Inferring Missing CV Skills using PU Learning and Variational Inference

Victor Verreet, Lennert De Smet, Robin Manhaeve, Pieter Delobelle, and Jessa Bekker

KU Leuven, Leuven, Belgium victor.verreet@kuleuven.be

Abstract. An accurate CV is essential for job seekers to find a suitable job, and for vacancies to be filled by the best candidate. However, writing a CV that accurately represents the skills of a person is not easy. Skills are often omitted as a person might deem them irrelevant or simply due to forgetfulness. This can lead to problems in matching people to vacancies. We propose a novel way to complete CVs by combining generative probabilistic models with positive unlabeled learning techniques. This allows us to more accurately estimate the real skills of the job seeker, or the skills they would easily acquire. The efficacy of our proposed method is illustrated on a synthetic dataset.

Keywords: Missingness · Probabilistic models · PU learning · Variational inference

1 Introduction

Writing a good CV is not easy. Job seekers can undersell themselves, or fail to mention crucial skills altogether [4]. CVs are often incomplete, and not always a good reflection of the person's true skills [5]. Also, some skills can be less likely to be mentioned than others, resulting in a biased subset of the true skills.

The Flemish public employment services provide a CV tool in which skills can be added from a custom ontology. Because the ontology is so large, a user cannot reasonably check every skill they possess in the ontology. As a result, their CV will likely be incomplete and biased to include some skills more.

In this paper we propose a method to probabilistically complete CVs and (partially) remove the initial bias present in CVs. The method uses a generative probabilistic model to predict how a CV was created. Using techniques from positive unlabeled (PU) learning and variational inference (VI), we can train the model and use it for completing CVs. During training, the model is made aware of biases via the so-called labeling mechanism, a construct from PU learning. This allows it to be trained in a less biased way.

The work [7] also uses PU learning to complete databases and reduce bias. Many other methods that try to augment CVs, use natural language representations, such as [8], where similarities are also learned in an unsupervised context using incomplete skills. [2] uses BERT models augmented with skill co-occurrences to find better neural representations. Recurrent neural networks are used in [6] to improve job matching and reduce the effects of human errors.

2 Background

2.1 Positive Unlabeled Learning

In the setting of PU learning, the task is to train a classifier that can predict the true class, positive or negative, using only positively labeled or unlabeled data during training. The classifier does not have access to any negatively labeled instances. In a classic PU learning setup, every instance has observed attributes $x \in \mathbb{R}^r$, an unobserved true class $y \in \{p, n\}$ and an observed label $s \in \{l, u\}$. The joint distribution factorizes as

$$P(x, y, s) = P(s \mid y, x)\, P(y \mid x)\, P(x) \tag{1}$$

where $P(x)$ is the distribution of the instances, $P(y \mid x)$ is the classifier and $P(s \mid y, x)$ is the labeling mechanism. This mechanism decides with what probability each data point gets selected for labeling. The following two equivalent statements are called the PU property and define the PU learning setting

$$P(s = l \mid y = n) = 0 \quad \text{or} \quad P(y = p \mid s = l) = 1 \quad . \tag{2}$$

As such we can define the classifier C and labeling mechanism M as

$$C(x) = P(y = p \mid x) \quad \text{and} \quad M(x) = P(s = l \mid y = p, x) \tag{3}$$

which together with Eq. 2 leads to the factorization

$$P(s = l \mid x) = C(x)\, M(x) \quad . \tag{4}$$

Note that $P(s = l \mid x)$ can be estimated from data and thus provides a signal to train the classifier C. However, the C that results from training, depends on the labeling mechanism M, as is evident from Eq. 4. If M is not constant, it can lead to biased learning. Choosing the right labeling mechanism can thus have a big impact on the accuracy of the classifier. For example, if the labeling probability is assumed to be only half of the true probability, then the classifier will compensate and predict positive with a probability twice as high to explain the observed label frequencies. The survey [1] gives an overview of PU learning.

