Job-Candidate Matching Research

José Gonzalo Domínguez – jose.dominguez@ut-capitole.fr

Université Toulouse Capitole 1, Toulouse 31100, France

Contenido

Abstract. The aim of this internship is to explore the impact of machine learning techniques on job-candidate matching. Based on a sample of data containing both job offers and candidates, I will implement various learning algorithms to assess their quality. In a second phase, the work is extended by designing a system of vigilance against biases (gender, geography...) with a view to ensuring fairness between candidates.

1. Preamble

1.1 IRIT

This project was proposed and funded by IRIT, a Joint Research Unit (UMR 5505) of the Centre National de la Recherche Scientifique (CNRS), the Institut National Polytechnique de Toulouse, the Université Toulouse 3 Paul Sabatier, the Université Toulouse Capitole and the Université Toulouse Jean Jaurès.

It is one of the largest UMR at the national level, with its 600 members, permanent and non-permanent, and about 100 external collaborators. Due to its multi-tutorial nature (CNRS, Toulouse Universities), its scientific impact and its interactions with other fields, the laboratory constitutes one of the structuring forces of the IT landscape and its applications in the digital world, both at regional and national level.

1.2 Agile methodology

Agile methodologies are iterative and incremental, which means it's known for breaking a project into smaller parts and adjusting to changing requirements. They prioritize flexibility, collaboration, and customer satisfaction.

We follow this approach because we thought that was the most adequate for our situation. Along the internship, we divide the project in sprints between every meeting focusing on low number of tasks every time. Requirements changed often since at beginning the path that we should have taken it was not very clear. Following agile methodologies allowed as to adapt quickly to what we need it at each time. Also, time management was a main issue since the internship was not very long and we had to be reasonable when establishing goals.

2. Job-Candidate Matching

In this internship, I study the design of matching markets which address problems of assignment where each participant is matched in some way, and where participants differ in preferences for their part of a match. Matching markets are used in several real-world settings: assigning graduating medical students to hospitals, public school assignment and more.

There are two different types of matching problems:

- two-sided matching, where there are two distinct groups of agents, and the problem is to match each agent on one side of the market with an agent on the other side.
- assignment problems, where there are items and agents with preferences on items, and the problem is to assign a distinct item to each agent.

The difference between these settings is that in two-sided matching, an agent does not just make a choice but also must be chosen; in comparison, in assignment problems an agent picks an item (and items themselves do not have preferences).

Our scenario is an example of two-side market, there is a group of candidates and a group of jobs, and the problem is to match each agent in one group to an agent in the other group, such that no agent is matched to more than one more agent. Also, preferences from every side of the market are known and can be deduced, for example, from the applications and requirements of every job posting. The outcome is a matching.

2.1 Gale-Shapley algorithm

It is an algorithm for finding a solution to the stable matching problem. It has been used for the National Resident Matching program since the early 1950s.

The stable matching problem seeks to pair up equal numbers of participants of two types, using preferences from each participant. The pairing must be stable: no pair of participants should prefer each other to their assigned match. In each round of the Gale–Shapley algorithm, unmatched participants of one type propose a match to the next participant on their preference

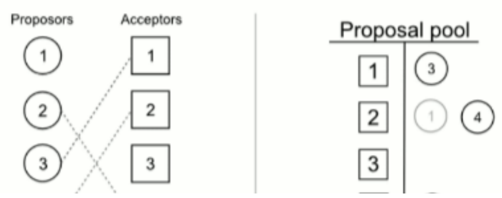

Figure 1. Gale-Shapley algorithm scheme.

list. Each proposal is accepted if its recipient prefers it to their current match.

The resulting procedure is a truthful mechanism from the point of view of the proposing participants, who receive their most-preferred pairing consistent with stability. In contrast, the recipients of proposals receive their least-preferred pairing.

