Designing Recommender Systems for the Labor Market

Guillaume Bied

université PARIS-SACLAY

Designing recommender systems for the labor market

Concevoir et évaluer les algorithmes de recommandation pour le marché du travail

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580 : sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat: informatique
Graduate School : Informatique et sciences du numérique. Référent : Faculté des sciences d'Orsay

Thèse préparée dans les unités de recherche LISN (Université Paris-Saclay, CNRS) et CREST (IP Paris, CNRS), sous la direction de **Bruno CREPON**, Professeur (CREST), le co-encadrement de **Philippe CAILLOU**, Maître de Conférences (Université Paris-Saclay)

Thèse soutenue à Paris-Saclay, le 10 juillet 2024, par

Guillaume BIED

Composition du jury

Membres du jury avec voix délibérative

Jamal ATIF Professeur, Université Paris-Dauphine	Président
Tijl de BIE Professeur, University of Ghent	Rapporteur & Examinateur
Christine LARGERON Professeure, Université Jean Monnet	Rapporteur & Examinatrice
Charlotte LACLAU Maîtresse de Conférences, Telecom Paris	Examinatrice
Thomas LE BARBANCHON Associate Professor, Bocconi University	Examinateur

THESE DE DOCTORAT
NNT : 2024UPASG035

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre: Concevoir et évaluer les algorithmes de recommandation pour le marché du travail
Mots-clés: systèmes de recommandation, marché du travail, équité, congestion

Résumé: En apprenant des appariements passés, les systèmes de recommandation ont le potentiel de réduire les frictions informationnelles sur le marché du travail. Cette thèse pose la question de la conception et de l'évaluation d'algorithmes de recommandation d'offres d'emploi, en s'appuyant sur des données détaillées fournies par le service public de l'emploi français.

Premièrement, nous proposons une nouvelle architecture neuronale pour la recommandation d'offres d'emploi. Cette architecture présente l'avantage de répondre au problème du démarrage à froid tout en passant à l'échelle. L'approche proposée est comparée à l'état de l'art en termes de performance hors-ligne. Elle est également évaluée sur le terrain en termes de satisfaction des utilisateurs au moyen d'expériences randomisées à grande échelle.

Deuxièmement, nous examinons les objectifs possibles qu'un concepteur pourrait assigner à un algorithme de recommandation d'offres d'emploi. Cette analyse est réalisée dans le cadre d'un modèle économique, qui nous permet de discuter les mérites et limites de différentes approches plausibles (satisfaire les critères de recherche exacts des demandeurs, apprendre des candidatures ou des embauches), et de les confronter aux besoins des demandeurs d'emploi.

Troisièmement, nous étudions le problème de la congestion qui peut survenir si les recommandations se concentrent sur un ensemble excessivement restreint d'offres, créant des conséquences nuisibles au niveau agrégé. Nous proposons une approche algorithmique utilisant des outils issus du transport optimal computationnel pour limiter ce phénomène, et étudions ses performances sur des données publiques et propriétaires.

Enfin, comme les algorithmes de recommandations sont entraînés sur des données issues du monde réel, ils peuvent reproduire ou aggraver certains comportements indésirables (discriminations) existants sur le marché du travail. Afin de répondre à ces inquiétudes, nous réalisons un audit fin de l'algorithme de recommandation (entraîné à partir des embauches) en se focalisant sur les inégalités de genre. En s'inspirant de la littérature en économie du travail, nous proposons des mesures des écarts genrés en termes de caractéristiques des recommandations (salaire, type de contrat...), en moyenne ou conditionnellement aux qualifications et préférences des demandeurs d'emploi. Selon nos résultats, l'algorithme reproduit, sans aggraver, les biais de genre présents dans les données d'entraînement. Nous proposons également une approche dite de "post-traitement" dont l'objectif est de réduire les écarts femmes-hommes en termes de caractéristiques des offres recommandées. Nous décrivons les arbitrages entre performance et équité que cette intervention implique.

Title: Designing recommender systems for the labor market
Keywords: recommender systems, labor market, fairness, congestion

Abstract: Recommender systems have the potential to reduce information frictions on the labor market by leveraging past interactions between job seekers and recruiters. This thesis presents several contributions to the design and evaluation of job recommender systems, leveraging detailed real-world data provided by the French Public Employment Service.

First, we propose a novel neural architecture for job recommendation, aimed at providing relevant recommendations in the cold-start setting while maintaining scalability. The proposed approach is compared to the state of the art in terms of off-line performance. It is also evaluated in the field in terms of user satisfaction, measured in the context of large-scale randomized experiments.

Second, we discuss the possible objectives that a designer could assign to a job recommender system. Based on a formal economic model, we discuss the merits and limits of different plausible approaches (satisfying job seekers' exact search parameters, optimizing for application or hiring probability), and confront them to job seekers' needs.

Third, we study the issue of the congestion that may arise if recommendations focus on an excessively small set of job ads, creating negative aggregate consequences. Leveraging tools from the computational transport literature, we propose a post-processing approach to congestion-avoiding recommendation, and assess its performance on proprietary and public datasets.

Finally, as recommender systems are trained on real-world data, they may replicate or worsen undesirable behaviors (discrimination) that may exist on the labor market. We provide a detailed audit of the proposed recommender system (trained on hiring data) in terms of gender inequalities. Drawing inspiration from the labor economics literature on the gender wage gap, we propose measures for gender gaps in recommendation characteristics, on average or conditionally on job seekers' qualifications and preferences. We find that the algorithm reproduces, but does not increase, gender gaps that exist in its training data. We propose a post-processing approach to reduce unconditional or conditional gender gaps, and describe the trade-offs it entails.

Acknowledgements

I am extremely grateful to my advisors Bruno Crépon and Philippe Caillou, as well as to Michèle Sebag, for trusting me and enabling me to grow as a researcher, while continuously providing invaluable advice and support throughout my PhD.

I would like to thank Tijl De Bie and Christine Largeron for reviewing this manuscript, as well as Jamal Atif, Charlotte Laclau and Thomas Le Barbanchon for accepting to take part in the jury.

I would also like to express my utmost gratitude to my co-authors: Victor Alfonso-Naya, Philippe Caillou, Bruno Crépon, Christophe Gaillac, Morgane Hoffmann, Charly Marie, Solal Nathan, Mitia Oberti, Elia Pérennès, Bertille Picard, and Michèle Sebag. In particular, it is hard to find words to thank Elia, on whom I had the chance of being able to rely throughout the ups and downs of the VADORE project, and whose thesis this is nearly as much as mine; and Morgane, for essential contributions and suggestions, and for constant intellectual stimulation, friendship and support throughout the last few years.

I warmly thank all members of the research community I had the chance to interact with at CREST and LISN. In a non-exhaustive fashion, this PhD would not have been the same without Chloé Antoine, Jeanne Astier, Alicia Bassière, Marion Brouard, Pauline Carry, Arnault Chatelain, Héloïse Cloléry, Léa Dubreuil, Ghewa Eldora, Adelaide Fabbi, Aurélien Frot, Mario Herrera, Margarita Kirneva, Alice Lapeyre, Florent Le Clerc, Claire Leroy, Eva Lestant, Pauline Lesterquy, Pauline Leveneur, Esther Mbih, Federica Meluzzi, Matéo Moglia, Martin Mugnier, Arnaud Pandevant, Félix Pasquier, Inès Picard, Roland Rathelot, Pedro Vergara Merino, Felix Schleef, Yuanzhe Tang, Arne Uhlendorff, Giulia Vattuone, Vincent Verger and Yiyun Zheng. I am especially grateful to office 4095 members, in particular Héloise de Gaulmyn, Morgane Hoffmann, Elio Nimier-David, Inès Moutachaker, Myriam Kassoul and Théo Valentin, for lively discussions and heartwarming support.

