Bringing Order to the Job Market: Efficient Job Offer Categorization in E-Recruitment

Emmanuel Malherbe
Multiposting
Paris, France
emalherbe@multiposting.fr

Mario Cataldi
LIASD
Université Paris 8
Paris, France
m.cataldi@univ-paris8.fr

Andrea Ballatore
Center for Spatial Studies
University of California, Santa Barbara
Santa Barbara, CA, USA
aballatore@spatial.ucsb.edu

ABSTRACT

E-recruitment uses a range of web-based technologies to find, evaluate, and hire new personnel for organizations. A crucial challenge in this arena lies in the categorization of job offers: candidates and operators often explore and analyze large numbers of offers and profiles through a set of job categories. To date, recruitment organizations define job categories top-down, relying on standardized vocabularies that often fail to capture new skills and requirements that emerge from dynamic labor markets. In order to support e-recruitment, this paper presents a dynamic, bottom-up method to automatically enrich and revise job categories. The method detects novel, highly characterizing terms in a corpus of job offers, leading to a more effective categorization, and is evaluated on real-world data by Multiposting (http://www.multiposting.fr/en), a large French e-recruitment firm.

Keywords

E-Recruitment, Job Categorization, Classification

1. INTRODUCTION AND MOTIVATION

E-recruitment can be defined as the process of matching people to appropriate jobs using different on- and off-line strategies and technologies. As the number of candidates and jobs varies by expanding, shifting, and contracting, the recruitment process presents remarkable challenges in information management, indexing, and matching. The usage of the web helps employers reach larger population of candidates, lowering geographic barriers [10]. Several companies currently handle large volumes of job offers and applications across the globe, including Multiposting (multiposting.fr), LinkedIn (linkedin.com), and Monster (monster.com).

Conventional recruitment methods deployed in Human Resource Management (HRM) are notorious for being costly and time-consuming, particularly when applied to large pools of offers and candidates. Since the 1970s, job categorization has been used in human resources departments for personnel selection and promotion, as well as for salary administration [2], and is currently recognized as a major issue in recruitment [1, 4]. To tackle the complexity of the matching process, a common approach relies on the categorization of job offers, identifying and grouping salient characteristics, tasks, and skills. Top-down categorizations encode domain knowledge, and are commonly used to explore, search, and analyze data. While fully bottom-up, keyword-based techniques can be applied to e-recruitment, curated and human-readable categorizations are necessary to obtain high-quality results, reducing the noise in the matching.

Existing categorization strategies fail to identify the terms that capture salient aspects in the descriptions of potential candidates. Similarly, job descriptions contain vague and overly generic information, which is hard to exploit in the matching process. Moreover, the vocabulary used by employers and recruiters change over time, reflecting the evolution of the labor market. For this reason, the detection of these variations is key to design an effective job categorization strategy that reflects the underlying data more closely.

Let us consider for example a category descriptor related to computer programming. The 'programming languages' field might present a fixed list of keywords including popular languages (e.g., Java, C++, Python), without including recent languages such as Clojure or Swift that are likely to be in high demand in the immediate future. In this context, it is considerably hard, even for a domain expert, to update the descriptors, anticipating the rapid changes in the industry. To tackle the challenges of e-recruitment, top-down and bottom-up approaches need to be leveraged, updating expert-defined resources with dynamic knowledge extraction from the data.

In this paper, we propose a data-driven, bottom-up approach to enrich the textual descriptions of job categories. Given a domain categorization, the approach automatically updates the terms from the corpus of job offers. The enriched descriptions are then used to support a field-to-field similarity computation in the matching between jobs and candidate categories. The matching focuses on the differences between structured textual fields and the extracted terms, maximizing the probability of a correct classification. The proposed approach is evaluated in different application contexts on real-world data collected by Multiposting, one of the largest companies in the e-recruitment sector. The French company provides a web platform for e-recruitment, and processes more than 3M job offers per year and 100M applications in 50 countries.

SIGIR'15, August 09 - 13, 2015, Santiago, Chile.

DOI: http://dx.doi.org/10.1145/2766462.2776779.