Solution

In 1962, Shapley and Gale proved that:

$$\forall \ x \text{ jobs} \wedge \ y \text{ candidates} \mid x = y$$
$$\exists \text{ solution}$$

In terms of job applicants and employers, it can be expressed as follows: in each round, one or more employers with open job positions each make a job offer to the applicant they prefer, among the ones they have not yet already made an offer to.

Each applicant who has received an offer evaluates it against their current position (if they have one). If the applicant is not yet employed, or if they receive an offer from an employer, they like better than their current employer, they accept the best new offer and become matched to the new employer (possibly leaving a previous employer with an open position). Otherwise, they reject the new offer.

This process is repeated until all employers have either filled their positions or exhausted their lists of applicants.

Implementation

This approach, where the proposers are the candidates, could be good for the interview process. Since they proposers are favoured in this matching method then they are more likely to go to the interview. Reducing the probability that the candidate receives one of it least preferred choices then it increases the attendance in interviews. Of course, it must [1]be taken into account that preferences from acceptors are also considers although in less magnitude of those from proposers.

In the opposite approach, where the positions/companies are the proposers, it boosts the choice of the most prepared candidates according to the companies' opinion. A bottleneck effect could appear since it is likely that most of the companies choose only the top candidates, leading to lack of proposals to the least prepared.

3. Recsys Challenge 2017

The RecSys Challenge 2017[1] was organized by XING, Politecnico Milano and Free University of Bozen-Bolzano. XING is a social network for business. People use it, for example, to find a job and recruiters use XING to find the right candidate for a job.

This edition of the RecSys Challenge aimed to better connect job seekers and recruiters via job recommendations. So, this challenge's objective was close to that of the a stable matching.

The ACM RecSys 2017 dataset [1] is a large-scale data collection of user-job interactions from the career oriented social network XING (European analogue of LinkedIn). Importantly, this is one of the only publicly available datasets that contains both user and item content information and cross-interactions between them. In total there are 1.5M users, 1.3M jobs and over 300M interactions. Interactions are divided into:

- 0, impression.
- 1, click.
- 2, bookmark.
- 3, reply or apply.
- 5, recruiter interest.

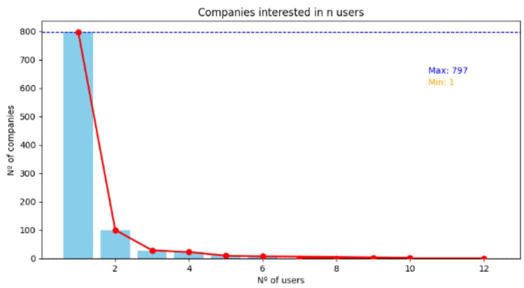

Figure 2. Number of companies interested in n users.

Each interaction is recorded with the corresponding type and timestamp.

[1] https://www.recsyschallenge.com/2017/

3.1 Statistics

We process the interaction data by removing duplicate interactions (i.e. multiple clicks on the same item) and deletes, and collapse remaining interactions into a single binary matrix R where R(u, v) = 1 if user u interacted with job v and R(u, v) = 0 otherwise. The training set contains 18.7M interactions. There are 1M of users and 0.5M items in the training set. It also contains 1.2 M of rows with same (user, item), therefore, for the same (u, i) there can be several interactions. In total, we have 1360 clicks or visits from different jobs to different users and in average, they are interested in 1.4 users.

In the first stage we considered "match" as the situation where company c saw the profile of user u and replied to c:

$$\text{Match } 3 = A[\text{interaction} = 3] \cap A[\text{interaction} = 5],$$

where A is the training set. Nevertheless, the low number of interactions from companies (interaction = 5) resulted in a low number of matches. Then, we thought of considering interactions 2 and 3 as well:

$$\text{Match } 123 = (A[\text{interaction} = 3] \mid A[\text{interaction} = 2] \mid A[\text{interaction} = 1]) \cap A[\text{interaction} = 5],$$

meaning that any user u with some interaction to company c and c also being interested in u would be considered.