I thank Nicole Bidoit-Tollu, Rémi Flamary and Anne Vilnat for taking part in my PhD follow-up committees.

I thank participants to the conferences and seminars I had the opportunity to present at, in particular at the FEAST workshops and IAB. The chapter on gender gaps greatly benefited from remarks at a CREST seminar, in particular from Xavier d'Haultfoeuille. The chapter on congestion benefited from discussions with Gwendoline de Bie and Marco Cuturi.

This work would not have been possible without *France Travail*'s support. In particular, I would like to thank Paul Beurnier, Hélène Caillol, Emmanuel Chion, Pierre-Antoine Corre, Yann de Coster, Alexandre Garel, Nicolas Greffard, Axel Gaugler, Florent Lefort, Cyril Nouveau, Sébastien Robidou and Chantal Vessereau.

I gratefully acknowledge financial support from GENES and Institut Louis Bachelier. The VADORE project benefited from funding by the DataIA convergence institute and the Chaire de Sécurisation des Parcours Professionnels. Work on fairness issues in job recommendation was partially funded by *France Travail*.

I would like to thank the administrative staff at ED STIC, at INRIA, CREST / GENES and Institut Louis Bachelier, as well as the IT team at CREST.

Finally, the support of my friends and family has been invaluable: thank you.

Contents

Introduction **8**

Part I: Accurate sparse job recommendation **16**

1 Related work **17**
1.1 Recommender systems (with implicit feedback) 17
1.1.1 Goals and evaluation of recommender systems 18
1.1.2 Types of recommender systems 20
1.1.3 Common approaches and architectures 22
1.2 Job recommender systems 25

2 Data description and analysis **28**
2.1 Institutional and geographical setting 28
2.2 Job seekers 29
2.3 Job ads 34
2.4 Interactions 37
2.4.1 Applications 37
2.4.2 Hirings 39

3 The Muse algorithm **43**
3.1 Proposed approach 43
3.1.1 Muse.0: candidate retrieval stage 43
3.1.2 Muse.1 and Muse.2: re-ranking stage 47
3.2 Validation: experiments *in silico* 48
3.2.1 Experimental settings 48
3.2.2 Results 49
3.3 Partial conclusion 53

Part II: Job recommendation beyond accuracy **55**

4 Value alignment **56**
4.1 Two job recommender systems 59
4.2 Do the two recommender systems recommend similar job ads? . . . 61
4.3 Does a recommendation algorithm dominate the other regarding job seekers' objective? 63
4.4 Should recommender systems reproduce job seekers' behavior? . . . 65
4.5 Partial conclusion 69

5 Field experiments **71**
5.1 March 2022 field experiment . . . 71
5.1.1 Experiment design . . . 72
5.1.2 Analysis . . . 73
5.2 June 2023 field experiment . . . 76
5.2.1 Experimental design . . . 76
5.2.2 Analysis . . . 78

6 Congestion-avoiding job recommendation **81**
6.1 Related work . . . 82
6.2 Overview of CAROT . . . 84
6.3 Results . . . 86
6.3.1 MAR Dataset . . . 86
6.3.2 JOB Dataset . . . 88
6.4 Partial conclusion . . . 89

7 Fairness in job recommendations: estimating, explaining, and reducing gender gaps **92**
7.1 Related work . . . 93
7.2 Measuring gender gaps in recommendations . . . 95
7.2.1 Outcomes of interest . . . 95
7.2.2 Measuring unconditional and conditional gaps . . . 96
7.2.3 Discussion . . . 97
7.3 Experimental setting . . . 99
7.4 Results . . . 100
7.4.1 Recommendation performance is higher for women . . . 100
7.4.2 Characteristics of job ads recommended to men and women are different . . . 101
7.4.3 Comparison of gender gaps in recommendations, hirings and applications . . . 103
7.5 A post-processing approach to reducing gender gaps in recommendations . . . 105
7.5.1 Methodology . . . 105
7.5.2 Results . . . 107
7.6 Partial conclusion . . . 111

Conclusion and perspectives **113**

Publications **117**

References **118**

Appendices **131**

A MUSE: hyper-parameters and configuration **133**

B MUSE: recall heterogeneity analysis **135**

C Complements on value alignment **137**
C.1 Algorithm details . . . 137
C.2 Calibration of the ML score into a hiring probability . . . 138
C.3 Complements on the model . . . 140
C.4 Robustness check - modelling applications without fixed effects . . . 143

D Complements on field experiments **144**
D.1 March 2022 field experiment . . . 144
D.1.1 Algorithms . . . 144
D.1.2 Survey design . . . 146
D.1.3 Attrition differential . . . 149
D.2 June 2023 field experiment . . . 150
D.2.1 Survey . . . 150

E Complements on congestion-avoiding job recommendation **152**
E.1 Additional tables . . . 152
E.2 Higher entropic regularization may not reduce congestion . . . 156
E.3 Hyperparameters . . . 158

F Complements on gender gaps **161**
F.1 Sample characteristics . . . 161
F.2 Additional tables . . . 163
F.3 Additional figures . . . 168
F.4 Details regarding the post-processing approach . . . 169
F.4.1 Formulation of gender gap constraints . . . 169
F.4.2 Efficiently evaluating L . . . 170

G Thesis summary in French (*résumé substantiel*) **171**

Introduction

Motivations According to the French National Institute of Statistics and Economic Studies (INSEE), the unemployment rate in France (excluding the island of Mayotte) was of 7.1% in the first quarter of 2023, and the share of young people aged 15 to 29 neither in employment nor training of 12.3%. Since the labor market constitutes a key source of income, social status and integration in developed countries, reducing unemployment is a key objective for public policies.

Unemployment can stem from a fundamental mismatch of skills and location between people and jobs ("structural" unemployment), but also from search frictions ("frictional" unemployment). The importance of the latter, linked to the costs of gathering information about job vacancies and labor availability, was highlighted in economics by Diamond, Mortensen and Pissarides [Dia82; Mor82; Pis85]. Reducing informational frictions and improving the labor market's "matching function" thus forms an avenue for public policies to reduce unemployment.

As job search increasingly moves online [Aut01; Kir22], recommender systems (RS), which help users find relevant items in large databases, may form a welcome addition to the policy maker's toolbox by helping job seekers locate relevant job ads at low marginal cost.

Evidence on labor market recommendation Despite growing interest in job recommendation in computer science [FC21; DB21; Mas+22], the causal impact of deploying job recommender systems at scale on labor market outcomes remains under investigation in economics.

Studies exist on the effects of automated occupation recommendation, using measures for occupation proximity based on labor market transitions or skill similarity between jobs. In a field-in-the-lab randomized control trial (RCT), [BKM19] study the effect of algorithmic occupation recommendations on job search in Edinburgh. The intervention increased the breadth of job applications and the number of job interviews for the treated, with effects driven by job seekers who initially searched narrowly. Translating occupational recommendation to the large-scale setting of the Danish Public Employment Service, [Alt+22] find their intervention to have positive effects on hours worked and labor earnings when targeting limited shares of job seekers on local labor markets (especially in occupations with limited prospects), but that spillover effects reduce the intervention's impact at scale.