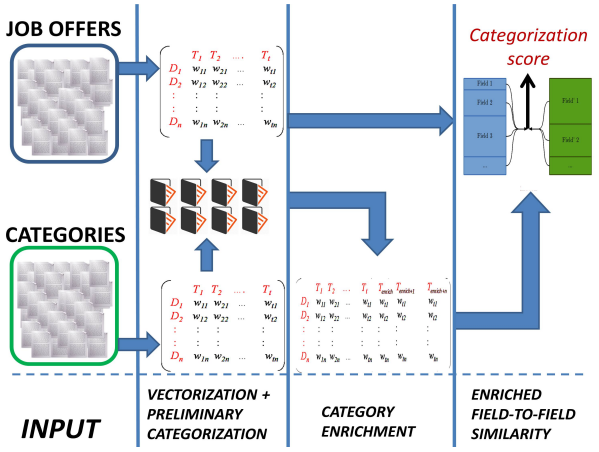

Figure 1: Flow diagram of the proposed hybrid job offer categorization method.

2. CONTEXT-BASED CATEGORY ENRICHMENT

The objective of this work is to support the categorization of a corpus of job offers by semantically enriching the job categories with recent and contextually relevant terms. The flowchart of the method is showed in Figure 1. In this study, we define two entities: job offers and job categories.

A *job offer* can be defined as a structured text document that formalizes an offer of employment, authored by an employer. A job offer contains textual fields, such as 'company description,' 'required skills,' 'education level,' depending on the specific conceptualization. Formally, a job offer o as a set of keyword vectors, each one representing a textual field $o = \{\vec{o_1}, ..., \vec{o_k}\}$ where k is the total number of textual fields in job offer o, and $\vec{o_i}$ (with $i < k$) is a vector of weighted keywords, which represents the frequency of terms of its i-th field. Each component $w_{k,i}$ of $\vec{o_i}$ is the term weight of the i-th vocabulary term in the k-th field of the considered job offer. This weight is calculated as:

$$w_{k,i} = \sqrt{tf_{k,i}} \left(1 + log\frac{|M|}{m_i}\right) \quad (1)$$

where $tf_{k,i}$ is the term frequency value of the i-th vocabulary term in k-th field. M is the entire set of job categories and m_i the number of job categories containing the i-th term in at least one field.

A *job category* is a description of a type of jobs provided by a domain expert, and has a structure analogous to that of job offers. We formalize a job category c as a set of vectors $c = \{\vec{c_1}, ..., \vec{c_l}\}$ where l is the number of textual fields in the job category description, and c_j a vector of weighted keywords that represent the frequency of terms of the j-th field of the job category (with $j < l$).

2.1 Enrichment of Job Categories

Given a corpus of job offers and job categories, we want to enrich a category c to a c' that includes new keywords to better reflect the job offers. As a first step, we retrieve the set of most representative categories for each job offer, quantifying the representativeness of categories through the cosine similarity of vector sums. Formally, for each offer $o \in O$, we compute the cosine similarity cos with category c as:

$$sim(o, c) = cos\left(\sum_{i=1}^{k} \vec{o_i}, \sum_{j=1}^{l} \vec{c_j}\right) \quad (2)$$

For each offer, we sort all categories in C in decreasing order in terms of similarity to $\vec{o}$. As the similarity values in this context follow a power law, we select a subset of categories so that the sum of their similarities sim reach 90% of the sum of all similarities $\sum_{j=1}^{l} sim(o, c_j)$, excluding the tail of categories with low similarities [8]. At the end of step, each job offer in O has a non-empty set of categories associated to it.

Using these sets, we search for the most contextually informative keywords for each category. The relevance weight $rw_{c,x}$ of a candidate keyword x for category c is computed as:

$$rw_{c,x} = log\left(\frac{\frac{r_x}{R_c - r_{c,x}}}{\frac{n_x - r_{c,x}}{N - n_x - R_c + r_{c,x}}}\right) \left|\frac{r_{c,x}}{R_c} - \frac{n_x - r_{c,x}}{N - R_c}\right| \quad (3)$$

where $r_{c,x}$ is the number of job offers associated to the category c containing the keyword x, and n_x is the total number of job offers containing x. R_c is the number of job offers associated to the category c, and N is the total number of job offers in the corpus. Intuitively, the keywords that appear frequently only in specific associations and rarely in others will tend to have higher weights.