Table 1: Statistics about interactions, from 2 different subsets.

	Interactions 3	Interactions 123
mode	1 (23,476 users)	1 (188,953 users)
max/user	150	695
average/user	1.8	6.8
total	62,321	3,727,897

We can see that there is a great difference between the total interactions between the 2 subsets. Therefore, Match 123 subset would be more interesting to study since it is expected to include more matches. The result considering this subset was 81 new "matches" out of the 3.7 M interactions. With those interactions we compute a simple directed graph, which represents the matches.

After that, we decided that it would be interesting to know the total interactions from those users that at least has one match and who appear in the graph from figure 3. Consequently, we would know how many interactions were not successful compared to the obtained matches, in other words, we were looking for the success ratio.

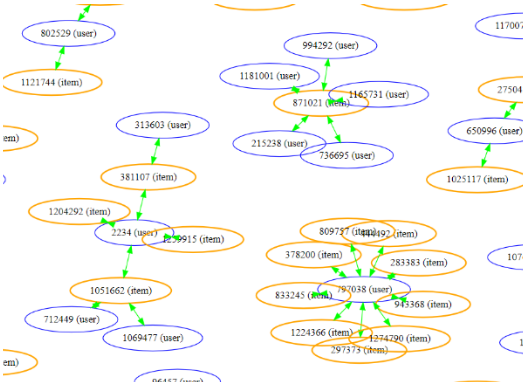

Figure 3. Fragment of the Match 123 graph that depicts mutual interactions from jobs(orange) and users (blue).

The result was the graph in figure 4, which includes 754 vertices where 60 of them are users. The green edges are same edges from graph in figure 3 that represent the matches, and the new black edges represent the interactions that did not receive any feedback from job positions. We also computed the ratio between matches and interactions for every user, in average matches/interactions = 0.1 of those who obtained at least one. The conclusions that we could derived from these statistics is that users should need to interact around 10 times in the best cases to obtain a match, considering that the matches are correlated to the number of interactions.

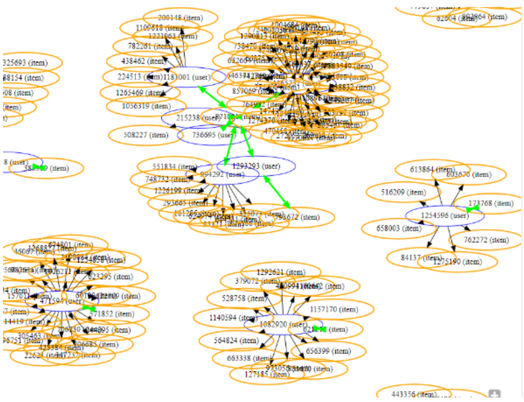

Figure 4. Fragment of graph showing all the interactions from users who at least have 1 match (in green).

4. Machine Learning

The implementation of machine learning in this research had the objective to compute a stable

matching efficiently and study in detail large amount of data. It will help us to predict probabilities of a sequence and find generalizable predictive patterns [2]. With an accurate trained model, we could study more specific scenarios:
- predicting importance of individual candidate attributes.
- recommending jobs to candidates based on the activity of similar users.
- predicting the variable demand of candidates regarding type of job or industry.

4.1 Treatment of imbalanced data for Machine Learning[2].

As we could see in our statistics, we found in the Recsys dataset that the number of unique user-item pairs for each interaction type was significantly reduced from the original count. For example, the number of unique impressions was reduced to only 21M from the original set of 314M in the offline dataset. Also, the number of interactions from companies, which we could see as the preferences of jobs over users, is extremely low if we compared to other types of interactions. This made the problem extremely sparse and will hinder the appropriate learning of our model.