Other studies directly provide evidence on the effect of personalized algorithmic recommendations in RCT settings. In a large scale experiment in France, [Beh+22] study recommendations of firms based on their predicted hirings at the firm $\times$ occupation level. The intervention had a net positive effect of 2% on women's

rate of return to employment (but no effect on men's), due to both an increase in search effort and a focus on recommended firms. In Sweden, [LHR23] study job recommendations generated by collaborative filtering applied to job seekers' click history, and find 0.6% higher employment within 6 months following first exposure to recommendations. Both studies find that recommendations expanding occupational scope tend to have higher effects.

The present work is concerned with the design of job ad recommender systems for the labor market, in the context of a partnership with the French Public Employment Service (PES), *France Travail*. By leveraging the wealth of information gathered by the institution on the labor market and modern machine learning methods, algorithmic job recommendations could improve matching on the labor market by reducing the cognitive effort required to explore large number of ads and leveraging other job seekers' past experience to identify promising opportunities. In contrast to expert systems, such as the institution's current matching solution, these methods may bypass the creation and maintenance of large ontologies, which may be rendered obsolete by the evolution of the labor market, and alleviate the difficulty of defining and weighting the importance of different aspects of match quality (distance, wage, occupation) based on expert knowledge only. Our main questions of interest are: how can past observed data be leveraged to design efficient job recommender systems, and could they ease matching on the labor market?

Challenges Given a performance metric to optimize (*e.g.* the recall indicator on hiring data), the design of job search recommender systems faces several challenges [Mas+22]. A first challenge is related to the public availability of relevant datasets, which has been key to breakthroughs in other application domains by enabling the benchmarking of machine learning approaches (*e.g.* ImageNet for computer vision). However, concerns over privacy and commercial interests hinder the publication of job recommendation datasets. The CareerBuilder dataset, as well as the RecSys 2016 and 2017 challenge datasets [Abe+16; Abe+17] constitute noteworthy exceptions, although the RecSys datasets are no longer or only partially available. Moreover, datasets associated to different application contexts vary in terms of the nature, source, quality and sparsity of gathered information on job seekers (nothing at all, structured administrative data, unstructured text), job ads and interactions (clicks, applications, hires). Thus, algorithms best suited to a given application may be inapplicable when transposed to another setting. We will focus on the application setting of the French PES, in which (largely tabular) administrative data exist on job seeker and job ads, but interactions are sparse at the applications and hiring level. A second challenge is linked to the so-called *cold start* issue, denoting the case in which few, if any, interactions are observed for a sizable share of job seekers. In this setting, classical approaches to recommendation, such as collaborative filtering algorithms relying on interaction history, may be inefficient or inapplicable. Yet cold start is the norm in job recommendation: one expects job seekers to only stay on a PES's platform until they find a job. In the *France Travail* data, sparsity of the interaction matrix is of the order of 10^{-8} at the hiring level, and 10^{-7} at the application level. Contextual data thus has to be leveraged

to generate recommendations. A third challenge is posed by the multi-faceted nature of the data: relevant information about job seekers and ads come under many forms, *e.g.* location, structured administrative data, text of CVs, and may require standardization [ASL23]. These sources of information must be merged seamlessly to be leveraged in the recommendation process. A fourth challenge is related to scalability: a suitable job recommendation algorithm in the PES's setting must be able to recommend tens of thousands of job ads to hundreds of thousands of job seekers in quasi real time.

Beyond efficiency in the sense of recall, issues specific to the labor market setting plague job recommender systems. The first is related to the question of *value alignment*. Many possible objectives exist for machine-learned recommender systems - *e.g.* reproducing clicks, applications or hiring behavior. Other expert systems, widely used in national PESs, focus on explicitly satisfying job seekers' search criteria. Do these approaches yield similar results? How well do they align with job seekers' best interests, and can we design approaches even better aligned with those? A second challenge is that job ads are *rival goods*: only a single or a few job seekers may be hired to fill a given job opening. The popularity bias in recommender systems [BCC17] is a well-documented tendency to over-recommend already popular items, ignoring the "long tail" of less frequently seen ones, and creating "winner-takes-all" effects. Already problematic when recommending non-rival goods, this tendency is a major concern in labor market settings [Mas+22], where congestion effects have been noted to hamper the effect of real-world policies [Cré+13; Alt+22]. Accordingly, congestion must be monitored, and, if necessary, be reduced - although such interventions may come at the cost of recommendation relevance as measured by standard indicators. A third challenge lies with respect to *fairness*: while job recommendation constitutes a high-stakes application of machine learning (*e.g.* with respect to the European AI Act), recommender systems risk reproducing or worsening discriminatory or unwanted biases when learning from real-world data.

Contributions A first contribution is Muse (MUlti-head Sparse E-recruitment), a job recommender system learned from hires, designed with a concern for scalability and performance in the cold start setting. Muse adopts a two-tiered architecture. A first tier dedicated to candidate selection adopts a "two-tower" structure, in which separate embeddings for job seekers and ads, designed to incorporate domain knowledge, ensure the approach's scalability. The second tier then leverages a more elaborate model and features to re-rank the job ads short-listed by the first tier. Muse is empirically validated on the proprietary *France Travail* data and on public (RecSys 2017) challenge data [Abe+17]. On the *France Travail* dataset, Muse is shown to outperform boosted tree ensembles inspired from the 2017 RecSys challenge winners [CG16; VYP17a] in terms of both performance and scalability. On the RecSys 2017 challenge dataset, Muse is also shown to outperform a state-of-art approach to cold start job recommendation [VYP17b]. Muse is also compared to the state of the art and to *France Travail*'s current expert system in large-scale randomized control trials in the field. Explicit and implicit measures of users' satisfaction regarding recommendations are shown to favor Muse over the selected benchmarks.

A second contribution is a discussion of value alignment in the job recommendation setting. We document that different algorithm designs empirically result in very different rankings. Based on a formal economic model, we discuss the merits and limits of different plausible approaches to recommendations (satisfying job seekers' exact requirements, optimizing for applications and hires). An algorithm aligned with job seekers' objective would combine the utility they derive from a job and their hiring probabilities to rank job ads, although estimating these elements from observed data is not trivial.

A third contribution is related to congestion in job recommendation. We provide empirical evidence that congestion may be a concern in the PES's setting. Leveraging tools from the computational transport literature, we propose a post-processing approach to congestion-avoiding recommendation, named CAROT. Its performance is assessed on the *France Travail* data and on a public dataset provided by [Li+19]. The terms of the trade-off between congestion and standard recommendation performance are investigated.

Our fourth contribution is an audit of MUSE in terms of gender gaps with respect to performance (recall) and to the characteristics of recommended job ads (*e.g.* wage, contract type) *w.r.t.* gender. In terms of recommendation performance, the algorithm is shown to have slightly stronger performance for women than for men, the difference being statistically significant. Drawing inspiration from the labor economics literature on the gender wage gap [Kit55; Oax73; Bli73], we propose measures for gender gaps in recommendation characteristics, proposing the possibility to control for job seekers' qualifications and preferences. In terms of job ad characteristics, the algorithm is shown to reproduce, but not increase, gaps found in its training data (namely hires). We propose a novel post-processing approach to reduce unconditional or conditional gaps, and illustrate the trade-offs it entails between recall and (conditional) gender gaps.

The algorithm's design and evaluation will eventually contribute to a growing literature in economics on the effects of personalized labor market recommendations [BKM19; Alt+22; Beh+22; LHR23].