2.2 Variational Inference

Consider a probabilistic model with observed variables d and unobserved variables h. Typically, h is some hidden state that we want to uncover. Unfortunately, the true posterior $P(h \mid d)$ is usually intractable to compute. Hence, the technique of VI introduces a tractable guide distribution $Q_\theta(h)$ to approximate $P(h \mid d)$. This guide is optimised with respect to θ to lie as close to the true posterior $P(h \mid d)$ as possible. Often one uses the evidence lower bound (ELBO) as maximization criterion

$$\text{ELBO} = \mathbb{E}_Q[\log P(d, h) - \log Q_\theta(h)], \tag{5}$$

because it bounds the data log-likelihood $\log P(d) \geq \text{ELBO}$. The ELBO is often seen as a more practical criterion compared to directly optimising the data log-likelihood. Also in this work, we will use VI with the ELBO to approximate the posteriors $P(y \mid s)$ and $P(x \mid s)$ from the PU learning setting.

3 Our Method

We formalize the generative task of CV completion as follows. There is a fixed set of K skills that a person can potentially have. As such a CV is represented by a binary vector $cv \in \{0,1\}^K$. The k entry is denoted by cv_k and indicates whether skill k is mentioned on the CV. We opt for this representation because skill tokens can easily be extracted from CVs represented in natural language. Furthermore, some public employment services already store CVs in this structured form. A CV completion is a mapping $f : \{0,1\}^K \to [0,1]^K$. The entries $f(cv)_k$ of the completion $f(cv)$ indicate how likely it is for the person to possess the skill k. We make the reasonable assumption that if a person mentions a skill on their CV, that they actually possess the skill. Symbolically,

$$\forall k : cv_k = 1 \implies f(cv)_k = 1 \quad . \tag{6}$$

We will later see how this assumption allows us to frame CV completion as a PU learning problem. We will now specify the generative probabilistic model used for the completion f.

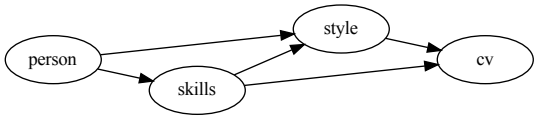

Fig. 1. A generative model used to complete CVs.

We propose the generative model depicted in Fig. 1 to complete CVs. This model starts with the *person* variable, which is represented in a latent embedding space $person \in \mathbb{R}^r$. Every person has a true skill set. Since K skills are considered in the model, they are represented as a vector $skills \in [0,1]^K$. Every person also has a CV writing style. The variable $style \in [0,1]^K$ can depend both on *person* and *skills*. It contains the probabilities that a person would mention a skill on their CV given that they have the skill. Finally, the CV is the product of both the true skills and the writing style.

The variables in the generative model can be related to the PU learning setup in the following way. The *person* variable follows the instance distribution $P(x)$, representing the diverse distribution of people. Next, the *skills* variable corresponds to the classifier $P(y \mid x)$, while *style* is the labeling mechanism $P(s \mid y, x)$. A given CV is then distributed according to $P(s \mid x)$, which factorises into the classifier and the labeling mechanism according to Equation 4.

Crucially, the PU property (Eq. 2) needs to be satisfied to formally cast optimizing the model as a PU learning problem. Given a CV completion f, we

know from Eq. 6 that mentioning a skill in a person's CV implies that the person has this skill. Using the previous relations, this property of f translates to

$$\forall k : P(y_k = p \mid s_k = l) = 1 \quad . \tag{7}$$

Put differently, the function f corresponds to the posterior distribution $P(y \mid s)$.

The generative model does differ from a classic PU learning setup in two ways: (1) there is a true class y_k for every skill k, thus we are dealing with multilabel classification, and (2) the attributes x are unobserved. However, these differences do not prevent us from applying PU techniques as they can be dealt with via multivariate PU learning and the inclusion of another generative component for x, respectively.

Two posterior distributions are of particular interest in this generative model, namely $P(y \mid s)$ and $P(x \mid s)$. The first posterior $P(y \mid s)$ indicates which skills a person is likely to actually have, given their CV. It corresponds to the CV completion function f. Although the PU property imposes that mentioning a skill implies having that skill, it is possible to have additional unmentioned skills. As such, the posterior can be used to complete a CV. The second posterior $P(x \mid s)$ is of interest because it indicates where in the latent embedding space the person lies. This allows similar CVs to be clustered and detect patterns in the data.