To describe all the metrics that are affected by imbalanced data we first have to define the following:
- True positive (TP). A correctly predicted value by a classifier indicating the presence of a condition or characteristic.
- True negative (TN). A correctly predicted value by a classifier indicating the absence of a condition or characteristic.
- False positive (FP). A wrongly predicted value by a classifier indicating that a particular condition or attribute is present when it's not.
- False negative (FN). A wrongly predicted value by a classifier indicates that a particular condition or attribute is not present when it is.

Here are the common evaluation metrics affected by imbalanced data:
- Accuracy. It measures the ratio between predicted instances to the total instances in the dataset.
- Precision. It measures the proportion of correctly predicted positive instances out of all the positive predicted positive instances.

$$Recall = \frac{TP}{TP + FN}$$

Figure 5. Formula of recall metric.

- Recall. The true positive rate measures the proportion of correctly predicted positive instances out of all actual positive instances.

4.2 Techniques to Handle Imbalanced Data: Oversampling

It is a resampling technique that aims to balance the class distribution by increasing the number of instances in the minority class. The goal is to ensure that the model sees a more balanced representation of both classed during training.

Pros: improved model performance. It helps the model better learn the characteristics of the minority class, leading to improved classification performance, especially for the minority class.

Cons: overfitting risk. Duplicating or generating synthetic instances can lead to overfitting if not controlled properly, especially if the synthetic data is too like the existing data.

4.3 Decision trees

They are non-parametric supervised learning method used for classification and regression. The goal is to predict a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Some advantages of using decision tress are:
- Simple to understand and to interpret, since they can be visualized.
- Requires little data preparation. Other techniques require data normalization, dummy variables need to be created and blank values must be removed.

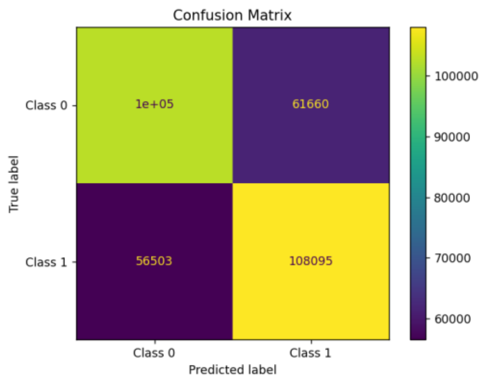

Figure 6. Confusion matrix showing the accuracy in both classes.

Our first objective using decision trees was to predict whether a user will receive interactions from companies or not. So, we were interested in finding the correlation between

[2]https://semaphoreci.com/blog
[3]https://scikit-learn.org

users attributes and the number of interactions that these users receive from jobs. This could help users in job platforms such as XING to better understand what companies are looking for. We train our model with lists of attributes from every user, every list contains binary values indicating whether the user has certain attribute, and the target values are binary indicating if the user received interactions.

Table 2: Classification report with oversampling.

Classes	**Precision**	**Recall**	**Support**
0	0.65	0.62	164340
1	0.64	0.66	164598
Accuracy		0.64	328938

The accuracy obtained in the results was not the ideal one but was better than the case when we do not use oversampling. Also, we checked that it was easier to classify users in 2 groups: those with interactions from companies and those with nothing.

The result showed that the most decisive attributes were 800 and 732. Importance is measured according to the capacity of the attribute to obtain more accurate results[3].

There are some other possibilities that we believe can increase the accuracy:

- Neighbors classifier
- Random forests. Random forest regressor could predict the approximate number of interactions that a user will receive.

5. Fairness theoretical analysis

In our research, we use the concept of unobservable theoretical construct to define fairness. It is an abstraction that describes the phenomena of theoretical interest such socioeconomic status or fairness.[6]

We propose measurement modeling from the quantitative social sciences as a framework for understanding fairness in computational systems. Quantitative social sciences refer to the field of study that applies statistical, mathematical, and computational techniques to understand social phenomena. Computational systems often involve unobservable theoretical constructs, such as socioeconomic status, teacher effectiveness, and fairness. Such constructs cannot be measured directly and must instead be inferred from measurements of observable properties (and other unobservable theoretical constructs) thought to be related to them—i.e., operationalized via a measurement model.