Thesis outline The first part of this work, based on [Bie+23c], studies efficient (in the sense of standard relevance measures such as recall) job recommendation when learning from sparse interactions. Chapter 1 reviews related work: after providing a brief overview of recommender systems based on implicit feedback, we present state of the art approaches to job recommendation. Chapter 2 presents the data collected by *France Travail* about job seekers, job ads and their interactions which is leveraged throughout this work, in order to motivate downstream design choices and discuss some of its limitations. Chapter 3 is dedicated to the proposed approach to sparse job recommendation: the MUSE recommender system. The approach's two-tiered design is presented in section 3.1. In section 3.2, MUSE is comparatively assessed on the *France Travail* and RecSys 2017 challenge datasets. Ablation studies are provided to assess the importance of key choices in the architecture's design. The second part of the thesis is dedicated to problems concerning job recommendations beyond accuracy. Chapter 4, based on a working paper in economics [Bie+23a], questions the design of job recommender systems from the

point of view of value alignment. Leveraging two plausible recommender systems (MUSE and an expert system), we show that different recommendation objectives lead to very different rankings, and discuss the objectives that job recommender systems should aim to optimize in the context of a formal economic model. Chapter 5, based on unpublished work, describes the results of two field experiments gathering job seekers' assessments of a variety of job recommender systems - based on hires (MUSE but also the state of the art), on applications, and the institution's expert system. Chapter 6, based on [Bie+21], is dedicated to the issue of congestion: a post-processing approach to congestion-avoiding recommendation, CAROT, is proposed and comparatively assessed on the *France Travail* and public [Li+19] data. Chapter 7, based on [Bie+23b], describes our proposed methodology to measure gender gaps in recommendations, provides an audit of MUSE with respect to those, and compares estimated gaps in recommendations to those observed in hiring and application data. Results from a post-processing approach to mitigate gender gaps are also presented. A general conclusion outlines perspectives for further research.

Recurring notations

This section briefly summarizes recurring notations and abbreviations. Given a $n \times m$ matrix M, we denote $M_{i,.}$ its i-th row, and $M_{.,j}$ its j-th column.

General setup

Notation	Meaning
I	Set of all users (job seekers)
i	User / job seeker index
J	Set of all items (job ads)
j	Item / job ad index
n	Number of users / job seekers
m	Number of items / job ads
x_i	Job seeker i's characteristics in a domain $\mathcal{X}$
y_j	Job ad j's characteristics in a domain $\mathcal{Y}$
M	Interaction matrix between job seekers and items
$\mathcal{P}(i)$	Set of job seeker i's positive interactions
k	Number of recommendations in a list
$r_{ij}, r(i,j)$	Rank of job ad j for job seeker i (according to a model)
$s_{ij}, s(x_i, y_j)$	Score used to rank job ads j for job seeker i

Algorithms

Notation	Meaning
MUSE.0	First tier of MUSE
MUSE.1	Second tier of MUSE (leverages hires only)
MUSE.1.Applications	Second tier of MUSE (learning from applications only)
MUSE.2	Second tier of MUSE (learning from hires & applications)
SDR	*France Travail*'s current expert system
PBS	Home-made expert system inspired by SDR
XGB	Boosted tree ensemble (XGBoost, [CG16])
MIX	Mixture algorithm combining a MUSE variant and PBS

MUSE hyperparameters (Chapter 3)

Notation	Meaning
η	Margin size in margin loss
ϕ	Job seeker embedding function
ψ	Job ad embedding function
$Var(x, y)$	Pair-wise features between x and y

Value alignment (Chapter 4)

Notation	Meaning
$U(x, y)$	Job seeker's utility
$\mathcal{U}(i, j)$	Proxy for U, based on a PBS variant
$p(x, y), p(i, j)$	Hiring probability conditional on application
$\mathcal{P}(i, j)$	Calibrated proxy for hiring probabilities scores based on MUSE.0
$\mathcal{PU}(i, j)$	$\mathcal{U}(i, j) \times \mathcal{P}(i, j)$
$\pi(x, y)$	Job seeker's perception of $p(x, y)$
A_{ij}	Whether job seeker i applies to job j
M^*_{ij}	Whether a match takes place if i and j meet ($M_{ij} = M^*_{ij} A_{ij}$)
c	Job seekers' cost of submitting an application
r	Cost of an application being rejected

Congestion-avoiding recommendation (Chapter 6)

Notation	Meaning
g	Monotonous function defining transport costs
C_{ij}	Cost of recommending j to i; $C_{ij} = g(s_{ij})$ or $g(r_{ij})$
ε	Entropic penalization weight in Sinkhorn's algorithm
μ	Uniform distribution on the n job seekers
ν	Uniform distribution on the m job ads
γ	Assignment plan (from OT)

Gender gaps (Chapter 7

Notation	Meaning
G	Gender (equals one if the job seeker is a woman)
X	All of job seekers' characteristics
Z	Job seeker characteristics on which to condition ($Z \subset X$)
Y	Job ad characteristic of interest (*e.g.* recommended wage)
δ	Average (unconditional) gender gap
τ	Gender gap conditional on characteristics
$m_g(z)$	$\mathbb{E}[Y \vert Z = z, G = g]$
$e_g(z)$	$\mathbb{P}[G = g \vert Z = z]$
γ	Assignment plan (from ILP)
w_{ij}	Characteristics of recommendations defining ILP constraints

Miscellaneous abbreviations

Notation	Meaning
AIPW	Augmented Inverse Propensity Weighting
AUC	Area Under the (ROC) Curve
DPAE	*Déclaration Préalable à l'Embauche* (administrative data on hires)
ILP	Integer Linear Program
IPW	Inverse Propensity Weighting
LTR	Learning to Rank
ML	Machine Learning
MLP	Multi-Layer Perceptron
NDCG	Normalized Discounted Cumulative Gain
OT	Optimal Transport
PES	Public Employment Service
RCT	Randomized Control Trial
RMSE	Root Mean Squared Error
ROME	*Répertoire Opérationnel des Métiers et Emplois* (job ontology)
RS	Recommender System
SVD	Singular Value Decomposition
TF-IDF	Term Frequency Inverse Document Frequency

Part I: Toward accurate sparse job recommendation

Chapter 1

Related work

Recommender systems seek to help users to navigate and engage with large sets of items. Such navigation is unprompted, in contrast to the query-based systems studied in Information Retrieval.

Recommender systems emerged as a theme in research and applications in the 1990s. Early examples include Tapestry [Gol+92], which filtered e-mails based on community annotations using a query language (coining the term "collaborative filtering"), and GroupLens [Kon+97], which recommended news articles using user similarities defined based on rating Pearson correlations. Key ideas of collaborative filtering, such as the memory-based user-based or item-based algorithms, as well as model-based approaches leveraging matrix factorization [Sar+00], were proposed during this decade. Interest in the domain was fostered by the 2006-2009 Netflix challenge [BK07], and prompted the creation of dedicated conferences such as the ACM Conference on Recommender Systems (henceforth RecSys) from 2007 onward. Recommender systems were also affected by the gain in prevalence of neural networks in machine learning in the 2010s. Today, they feature among the most ubiquitous applications of machine learning, as they are present at the core of the services of information-economy behemoths such as Amazon (online shopping), Netflix (movies), Google News (news articles), Youtube (videos), Facebook (friends, content and news), Tripadvisor (hotels) or Spotify (music).

Subsection 1.1 sets up basic vocabulary regarding recommender systems, focusing on the case of implicit feedback since the data on interactions we will consider throughout this work (applications, hires) is of such nature [1]. Subsection 1.2 reviews some state of the art approaches for job recommendation, as well as some field-specific challenges.