4 Results

Experiments have been performed to answer three research questions:

Q1 Can we learn a meaningful representation for people in the latent embedding space? Are similar CVs clustered together?
Q2 Can we infer missing skills on a CV? How do our assumptions of the labeling mechanism affect the bias during inference?
Q3 Does our approach scale well in the number of skills and instances?

All experiments have been performed on synthetic data. This has the benefit that we can control the number of skills and instances, as well as know the true labels of the instances. For real-world PU data, the labels are often not known, which makes evaluation tricky. The data has been generated as follows. Every person is assumed to belong to one of ρ clusters and every cluster has κ unique skills assigned to it. Furthermore, there are κ' universal skills that are assigned to all clusters. In total there are $K = \rho\,\kappa + \kappa'$ skills. For every skill, the probability to have it is $1 - \alpha$ if the skill is assigned to the person's cluster and α otherwise. Finally, a probability $\mu_k \in [\mu_{\text{Low}}, \mu_{\text{High}}]$ for skill k to be labeled is generated for every skill. This is the true labeling mechanism. In all experiments $N = 5000$ instances were used for training, except for the scalability experiment. We always optimized the ELBO (Eq. 5) using the Adam optimizer [3] from the Pyro[1] library with an initial learning rate of $\lambda = 0.001$ over $I = 10000$ iterations.

[1] `https://pyro.ai/`

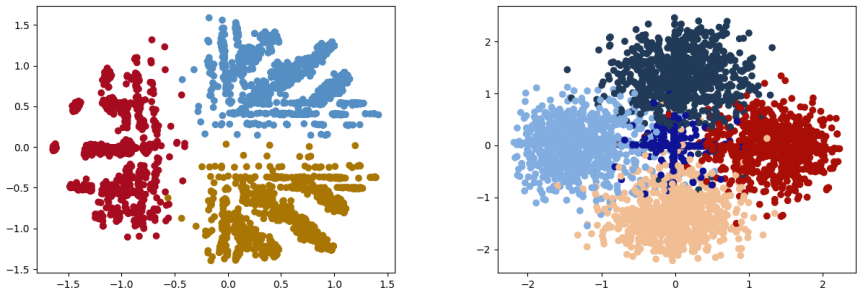

Fig. 2. Clustering in the latent space for $\rho = 3$ on the left and $\rho = 5$ on the right. Instances are assigned a colour based on the true cluster they were generated from.

To answer question Q1, we have trained generative models and visualized the data in the latent embedding space. For this reason we have chosen the latent space to have a dimension of 2. Fig. 2 shows a learned clustering for $\rho = 3$ and $\rho = 5$ and respective probabilities $\alpha = 0.1$ and $\alpha = 0.2$. As is visible in the figure, the instances from different clusters get grouped per cluster in the latent space. Close instances in the latent space will thus have similar skills and CVs.

$\alpha \backslash \beta$	1.00	0.75	0.50	0.25
1.00	0.0002	0.2540	0.2591	0.2458
0.90	0.0884	0.2624	0.2531	0.2429
0.80	0.1616	0.2546	0.2498	0.2548
0.70	0.2145	0.2506	0.2549	0.2615
0.60	0.2401	0.2536	0.2534	0.2532

Table 1. Mean squared errors after training with varying assumptions on the labeling mechanism.

For question Q2 we have trained multiple generative models that make different assumptions about the labeling mechanism and compared the mean squared error between the true skills and the predicted skills vector for these assumptions. More concretely, the true labeling mechanism is generated with $\mu_k \in [0.5, 1.0]$ independently and uniformly for every skill k. We then train the model with an assumed labeling mechanism $\mu'_k = \beta\mu_k$ with β a constant. Note that for $\beta = 1$ the correct labeling mechanism is assumed. The mean square error is reported in Tab. 1 for varying values of α and β. Assuming the correct labeling mechanism leads to the lowest error. Decreasing β means that the assumed labeling mechanism becomes less correct, and the error rapidly increases. The closer α gets to 0.5 the more the skills gets mixed between different clusters. Consequently it becomes increasingly difficult to predict the correct skills, the effect of which can also be observed in the errors.