The contested nature of fairness makes it inherently hard to measure: If there are multiple theoretical understandings of a construct, then it is imperative to articulate which understanding is being operationalized. A measurement model is a statistical model that links unobservable theoretical constructs, operationalized as latent variables, and observable properties—i.e., data about the world.

$$\min_{f \in \mathcal{F}} \mathbb{E}\big[\ell(Y, f(X))\big]$$

$$\text{such that } \forall a \in \mathcal{A}: \quad \mathbb{E}\big[\ell(Y, f(X)) \bigm| A = a\big] \leq \zeta_a.$$

Figure 8. Bounded group loss according to [7]

5.1 Fairlearn

There are many approaches to conceptualizing fairness. In Fairlearn, they follow the approach known as group fairness, which asks: Which groups of individuals are at risk for experiencing harms?

The relevant groups (also called subpopulations) are defined using sensitive features (or sensitive attributes), which are passed to a Fairlearn estimator as a vector or a matrix called sensitive features (even if it is only one feature). The term suggests that the system designer should be sensitive to these features when assessing group fairness.

Let X denote a feature vector used for predictions, A be a single sensitive feature (such as age or race), and Y be the true label. Parity constraints are phrased in terms of expectations with respect to the distribution over (X, Y, A). For example, in Fairlearn, we consider the following types of parity constraints:

- Demographic parity. It seeks to mitigate allocation harms, whereas bounded group loss primarily seeks to mitigate quality-of-service harms. Equalized odds and equal opportunity can be used as a diagnostic for both allocation harms as well as quality-of-service harms. [7]

$$\mathbb{E}[h(X) \mid A = a] = \mathbb{E}[h(X)] \quad \forall a.$$

Figure 7. Formula for demographicl parity in binary classification

- Bounded group loss. In this case, the constrained optimization formulation follows directly from the definition. For the sake of flexibility, we allow specifying a different bound C_a for each attribute value.

If it is not possible to achieve a loss below C on some group, then to achieve fairness we need to collect more data for that group or develop more informative features for individuals in that group.

6. Conclusion

This internship and the work done during this period was dedicated to research about job-candidate matchings and how to effectively choose the most adequate person for each position. The conclusion derived from this study tend to:

- facilitate users from platforms like the one mentioned in this research, XING, to understand the features which companies are interested in.
- Implement new techniques and technologies in job platforms to facilitate and assure the best matching possible.

Extension points

Collaborative filtering: technique used in well-known companies; it is used to recommend items to users based on the interests of other users. With Recsys dataset it is possible to measure similarity between users and train a model to predict whether a user will interact with certain job posting.

Study temporary trends: using timestamp can be useful to improve the accuracy of recommendations. To address these issues, a novel temporal recommendation can be implemented using predicting users' similarities in the future. To this end, the similarity between two users is calculated in different time-windows which leads to having a trend of the similarity between them over time. [10]

7. Research papers to consider.

DropoutNet: Addressing Cold Start in Recommender Systems [3]

Models chosen for recommender systems focused on modelling user-item interactions, and few of them have been developed for cold start. They proposed a neural network based latent model called DropoutNet to address the cold start problem in recommender systems. In this approach they focus on the optimization and show how neural networks can be explicitly trained for cold start through dropout.

Content-based Neighbor Models for Cold Start in Recommender Systems [4]

Cold start remains a prominent problem in recommender systems. Slow progress in this area can be partially attributed to the lack of publicly available benchmarks to validate and compare models.

The challenge organizer XING released a large, scaled data collection of user-job interactions from their career oriented social network. In this paper we present our approach to this challenge, we used a combination of content and neighbor-based models winning both offline and online phases. Our model produced the most consistent online performance wining four of the five online weeks.