1.1 Recommender systems (with implicit feedback)

This section begins by a discussion of the goals and evaluation of recommender systems. We then proceed to high-level families of approaches to recommendation,

[1] For a broader view of recommender systems, we refer the reader to the textbook [Agg16], especially for its extensive coverage of collaborative filtering, and to [Zha+19] for a survey of pre-2019 approaches relying on neural networks.

and to discuss a few common recommender system architectures, before presenting links between recommendation and the *learning to rank* problem.

Set-up and notations This section uses the following setup and notations. We observe a set $I = \{1, \dots n\}$ of n users with context $x_i \in \mathcal{X}$ for each user i, and a set $J = \{1, \dots m\}$ of m items with context $y_j \in \mathcal{Y}$ for each item j. For a subset of user-item pairs $I \times J$, we observe interactions M_{ij} between users and items. These interactions may be explicit - in the sense that user i rated item j on a continuous or ordinal scale - or implicit (*e.g.* a click, application, hiring between i and j). Our exposition will focus on the implicit feedback setting due to the nature of the data encountered in this work (applications, hires)[2]. We denote $\mathcal{P}(i) = \{j | M_{ij} = 1\}$ the items with which i has positive implicit interactions. Given a user i, a recommender seeks out to output a list of k recommended items (with k a fixed small integer).

1.1.1 Goals and evaluation of recommender systems

Goals of recommender systems Recommender systems may strive to achieve several objectives from the point of view of their end user [Agg16; Her+04]. The primary one is item relevance, *i.e.* recommending items an individual finds useful - although usefulness is not always trivial to define and measure ([KMR23]; more on what this may mean for job recommendation in Chapter 4). Secondary objectives include novelty, serendipity, and recommendation diversity. Multi-stakeholder settings may pit the interest of the user against that of items (e.g. recruiters), those of the platform, of other users (if items are rival goods) and society as a whole. In the following, we focus on the evaluation of whether a given recommender system performs well in terms of item relevance.

"In the field" evaluations Randomised control trials (RCTs), also called A/B tests, randomly assign users to a treated and a control group which differ by the version of the recommender system they are exposed to. Under standard assumptions [IR15], the two groups may be compared to assess the causal impact of the treatment (*i.e.* the impact of switching versions of the system) on metrics of interest. These metrics of interest can be defined in terms of reactions to the recommendations (*e.g.* ratings, clicks on the recommended job ads) or in terms of downstream user behavior (*e.g.* engagement with a website, job search behavior, speed of return to employment). However, randomized control trials may be costly, lengthy to organize, run the risk of affecting user experience, and may not always feasible (*e.g.* for practical or ethical reasons).

When the key quantity of interest (*e.g.* satisfaction with the recommendations) can be provided by users, surveys can also be used to measure user satisfaction and contrast several versions of recommender systems. Surveys nonetheless come with issues of their own: they may inconvenience users; user utility may not always be easy to elicit, and selection bias in survey responses must be accounted for.

[2]Most of the discussion in this section can easily be adapted to the case where $M_{ij} \in \mathbb{R}$. Ordinal data may however call for more elaborate modeling.

Offline evaluations A recommender system's quality may also be assessed on logged data, based on its ability to generate rankings in which positive interactions are ranked above non-positive ones on a test set. Assume access to test set of pairs, $S^{test} \subset I \times J$, that has not been used for training. Let n_{test} denote the number of users present in the test set, and, for a test set user i, $\mathcal{P}^{test}(i) = \{j|(i,j) \in S^{test}, j \in \mathcal{P}(i)\}$ be the set of pairs corresponding to i's positive interactions in the test set, and $I^{test}(i) = \{j|(i,j) \in S^{test}\}$ be all test set items in pairs involving i. Let $r(i,j)$ denote the rank of item i for job seeker j among $I^{test}(i)$ according to the assessed recommender system. The *recall@k* [3] may be defined as:

$$\text{recall@}k = \frac{1}{n_{test}} \sum_i \frac{\sum_{j \in \mathcal{P}^{test}(i)} \mathbb{1}\{r(i,j) \leq k\}}{\min(|\mathcal{P}^{test}(i)|, k)}$$

If $|\mathcal{P}^{test}(i)| = 1$ for all i (an rough approximation for hiring data), the recall@k is simply the share of job seekers for which the algorithm correctly ranks their future job among the top k recommendations among the test set's job ads. Due to its intuitive nature, the recall will be the main performance metric used in the following. Other ranking quality measures including the Normalized Discounted Cumulative Gain (NDCG)[4], Mean Reciprocal Rank, Mean Average Precision, Expected Reciprocal Rank, Kendall's tau, and Spearman's rho.

As we shall see, rather than optimizing the metric of interest (*e.g.* recall) directly, it is common to turn to surrogate learning problems. For instance, one may learn a classifier yielding an estimate of interaction probability given job seeker and job ad features $\mathbb{P}(M_{ij} = 1|x_i, y_j)$ and rank items according to their predicted interaction probability given the context of user i. In such cases, standard classification metrics, such as the area under the Receiver Operating Characteristic (ROC) curve, may also be used to assess the algorithm's performance.

Two remarks are in order. First, if an item j is not part of $\mathcal{P}(i)$, this may either mean that i) i was aware of j's existence (*e.g.* j was shown to i on a web page) but i didn't interact positively with it; or ii) that i was never exposed to j and possibly unaware of j's existence. The principle of revealed preferences justifies rankings items in case i) lower than those in $\mathcal{P}(i)$. How to treat items in case ii) is less clear, especially since the set of items a person is exposed to typically depends on a prior logging strategy (which may not always show job seekers the most relevant job ads). Accordingly, it is debatable whether performance measures should be measured among proven interactions, or among all test set pairs. Second, we can not compute performance metrics for users for which we observe no interactions, *i.e.* $\mathcal{P}(i) = \emptyset$. When working with hiring data, we may not observe hires for some

[3] This information retrieval definition differs from recall in the classification setting, defined as the ratio of true positives over the sum of true positives and false negatives. The two notions may be reconciled if one considers a ranking model which ranks job ads j based on some score s_{ij}, and consider "positive" predictions to be those in the top-k list (*i.e.* job ads above an individual-specific threshold leaving k of them in the list).

[4] The NDCG, which weights the placement of correct items by a logarithmic factor proportional to their position in rankings, is frequently used in the literature. In early experiments in the development of Muse, comparison of algorithms based on their recall were consistent with comparisons in terms of NDCG, leading us to focus on recall due to its simplicity.

individuals; and these individuals are likely to differ from hired job seekers (more in Chapter 2). Thus, even if we define relevance as a ranking of job ad by hiring likelihood, using the recall directly as a proxy for relevance may lead to biased estimates in the sense that they inform us about model quality on the population of hired people only.

1.1.2 Types of recommender systems

To discuss the variety of existing recommender systems, we rely on a taxonomy proposed by [Bur02], reproduced in Table 1.1, which is based on the background, inputs and recommendation recommender systems leverage.

Technique	Background	Input	Process
Collaborative filtering	Ratings M_{ij} for all I	Ratings M_{ij} for user i	Extrapolate from ratings of users in I similar to i
Content-based	Features of items in J	i's rating of items in J	Generate a classifier that fits i's rating behavior and uses it on j
Demographic	Demographic information about I and their item ratings	Demographic information about i	Identify users demographically similar to u, and extrapolate from their ratings of j
Utility-based	Features of items in J	A utility function over items in J that describes i's preferences	Apply the function to the items and determine j's rank
Knowledge-based	Features of items in J + knowledge of how these items meet a user's needs	A description of i's needs or interests	Inter a match between j and i's needs

Table 1.1: Taxonomy of recommender systems according to [Bur02]

This taxonomy enables the authors to discuss the merits and weaknesses of the different techniques, as reproduced in Table 1.2.