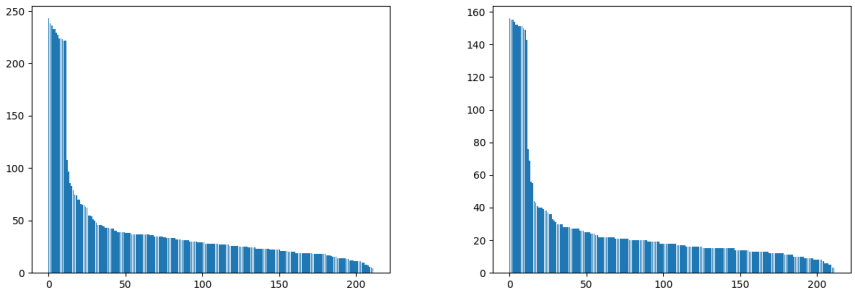

Fig. 3. Distribution of skills, ordered by number of occurrences, in 2 different clusters.

Lastly we perform an experiment with more instances and a larger number of skills to illustrate scalability for Q3. A model is trained with $\rho = 50$ clusters, $K = 400$ skills and $N = 10000$ instances. After training, the data is clustered using the K-means algorithm. The distribution of the skill occurrences, ordered by number of occurrences, are shown for 2 clusters in Fig. 3. The distribution is very similar for the remaining 48 clusters. The distribution is very peaked, indicating that meaningful clusters have still been learned in this setting of large skill and instance numbers. Every learned cluster only has a limited set of skills attributed to that cluster and has a long tail of noisy skills mixed in from other clusters.

5 Conclusion

Our method is able to probabilistically complete the skills of a CV for better job matching and other downstream tasks. A meaningful clustering is learned in the latent embedding space. Furthermore, taking into account the labeling mechanism helps to make the completions less biased. The method is scalable, which is important when applying it to real world data. For future work, the method can be applied to a large database of CVs from the Flemish public employment services. More complicated labeling mechanisms and interactions between various skills can also be investigated. Ideally, domain experts can represent their knowledge of the labeling mechanism to decrease the bias in CV completions.

Acknowledgements

This research received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme.

References

1. Bekker, J., Davis, J.: Learning from positive and unlabeled data: a survey. Machine Learning **109**(4), 719–760 (Apr 2020). https://doi.org/10.1007/s10994-020-05877-5, https://doi.org/10.1007/s10994-020-05877-5
2. Decorte, Jens-Joris and Van Hautte, Jeroen and Demeester, Thomas and Develder, Chris: JobBERT : understanding job titles through skills. In: FEAST, ECML-PKDD 2021 Workshop, Proceedings. p. 9 (2021), {https://feast-ecmlpkdd.github.io/papers/FEAST2021_paper_6.pdf}
3. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2017)
4. Nemanick, Jr, R.C., Clark, E.M.: The differential effects of extracurricular activities on attributions in resume evaluation. International Journal of Selection and Assessment **10**(3), 206–217 (2002)
5. Piopiunik, M., Schwerdt, G., Simon, L., Woessmann, L.: Skills, signals, and employability: An experimental investigation. European Economic Review **123**, 103374 (2020)
6. Qin, C., Zhu, H., Xu, T., Zhu, C., Jiang, L., Chen, E., Xiong, H.: Enhancing person-job fit for talent recruitment: An ability-aware neural network approach. In: Proceedings of the 41st International ACM SIGIR Conference. pp. 25–34 (06 2018). https://doi.org/10.1145/3209978.3210025
7. Schouterden, J., Bekker, J., Davis, J., Blockeel, H.: Unifying knowledge base completion with pu learning to mitigate the observation bias. Proceedings of the AAAI Conference on Artificial Intelligence **36**(4), 4137–4145 (Jun 2022). https://doi.org/10.1609/aaai.v36i4.20332, https://ojs.aaai.org/index.php/AAAI/article/view/20332
8. Zbib, R., Alvarez, L., Retyk, F., Poves, R., Aizpuru, J., Fabregat, H., Simkus, V., García-Casademont, E.: Learning job titles similarity from noisy skill labels (07 2022). https://doi.org/10.48550/arXiv.2207.00494