Syntax-based Skill Extractor for Job Advertisements

Ellery Smith, Martin Braschler, Andreas Weiler
Institute of Applied Information Technology
Zurich University of Applied Sciences
Winterthur, Switzerland
{smil, bram, wele} @zhaw.ch

Thomas Haberthuer
Skillue AG
Basel, Switzerland
thomas.haberthuer@skillue.com

***Abstract*—In the context of extracting relevant skill-terms from job advertisements, we propose a syntax-based method for generating large amounts of machine-labelled text from a small amount of human-labelled data. This is then used to solve the vocabulary problem and significantly increase recall when detecting skills.**

***Index Terms*—document handling, syntactical parsing, information extraction, online recruitment**

I. Introduction and Goal

The extraction of skills from job advertisements is an interesting, but challenging problem. Many employment-oriented companies like LinkedIn, Monster, or Indeed invest a lot of time and manpower in solving the problem. However, most of the previous work (e.g., [1]–[3]) tries to tackle the problem by extracting entities with the support of large ontologies and taxonomies in the corpus of job advertisements. We argue that we are able to extract meaningful skills from job advertisments by analysing the syntactic patterns of the textual content.

II. Approach

A. The Vocabulary Problem

When a new job is encountered by a skill extractor, it may contain many terms that have never been encountered during training. For instance, there may have been jobs about *Chemical Biology* or *Molecular Biology* in the training set, but none concerning *Genetic Biology*. Suppose we have the two sentences, where red terms are not present in the training set, and the desired skills are underlined:

1) A keen interest in workplace management is essential.
2) A good knowledge of genetic biology is essential.

Both *keen* and *genetic* have never been encountered before, and thus cannot be distinguished in a term-based model. One approach to this problem is to use semantic information; however, such systems are often computationally expensive and encounter vocabulary problems of their own.

B. Syntactic Patterns

By looking at a set of jobs with manually-labelled skills, we can observe some patterns in the distribution of syntactic dependencies. When we look at dependencies between pairs of skill terms (for instance, in the phrase '*English* or *German* is required.'), only a small amount of dependency types occur, as shown in Figure 1. However, when looking at the dependencies of non-skill terms (cf. Figure 1 (right)), the distribution reflects that of normal language. It is clear that a very strong pattern exists in the grammar of skill terms, likely due to the homogeneous language style adopted when writing a job advertisement.

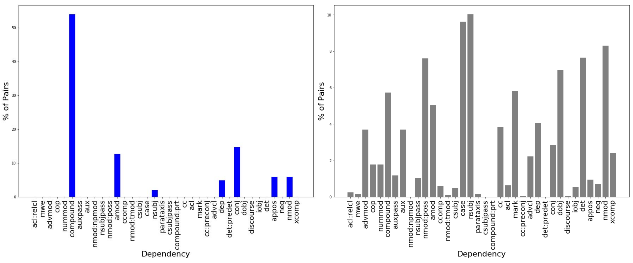
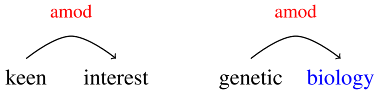


Fig. 1: Syntactic Dependencies from skill term to skill term (left) and non-skill terms (right).

In the example above, with *keen* and *genetic*, the term *biology* is a known skill and the term *interest* is a known non-skill term. However, both pairs of terms are syntactically identical, and are related by the 'adjectival modifier' (amod) dependency:



Based on the information in Figure 1, we can construct a rule that will mark *genetic biology* as a skill, despite never encountering the word *genetic* before. We can also discard the term *keen* as a non-skill term with the same approach.

C. A Baseline Classifier

However, in order to detect additional skill terms in this manner, we must first identify *some* skill terms in a job ad. And since the accuracy of the syntax-based method is reliant on these initial terms, a high precision is required. We define a very simple confidence measure, τ, as follows:

$$\tau = \frac{\text{No. of times a term is a skill}}{\text{No. of occurrences}}$$

In Figure 2 we plot the proportion of terms at each confidence score. There are very few ambiguous terms in the dataset, and by simply using $\tau \geq 0.5$ as a classifier, we

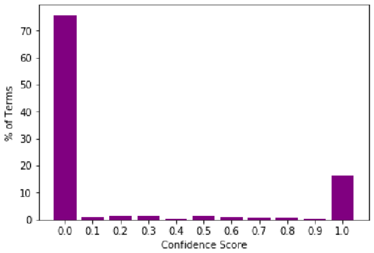
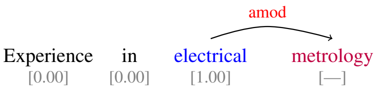


Fig. 2: Distribution of confidence scores (τ) across all terms.

achieve 80% precision. However, due to the vocabulary issues explained above, this method has very low recall (49%). This is an appropriate starting point for using the syntax-based methods to increase recall.