Natural Interviewing Equilibria for Stable Matching [5]

In this paper we study a market where one side maintains a common preference master list, while the other side have idiosyncratic preferences which they can be refine by conducting a limited number of interviews. We provided a payoff function for this imperfect information game and find that this game always has a pure strategy equilibrium.

Toward job recommendation for all [8]

This paper focuses on e-recruitment, i.e. the design and exploitation of recommender systems selecting job ads best suited to job seekers. This type of recruitment faces some difficulties compared to item recommendations. Firstly, due to the sensitivity of the data, the domain lacks a comprehensive open benchmark supporting the comparative assessment of the algorithms. The RecSys challenge 2017 dataset to their knowledge is the most representative dataset (impossible to access, as far as I know).

Secondly, e-Recruitment involves rival goods (a single job can be attributed to a single job seeker), with a sparse interaction matrix, and recommendation mostly considers new job seekers and recent job ads. Both features increase the complexity of the e-R problem, additionally the description of the data is heterogenous.

An Empirical Study of Rich Subgroup Fairness for Machine Learning [9]

This study empirically evaluates Kearns et al.'s 2018 rich subgroup fairness algorithm on four real datasets. Key findings include:

- The algorithm converges quickly with heuristic implementations.
- Significant fairness gains are achieved with minimal accuracy loss.
- It outperforms traditional marginal fairness approaches in addressing subgroup unfairness.

- Rich subgroup fairness proves to be a viable concept in practice.

The research demonstrates the algorithm's effectiveness in balancing fairness and accuracy across various subgroups, supporting its practical application in machine learning fairness.

8. References

[1] F. Abel, Y. Deldjo, M. Elahi, and D. Kohlsdorf. Recsys challenge 2017. http://2017. recsyschallenge.com, 2017.

[2] Bzdok, D., Altman, N. & Krzywinski, M. Statistics versus machine learning. Nat Methods 15, 233–234 (2018). https://doi.org/10.1038/nmeth.4642

[3] Volkovs, M. N., Yu, G., Poutanen, T., & Shaukat, A. (2017). Content-based Recommendations with Poisson Factorization. In Advances in Neural Information Processing Systems (NIPS). Retrieved from https://www.cs.toronto.edu/~mvolkovs/nips2017_deepcf.pdf

[4] Volkovs, Maksims et al. "Content-based Neighbor Models for Cold Start in Recommender Systems." Proceedings of the Recommender Systems Challenge 2017 (2017): n. pag.

[5] Drummond, Joanna et al. "Natural Interviewing Equilibria for Stable Matching." (2016).

[6] Abigail Z. Jacobs and Hanna Wallach. 2021. Measurement and Fairness. In Conference on Fairness, Accountability, and Transparency (FAccT '21), March 3–10, 2021, Virtual Event, Canada. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3442188.3445901

[7] Agarwal, Alekh et al. "Fair Regression: Quantitative Definitions and Reduction-based Algorithms." ArXiv abs/1905.12843 (2019): n. pag.

[8] Guillaume Bied, Solal Nathan, Elia Perennes, Morgane Hoffmann, Philippe Caillou, et al.. Toward Job Recommendation for All. IJCAI 2023 - The 32nd International Joint Conference on Artificial Intelligence, Aug 2023, Macau, China. pp.5906-5914, ⟨10.24963/ijcai.2023/655⟩. ⟨hal-04245528⟩

[9] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. 2019. An Empirical Study of Rich Subgroup Fairness for Machine Learning. In Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* '19). Association for Computing Machinery, New York, NY, USA, 100–109. https://doi.org/10.1145/3287560.3287592

[10] Joorabloo, Nima & Jalili, Mahdi & Ren, Yongli. (2019). A New Temporal Recommendation System based on Users' Similarity Prediction. 555-560. 10.5220/0008377205550560.