What we wish to highlight from this taxonomy is that not all recommender systems leverage machine learning; and that the relevance of different approaches depends on available data, objectives and domain knowledge.

Hybrid recommender systems, which combine the recommendation logic of the different types of recommender systems described, strive to leverage the strengths of these different approaches. [Bur02] provide a taxonomy of these hybridation approaches, reproduced (up to minor changes) in Table 1.3.

For job recommendation, the taxonomy enables us to sharply contrast the collaborative filtering (leveraging ratings only) and utility-based / knowledge-based (leveraging only features and expert knowledge, without any learning) approaches from other ones. However, in many job recommendation settings, we observe some ratings (at least for past users), some user demographics and item features, all of which are relevant for generating recommendations. Thus, most common machine-learning based approaches attempt to leverage all three sources, and fall into the categories of demographic, content, and /or collaborative filtering-based hybrids.

Technique	Pluses	Minuses
Collaborative filtering	A) Can identify cross-genre niches; B) Domain knowledge not needed; C) Adaptive (quality improves over time); D) Implicit feedback sufficient	I) New user ramp-up problem; J) New item ramp-up problem; K) “Grey sheep" problem; L) Quality dependent on large historical dataset; M) Stability vs plasticity problem
Content-based	B, C, D	I, L, M
Demographic	A, B, C	I, K, L, M; N) Must gather demographic information
Utility-based	E) No ramp-up required; F) Sensitive to changes of preference; G) Can include non-product features	O) User must input utility function; P) Suggestion ability static (does not learn)
Knowledge-based	E, F, G, H) Can map from user needs to products	P, Q) Knowledge engineering required

Table 1.2: Tradeoffs between recommender systems by [Bur02]

Hybridation method	Description
Weighted	Combine scores / votes of several RS to produce a single recommendation list
Switching	Switch between RS depending on situation / context
Mixed	Present recommendations from several RS at the same time
Feature combination	Features from different RS data sources (e.g. demographics augmented with descriptions of users’ past clicks) are thrown into a single recommendation algorithm
Cascade	One RS refines the recommendations given by another
Feature augmentation	Use output of a RS as an input for another
Meta-level	The model learned by a RS is used as input to another

Table 1.3: Taxonomy of hybrid methods by [Bur02]

1.1.3 Common approaches and architectures

After mentioning the links between recommender systems and learning to rank, this section reviews prominent approaches to recommendation based on collaborative filtering and contextual information before briefly discussing strategies used in practice to ensure scalability.

Recommendation and learning to rank In essence, recommendation is a *learning to rank* (LTR) problem [Liu09; Bur10]: what matters is not predicted values at the pair level, but the rankings provided to users, and especially the top of these rankings. Since ranking quality measures (*e.g.* recall or NDCG) are not trivial to optimize directly, one typically turns to auxiliary learning problems. Different flavors of learning to rank exist, set up as different learning problems for a scoring function $s(i, j)$ aimed at ranking items. In the *pointwise* approach to LTR, recommendation is viewed as a *missing value prediction* problem: among unseen pairs, what would be the ones with highest predicted ratings $s(i, j)$ if presented to the users? Pointwise LTR attempts to predict the relevance of item j for user i by predicting whether M_{ij} is equal to one given user and items, hence learning a scoring function $s(i, j)$ such that $L(s(i, j), m_{ij})$ is low, with L a loss function (*e.g.* binary cross-entropy, mean squared error). This simply follows the so-called probability ranking principle in information retrieval [Rob77]: documents should be ranked in order of the probability of relevance or usefulness. In the *pairwise* approach to LTR, one attempts to predict whether item j ranked higher than item j' for user i. For instance, one may learn s to approximate $\mathbb{P}(M_{ij} > M_{ij'})$ by $\sigma(s(i, j) - s(i, j'))$ where σ is the logistic CDF, using a binary cross entropy loss (see *e.g.* RankNet [Bur+05] as an example). Various learning methods, losses (contrastive, triplet, margin), and insights from (deep) metric learning (see [Kul13; BHS22; KB19] for reviews of metric learning, and for instance [ML10; Hsi+17] for applications to recommendation) can be leveraged in the same spirit. In the *listwise* approach to LTR [Cao+07], one directly tries to optimize the value of evaluation measures (*e.g.* NDCG) on all queries of the training data. As of the writing of this thesis, the pointwise and pairwise LTR approaches still seem the most prevalent in practice for recommender systems.

Collaborative filtering We define collaborative filtering in the narrow sense of recommender systems that only rely on ratings M_{ij} to carry out recommendations. For training purposes, we will focus on one-class recommendation setting - that is, filling in $M_{ij} = 0$ when it is not known if i viewed j in the past -, since it is the most relevant to the labor market setting. Indeed, it is likely that a job that a user applied to (or that is a user's future position) is more relevant than a job ad selected purely at random.

The simplest collaborative filtering methods are *memory-based*, or *neighborhood-based* ones:

- Item-based: if you liked item j, recommend item j' because item j is "close"

to item j'. Rankings are then based on predicted scores

$$s_{ij} = \frac{\sum_{j'} \text{sim}(j', j) M_{i,j'}}{\sum_{j'} |\text{sim}(j', j)| \mathbb{1}\{M_{i,j'} > 0\}}$$

where $\text{sim}(j', j)$ is a similarity measure between two items j and j'. A typical choice for $\text{sim}(j', j)$ is the cosine similarity: in the binary case, the number of users who clicked on both j and j', normalized by the product of the square roots of the number of clicks on both job ads.

- User-based: you will like item j because users "like you" (in the sense of having clicked on similar items as you) also liked item j

$$s_{ij} = \frac{\sum_{i'} \text{sim}(i, i') M_{i',j}}{\sum_{i'} |\text{sim}(i, i')| \mathbb{1}\{M_{i',j} > 0\}}$$

where $\text{sim}(i, i')$ is a similarity measure between i and i'.

One can also consider *model-based* approaches to collaborative filtering, *i.e.* attempt to model M_{ij} based on observed ratings. Singular Value Decomposition (SVD) type approaches resort to a learning problem of the type:

$$\min_{l^{\text{user}}, l^{\text{item}}} \sum_{(i,j) \in I \times J} (M_{ij} - (l_i^{\text{user}})^T l_j^{\text{item}})^2 + \lambda(||l^{\text{user}}||^2 + ||l^{\text{item}}||^2) \tag{1.1}$$

where l^{user} and l^{item} are latent, low-dimension vector representations of users and items respectively, and $\lambda > 0$ is a regularization factor introduced to avoid overfitting. The link with SVD [Sar+00] is the following. Any $n \times m$ matrix M can be decomposed in a singular value decomposition $M = U\Sigma V^T$ where $U \in \mathbb{R}^{n \times n}$ is an orthornormal matrix (*i.e.* $UU^T = I$), $\Sigma \in \mathbb{R}^{n \times m}$ is rectangular diagonal (with diagonal entries by convention sorted from high to low), and $V^T \in \mathbb{R}^{m \times m}$ is orthornormal. A rank-r approximation of M, with $r \leq \min(m, n, rank(M))$, can be proposed as

$$\tilde{M} = U_{1:n,1:r} S_{1:r;1:r} V^T_{1:r,1:m}$$

This approximation is optimal in terms of Froebenius norm, defined for a matrix M as $||M||_F = \sqrt{\sum_{i,j} M_{ij}^2}$, in the sense that:

$$\tilde{M} \in \arg \min_{M', \text{rank}(M') \leq r} ||M - M'||_F$$

When $\lambda = 0$, a solution to the optimization problem in equation 1.1 would then be

$$l^{user} = (U_{1:n,1:r}(S_{1:r;1:r})^{1/2}), \quad l^{item} = ((S_{1:r;1:r})^{1/2} V^T_{1:r,1:m})$$

Variants of model-based matrix factorization approaches also include non-negative factorization and margin factorization.