Subsequently, for each category, we select the terms having a positive enrichment value as as a set of enriching keywords $K_{x \to c}$. This set is used to compute the enrichment vector to be added to the original job category vector c. The enrichment weight $e_{c,x}$ of the term x for category c is calculated as:

$$e_{c,x} = \sqrt{r_{c,x}} \left(1 + log\left(\frac{M}{m_x}\right) + rw_{c,x}\right) \quad (4)$$

where $rw_{c,x}$ is the relevance weight, $r_{c,x}$ is the number of job offers associated to the category. M the total number of job categories, and $m_x$is the number of job categories c containing the term x in its keywords set $K_{x \to c}$. The homogeneity of the function is assured by expressing $rw_{c,x}$ as a sum of probabilities while $log(\frac{M}{m_x})$ is assimilated to the probability of observing term x in a category [9].

Finally, we define the enriched job category as an extended set of vectors $c' = \{\vec{c_1}, \vec{c_2}, ..., \vec{c_l}, \vec{ev_c}\}$ where $\vec{ev_c}$ encodes the knowledge extracted from the corpus. The other vectors of the category remain unchanged.

2.2 Matching Job Offers and Job Categories

A job offer o can now be matched against the enriched job categories. For this purpose, we define a field-to-field similarity (FtFs) matching approach, taking into account both the information extracted from the corpus and the structure of the original data. First, we define a FtFs matrix as:

$$S(o, c') = \begin{Bmatrix} s(o,c')_{1,1} & s(o,c')_{1,2} & ... & s(o,c')_{1,l} \\ s(o,c')_{2,1} & s(o,c')_{2,2} & ... & s(o,c')_{2,l} \\ ... & ... & ... & .... \\ s(o,c')_{k-1,1} & s(o,c')_{k-1,2} & ... & s(o,c')_{k-1,l} \\ s(o,c')_{k,1} & s(o,c')_{k,2} & ... & s(o,c')_{k,l} \end{Bmatrix} \quad (5)$$

	Initial categories	Enriched categories	Offers
Documents	531	531	215,375
Terms	9,286	13,761	55,209
Fields	8	8	4

Table 1: The corpus of job offers and categories provided by Multiposting and Pôle Emploi.

Each element $s(o,c')_{i,j}$ represents the normalized similarity between the i-th field of the offer o (with $i < k$) and the j-th textual field of the enriched category c' (with $j < l$), that is:

$$s(o,c')_{i,j} = \frac{cos(\vec{o}_i, \vec{c}_j)}{\sum_{p=1}^{m}\sum_{q=1}^{l} cos(\vec{o}_p, \vec{c}_q)} \quad (6)$$

where *cos* denotes the cosine vector similarity. This similarity calculation leverages the differences between the fields. For instance, the offer of a programming job can have highly discriminant terms in the 'skills' field, but not in the 'employer description' field, which contains a set of general terms occurring in all offers by the same employer. Following this intuition, the similarity s between the offer o and the enriched category c' is therefore calculated as:

$$sim(o,c') = \sum_{i=1}^{k}\sum_{j=1}^{l} \lambda_{i,j}\ s(o,c')_{i,j} \quad (7)$$

where $\lambda_{i,j}$ is a similarity probability between textual field calculated based on a training data-set (see [5] for details). The higher $s(o,c)$, the more similar the offer o to the considered category c. The function has no upper bound, while the lower bound is 0.

At the end of this matching step, we have a matching score $s(o,c')$ for each enriched category c'. The last step consists of the selection of the most similar categories, obtaining the strength of association between job offer o and a set of enriched categories c'. This data-driven categorization embeds knowledge from the corpus of job offers, overcoming the limitations of the initial static categorization.

