young coloured persons. Employment. Racial 
discrimination (BNB/PRECIS).

Ontology learning from domain specific web documents. 
2009.


and white-collar workers in britain. Race Class, 

external knowledge for named entity recognition. In Joint 
Conference on Empirical Methods in Natural Language 
Processing and Computational Natural Language Learning, 

labor practices in a networked age. October 2014.

estimation of word representations in vector space. CoRR, 
abs/1301.3781, 2013.

in the labour market. Cambridge Journal of Economics, 

multilingual named entity recognition,â¬ã aclâ¬ã08, 2008.

professional service firms. American Sociological Review, 

semantic relatedness using wikipedia. In Proceedings of the 
21st National Conference on Artificial Intelligence - 

[23] D. SÀANEZCHEZ and A. MORENO. Creating ontologies 
from web documents. In Recent Advances in Artificial 
Intelligence Research and Development. IOS Press, 113, 
pp.11-18.