Bayesian Personalized Ranking from implicit feedback [Ren+12] instead takes a pairwise stance, trying to rank positive pairs above missing values or viewed-but-not-clicked interactions in a Bayesian setting.

Neural network-based approaches generalize matrix factorization approaches to generate and merge user and item embeddings in a learned fashion rather than through a dot product. For instance, Neural Collaborative Filtering [He+17], taking as input a one-hot representation of users $x_i \in \{0,1\}^n$ and items $y_j \in \{0,1\}^m$, seeks to predict interactions M_{ij} through a score parametrized as

$$s_{ij} = f(P^T x_i, Q^T y_j)$$

where P^T and Q^T are learnt embeddings, and f is a multi-layer feed-forward neural network. The gains compared to matrix factorization remains under discussion [Ren+20], and should be considered in light of added computational costs compared to matrix factorization.

A key limit of collaborative filtering methods is their weakness in the *cold start* setting: that is, when no previous interactions exist for job seekers or job ads. This is the rule rather than the exception in a labor market setting, where interaction history is typically rather short. The issue becomes only worse if one wishes to work with hiring data rather than clicks or applications.

Feature-based recommendation While collaborative filtering-based methods leveraged only interactions M, let us define feature-based recommender systems as recommender systems that rely on user features x_i and item features y_j, merged together, and perhaps provided with feature augmentation from *e.g.* past user and clicks, into a joint representation z_{ui}, to generate recommendations[5].

When both x_i and y_j are descriptions of users and items in natural language, they may be represented in a joint embedding space in an unsupervised fashion. For instance, one can learn a SVD on the concatenation of the multi-hot or TF-IDF representations of resumes and job ads, and recommend to a job seeker the job ads closest to his resume in the latent space. The issue, however, is that the recommendations' quality depends on the quality of distances in the latent space. As emphasized by [Sch+17], job seekers and job ads "do not speak the same language". For instance, a job announcement may read "You take in charge the physical reception of the patients, the management of the planning and the patient records" whereas a resume may read "Secretary accountant; Capture and storage of documents". Due to such differences in vocabulary, job seekers' representations are not guaranteed to be close to those of relevant job ads in a latent space constructed without supervision.

More promising approaches to recommendations try to leverage both user features and item features z_{ij}, as well as past interactions M_{ij}, to learn a score $s(z_{ij})$ by which job ads j may be ranked for a given user i. For instance, one may fit a generalized linear model to fit past interactions M_{ij} based on features z_{ij} (possibly augmented with feature engineering), and rank job ads by predicted $\hat{M}_{\cdot,j}$ for a

[5]This terminology is introduced to avoid the ambiguity with respect to common definitions in the literature. What we call feature-based recommendation amounts to what [FC21] call content-based recommendation; yet "content-based recommendation" is defined otherwise in the taxonomies of [Bur02] or [Agg16]. The delimitation between feature-based recommendation and collaborative filtering remains somewhat vague: consider the extreme case discussed above where $x_i = \{0,1\}^n$, $y_j = \{0,1\}^m$ are one-hot representations of users and items.

given user context. More recently, the Wide & Deep architecture [Che+16], implemented at Google Play, seeks to improve on performance when user-item inputs are sparse by leveraging deep neural networks with embeddings, learning a deep network to learn user-item interactions for sparse features, while retaining a wide linear model component to combine the benefits of memorization and generalization. The DeepFM architecture [Guo+17] proposes an architecture combining matrix factorization and deep layers to provide for both low and high order feature interactions. Similarly (though initially proposed for pure collaborative filtering), [He+17] combine MLP layers and a Generalized Matrix factorization layer (*i.e.* an elementwise product of user and item embeddings) in a deep neural recommender system. In any case, as highlighted by [Beu+18; Jay+20], the key question is how to learn good representations of sparse features; a key intuition is that multiplicative interactions might help in that context.

Scalability Scalability is paramount to handling real-world problems with millions of users and items. The popularity of models relying on neural or SVD-based embeddings, asides from their performance, relies on their compatibility with approximate closest-neighbors or maximum inner product search methods [Mat+18]. A canonical structure for large-scale recommendation has thus emerged to balance speed and efficiency: in a first stage, a "two-tower", embedding-based model selects promising hundreds or thousands of job ads ("candidate retrieval") using (possibly approximate) nearest neighbor search; these candidates are re-ranked in a second stage, with a model potentially leveraging more elaborate features and architecture.

1.2 Job recommender systems

The state of the art in job recommendation has been surveyed by [FC21; DB21; Mas+22]. Before delving into a short review of job recommendation models, it is worth noting some challenges job recommendation entails [Mas+22]: short interactions history and sparsity, and multi-faceted data on job seekers and ads, pose an important challenge; both suitability (of a job seeker profile to a job posting) and preferences matter in order to generate relevant recommendations (and both notions are multi-faceted); job ads are rival goods (*i.e.* have capacity constraints); and job recommendation is a high-risk domain, leading to fairness and trustworthiness concerns. We shall presently focus on works focusing on efficient job recommendations with respect to standard metrics. Aspects related to the balance of suitability and preferences, to the rival-goods nature of job recommendation and to fairness issues will be reviewed in Chapters 4, 6 and 7 respectively.

Knowledge-based recommender systems Also known as expert systems, these approaches, which represent 12.7% of the published works surveyed by [FC21], generate recommendations based on expert knowledge encoded in detailed ontologies (*e.g.* of jobs, skills, contracts) and measures of distance between their entities. As a leading example, the WCC ELISE Smart Search&Match solution, which matches job seekers and job ads based on explicit requirements (in terms of job type, wage,

skills ...), is used by several national Public Employment Services (France, Flanders, Germany) as well as by private actors such as Robert Half. These approaches have undeniable merits. Since they provide job seekers (almost) exactly what they require, their recommendations avoid the risk of utter irrelevance, which can harm user experience and a public institution's image. Since their data requirements are limited to a job seeker and the job ads' description, they are robust to cold start. Moreover, they are often interpretable by design, and raise fewer concerns about privacy and fairness than systems learned to reproduce past data. However, defining ontologies and relations between their entities (*e.g.* job similarities) requires expertise and constant maintenance to account for a shifting labor market. Finding a simple model specification that covers all possible individual situations while remaining truthful to experts' intuitions is extremely challenging. The approach's output is also extremely brittle with respect to the many input parameters that must be manually specified, *e.g.* the respective weights and definition of occupation, skill, or education similarities. If the approach implements filters to avoid showing irrelevant job ads, a risk always exist of finding no jobs at all to show to the user.

Collaborative filtering Collaborative filtering represent 6.35% of the work surveyed by [FC21]. For instance, a matrix factorization approach akin to the SVD-based model presented above (equation 1.1), specialized to implicit ratings by [HKV08], was implemented at the Swedish PES and studied by [LHR23]. Another example is Indeed's recommender system, which - at least in 2016 - leveraged user-to-user collaborative filtering, implemented with Apache Mahout. Graph-based learning methods for link prediction may also be leveraged - for instance, [Mas+23] use Conditional Network Embeddings [KLD18] as a backbone recommendation method on datasets provided by VDAB (the Flemish PES) and CareerBuilder. However, the application of collaborative filtering methods remains limited to settings where sufficiently large interaction histories are observed for users.