3. EVALUATION

The proposed categorization approach was evaluated through several experiments, using real job offers and job categories. For the evaluation, company Multiposting provided a corpus of 215,375 job offers, advertised on its online platform from 2009 to 2014. A set of 531 static job categories was obtained from Pôle Emploi, the French governmental agency that supports job seekers (www.pole-emploi.fr). The categories are part of the Répertoire Opérationnel des Métiers et des Emplois (ROME) nomenclature, which can be seen as the French equivalent of O*NET, a job categorization widely used in the US (www.onetcenter.org) [3].

A job category by Pôle Emploi is structured into 8 textual fields ('title,' 'definition, 'admission,' 'conditions,' 'environment' 'competences,' 'activities,' and 'labor mobility'), and is linked to manually defined similar categories. By contrast, the Multiposting job offers have only 4 textual fields: 'title,' 'job description,' 'candidate profile,' and 'company description.' Both offers and categories are in French. Table 1 summarizes the characteristics of these data-sets.

As ground truth to evaluate our approach, we produced an annotated set of 1,339 job offers, randomly selected from the corpus. Two teams of 3 domain experts performed a manual categorization, manually selecting the most fitting category for each job. When the experts disagreed about the categorization of a job, the conflict was discussed and resolved. In the instances where the conflicts were not easily settled, the job offers were discarded as too ambiguous.

To obtain a realistic picture, we applied four methods to the manually-generated data-set, comparing the performance of our approach with three alternative methods:

- Cosine Measure (*CM*): In this method, the corpus was categorized only with the cosine similarity measure in Equation 2. This approach does not enrich the categories and does not consider the field-to-field similarity of Section 2.2.
- Enriched Cosine Measure (*e-CM*): The categorization was performed by applying the standard cosine similarity measure (Equation2) to the enriched categories, but without using the field-to-field similarity.
- Field-to-Field Similarity (*FtFs*): This method categorizes the job offers only with the structural information (Section 2.2), without enriching the categories.
- Enriched Field-to-Field Similarity (*e-FtFs*): This is the proposed approach, which combines the previous methods.

Error rate (*E*). First, we observed the categorization errors with respect to the ground truth. The error rate E_1 is calculated as $1 - p$, where p is the categorization precision with respect to the top category selected by each of the four approaches. E_1 ranges from 0 to 1, where 0 means a perfect categorization, and 1 is an entirely wrong categorization.

Receiver Operating Characteristic (*AAC-ROC*). This measure evaluates binary classifiers by plotting two parameters: TP rate (fraction of True Positives) and FP rate (fraction of False Positives) [7]. To compare the methods, it is possible to reduce the ROC to a single scalar value by measuring the area above the curve (AAC), which quantifies the overall error.

Hinge-Loss (*HL*). The Hinge-Loss function computes the average distance between the proposed model and the valid data for multi-class categorizations[6]. For each job offer, HL considers the vector of similarity values of all categories generated by a method. This vector is then compared to the ground truth vector, in which the correct category is marked as 1, while the others are set to 0. The distance between a vector and the ground truth is inversely proportional to the quality of the categorization method.

As the top category is particularly important, we first observed the results for the top ranked category $argmax_c\ s(o,c)$ with indicators HL_1 and $AAC\text{-}ROC_1$. Subsequently, considering that the each category in the ROME nomenclature also points to similar categories, we tested if the candidate category proposed by each approach falls in this extended set of valid categories, with indicators E_{sim} and HL_{sim}.

As the results in Table 2 show, our Enriched Field-to-Field Similarity method (*e-FtFs*) outperforms the alternative methods with respect to all of the evaluation metrics. More specifically, *e-FtFs* obtained 0.25 for E_1, and 0.21 for E_{sim}. Considering the fact that job offers and categories were generated by different communities using different vocabularies, in different time ranges, we consider this error

	E_1	E_{sim}	HL_1	HL_{sim}	*AAC-ROC*
CM	0.38	0.33	0.797	0.722	0.234
e-CM	0.28	0.22	0.729	0.629	0.187
FtFs	0.28	0.24	0.673	0.484	0.168
e-FtFs	**0.25**	**0.21**	**0.645**	**0.472**	**0.152**

Table 2: Performance of the proposed approach (*e-FtFs*) compared with three alternatives, on a set of 1,339 job offers and 531 job categories.

rate promising. The HL indicators follow a similar trend, showing that the *e-FtFs* performs better than the other methods even when adopting a fuzzy categorization.