D. Applying Syntactic Rules

A worked example from a job ad for a Swiss communication company proceeds as follows:



Since the term *electrical* scores 1.0, and the previously unseen term *metrology* has the term *electrical* as an adjectival modifier (amod), we extract the far more relevant skill *electrical metrology* from this sentence.

After using this syntactic method to find new skills, we can use these new skill terms to recursively find further skills, by essentially walking the dependency graph. In an example sentence (cf. Figure 3) from an unlabelled job ad, only the terms *mechanical* and *prototypes* are known from training. With our approach we are able to extract the skills: *mechanical design*, *protoypes* and *3D Printing*.

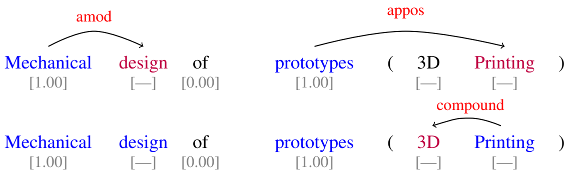


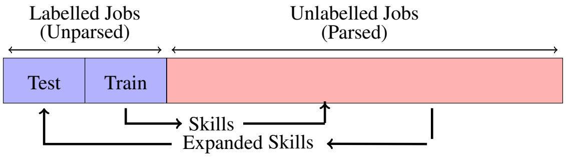
Fig. 3: Example of skill extraction of an unlabelled job advertisement.

E. Implementation and Evaluation

This method can be used on a single job ad to expand its set of skill terms – however, this requires a syntactic parse of every new incoming job. This may be computationally expensive, so we present a method which does not require a new job ad to be parsed.

Suppose we have a small sample of manually labelled jobs, J_L, and large set of unlabelled jobs, J_U. We can automatically label each job in J_U using the method described in Figure 3. This new labelling can then be fed into the baseline algorithm to produce a new set of confidence measures. This process can then be applied repeatedly on J_U until no new terms can be labelled.

Thus, we can use the same algorithm as the baseline method, but with significantly more synthetically generated data. To conduct an evaluation of our approach, we test the system in the following manner:



We use a set of 100 manually labelled job advertisements containing 27055 terms, 3045 of which are skills. These are split into a training portion (representing J_L), and an evaluation portion. These are used with a set of 10,000 unlabelled jobs (J_U). The results in comparison to the baseline are show below. There is a significant improvement in recall, and only a minor drop in precision, while maintaining the same computational efficiency as the baseline method.

	Precision	**Recall**
Baseline	80%	49%
With Syntax	77%	70%

F. Generating a Synthetic Dataset

When using the same 100 labelled jobs to generate labels for 10,000 unlabelled jobs, we can see in the table below that the vocabulary expands significantly, while using only a small amount of initial data – and the results shown above demonstrate that these additional labels are largely accurate.

	Jobs	**Skills**	**Non-Skills**	**Skills per Job**
Manually Labelled	100	3045	24010	30.45
Automatically Labelled	10,000	340656	3090668	34.06

III. Conclusions

In this work, we have shown that it is possible to expand a small amount of human-labelled data into a large amount of comparatively accurate machine-labelled data, by identifying linguistic patterns. The evaluation shows that our approach significantly improves recall, at minimal cost to precision.

Acknowledgment

The work was supported by the Swiss Commission for Technology and Innovation under grant 27177.2 PFES-ES.

References

[1] I. Kivimäki *et al.*, "A Graph-Based Approach to Skill Extraction from Text," Proc. of TextGraphs-8 Graph-based Methods for NLP, pp. 79–87, 2013.

[2] B. Mathieu *et al.*, "LinkedIn Skills: Large-scale Topic Extraction and Inference," Proc. of the 8th ACM Conf. on Recommender Systems, pp. 1–8, 2014.

[3] D. Çelik *et al.*, "Towards an Information Extraction System Based on Ontology to Match Resumes and Jobs," Proc. of the 37th IEEE Annual Computer Software and Applications Conf., pp. 333-338, 2013.