With respect the other indicators, HL shows higher error values. This is due to the fact that we compare the performance of a fuzzy classifier (the automatic method) with a binary one (the manual classification). As a result, the evaluation is very pessimistic: to obtain an error value equals to 0, the method should return a null vector with only the correct category set to 1. While both *e-CM* and *FtFs* outperform the simple cosine measure *CM*, the best performance is obtained only when combining the two methods into *e-Ftfs*, confirming the benefits of both the bottom-up category enrichment and the field-to-field similarity.

4. CONCLUSION

In this paper, we presented a novel categorization method targeted at industrial applications in e-recruitment. Our method semantically enriches the job categories, and then leverages the structure of textual documents using a field-to-field similarity comparison. The approach was evaluated on real-world data by the French corporation Multiposting. The experiments showed several improvements in the task of job categorization, providing a novel solution for weaving dynamic, bottom-up knowledge into static job categories.

The classification problem of job offers we tackled in this paper deserves further investigation. We plan to explore alternative families of categorization approaches, drawing on supervised and unsupervised machine learning techniques. While our current approach relies on a bag of words model, performance might be improved with more semantic knowledge, for example clustering synonyms, hypernyms, and hyponyms. To better support the exploration and search of job offers, we will integrate named entity recognition and classification (NERC) into the method, identifying companies, universities, and government agencies. The usability of the job categories might also be improved by grouping them into meaningful hierarchies, helping e-recruitment operators face rapidly changing labor markets.

References

[1] R. Boselli, M. Cesarini, F. Mercorio, and M. Mezzanzanica. How the Social Media Contributes to the Recruitment Process? In A. Rospigliosi and S. Greener, editors, *Proceedings of European Conference on Social Media (ECSM)*, pages 10–11, Brighton, UK, 2014.

[2] E. T. Cornelius, T. J. Carron, and M. N. Collins. Job analysis models and job classification. *Personnel Psychology*, 32(4):693–708, 1979.

[3] E. C. Dierdorff, J. J. Norton, C. M. Gregory, D. Rivkin, and P. M. Lewis. O*NET's National Perspective on the Greening of the World of Work. In A. H. Huffman and S. R. Klein, editors, *Driving Change with I-O Psychology*, pages 348–378. Routledge, New York, 2013.

[4] T. Gonzalez, P. Santos, F. Orozco, M. Alcaraz, V. Zaldivar, A. D. Obeso, and A. Garcia. Adaptive employee profile classification for resource planning tool. In *SRII Global Conference*, pages 544–553. IEEE, 2012.

[5] E. Malherbe, M. Diaby, M. Cataldi, E. Viennet, and M.-A. Aufaure. Field Selection for Job Categorization and Recommendation to Social Network Users. In *Proceedings of the IEEE/ACM International Conference on Advances in Social Network Analysis and Mining (ASONAM)*, pages 588–595. IEEE, 2014.

[6] R. Moore and J. DeNero. L1 and L2 Regularization for Multiclass Hinge Loss Models. In *Symposium on Machine Learning in Speech and Natural Language Processing (MLSLP)*, pages 1–5, Bellevue, WA, 2011.

[7] Z. Omary and F. Mtenzi. Machine learning approach to identifying the dataset threshold for the performance estimators in supervised learning. *International Journal for Infonomics (IJI)*, 3:314–325, 2010.

[8] Y.-J. Park and A. Tuzhilin. The Long Tail of Recommender Systems and How to Leverage It. In *Proceedings of the 2008 ACM Conference on Recommender Systems (RecSys'08)*, pages 11–18. ACM, 2008.

[9] I. Ruthven and M. Lalmas. A survey on the use of relevance feedback for information access systems. *The Knowledge Engineering Review*, 18:95–145, 2003.

[10] S. Snell, S. Morris, and G. Bohlander. *Managing Human Resources (17e)*. Cengage Learning, Boston, MA, 2013.