Feature-based recommender A large portion of the job RS literature (forming 26.98% of [FC21]'s surveyed papers, to which can be added the bulk of the 33.33% classified as "Other types" by the authors) focuses on feature-based recommender systems. LAJAM [Sch+17; Sch18] uses natural language inputs describing both job seekers and job ads, and past interactions, to generate embeddings for job seekers and job ads using a Siamese network and a contrastive loss (resulting in a two-tower structure). Similarly, Randstad's talent recommender [Lav21] fine tunes embeddings of CVs using job transition history, using a multi-lingual bi-encoder BERT model. CareerBuilder's recommender system [Zha+21] fuses embeddings for CVs from raw text (learned on applications), skills parsed from raw CV text (based on job transitions and skill co-occurrences, leveraging a job-skill interaction graph), as well as location for candidate retrieval in a two-tower structure, leveraging further contextual features in a re-ranking stage.

Due to a focus on settings when interactions are sparse and cold start is prevalent (as the *France Travail* data which will be presented in Chapter 2), our proposed approach, Muse, takes the form of a feature-based recommender. As [Zha+21],

Table 1.4: Examples of different recommender systems

References	Setting	Knowledge-based	Collaborative Filtering	Feature-based	Target variable
WCC Elise	National PESs, Robert Half	x			
[LHR23]	Swedish PES		x		Clicks
[Zha+21]	CareerBuilder			x	Applications
[Shi+22]	LinkedIn		x	x	Applications, "save"
[VYP17a]	Xing challenge		x	x	Mainly impressions, clicks
[Ma+22]	Indeed	x	x	x	Clicks, Applications

its closest neighbor in the literature, MUSE adopts fused embeddings for candidate retrieval, with a more flexible representation of geographic information. In the re-ranking stage, instead of a linear specification, MUSE leverages a flexible architecture incorporating multiplicative interactions, and combines information from applications and hires in its learning process.

Hybrids Hybrid job recommender systems leverage characteristics of several of the previously described approaches. Since both contextual information and collaborative filtering data are often available, hybrids between feature-based recommendation and collaborative filtering are especially prevalent. The approach of the RecSys 2016 challenge winners [Xia+16] relied on an ensemble of boosted trees and Hawkes processes, with a feature engineering strategy including both content and behaviorial information. Similarly, the RecSys 2017 challenge winners [VYP17a] predict positive interaction using boosted tree ensembles, leveraging user and item features as well as pair-wise features comparing ranked job ads to those with which the job seeker interacted in the past. LinkedIn's recommender system predicts matches using a large-scale generalized linear model, adding personalization by the inclusion of individual-level and recruiter-level fixed effects when sufficient interactions exists [Zha+16; Ozc+19; Shi+22]. Dropoutnet [VYP17b], used as benchmark in Chapter 3, propose a general, simple methodology for the hybridization of content-based recommendation and collaborative filtering to handle cold start. The method assumes a collaborative filtering (*e.g.* SVD-based) recommender system is available in the warm start case. The Dropoutnet architecture maps content and collaborative filtering in the same latent space, using dropout to learn the content-based latent to reconstruct the collaborative filtering latent, enabling generalization when the collaborative filtering input is missing. Various more complex hybridization strategies exist: for instance, Indeed post-filters recommendations (based on variants of collaborative filtering) using a mixture of rules based on expert knowledge and feature-based learning [Ma+22].

Reciprocal recommendation [YAÖ21] views online recruiting as a reciprocal recommendation problem. Their approach learns a bi-objective deepFM model [Guo+17], with two heads predicting applications and positive recruiter feedback. The multi-objective framework enables improvements (in terms of AUC and log-loss) for both heads compared to their standalone version, and reciprocal recommendation proceeds by considering a weighted sum of the two heads' outputs.

Chapter 2

Data description and analysis

This section introduces the data provided by *France Travail*, which will be used throughout this work. After introducing the institutional setting (section 2.1), we will describe the data available on job seekers (section 2.2), job ads (section 2.3) and their interactions (section 2.4), highlighting its wealth and limitations.

2.1 Institutional and geographical setting

France Travail (formerly *Pôle emploi*) is a French governmental agency which registers job seekers, helps them find jobs and provides them with assistance - in particular, unemployment benefits. *Pôle emploi* was created in 2007 as a merger of the ANPE, which was dedicated to counseling, and the ASSEDIC, which was in charge of unemployment insurance. As part of its missions, *France Travail* collects a wide range of information on job seekers, job ads, and labor market interactions.

In the following, we provide descriptive statistics relative to the Auvergne-Rhône-Alpes region of France from 2019 to mid 2022, which will be the setting of most of the present work. This focus on a single region was chosen in agreement with *France Travail* to facilitate data handling and experimentation. Auvergne-Rhône-Alpes was because of its size - with 8 million inhabitants in 2018, it is the second largest region in France - and the variety of the territories it comprises. Indeed, local situations range from urban centers in the Rhône department (in which Lyon, one of the three largest cities in France, is located) to medium-sized cities (Grenoble, Saint-Etienne, Clermont-Ferrand) and rural departments (*e.g.* Allier, Cantal). In 2018, according to INSEE, the region was slightly wealthier than the national average, with a median income of 22 480 euros (830 euros more than the national one), hiding considerable geographic heterogeneity (from 26 000 euros in Haute-Savoie to less than 20 600 in Allier and Cantal). The unemployment rate in Auvergne-Rhône Alpes was slightly lower than the national average in 2019 (7.3% against 8.4% nationally), with sizeable variation among departments (it ranges from 5.0% in Cantal to 9.3% in Ardèche and Drôme)[1].

While the results presented in Chapters 3, 5 and 7 were obtained in the described setting, those presented in Chapters 4 and 6 were obtained on different (and older)

[1]Source : Insee, taux de chômage localisés.

Chapters	Region	Time window	Size (job seekers, ads, hires)	Notes
3, 5, 7	Auvergnes-Rhône-Alpes	2019-mid 2022	1.3M, 2.2M, 258k	
4	Rhône-Alpes	2019	1.2M, 515k, 75k	Core results: transportation & logistics
6	Ile-de-France	2019	1.6M (spells), 477k, 43k	Core results: transportation & logistics

Notes: This table describes the datasets used in this work in terms of location, time window and sample sizes.

Table 2.1: Datasets

Job seekers' administrative status	Share
Immediately available, looking for full-time & indefinite duration	65.4%
Immediately available, looking for part-time & indefinite duration	8%
Immediately available, looking for definite duration job	15.8%
Looking for a job but not immediately available	4.8%
Looking for a job but already have a job	5.9%

Notes: This table describes the administrative status of job seekers in Auvergne-Rhône-Alpes from 2019 to mid 2022 (n=1 210 854).

Table 2.2: Categories of job seekers registered at France Travail

datasets, as recapitulated in Table 2.1.

2.2 Job seekers

Scope and institutional context Job seekers may register at *France Travail* if they are looking for a job, regardless of whether they currently have one. Registration is mandatory in order to receive unemployment benefits.

During the window of observation, we observe 1 210 854 unique job seekers. Since job seekers may leave and re-enter *France Travail* registries, potentially changing their search parameters, job seekers will be described at the level of their employment spells - defined as a period during which they are continuously registered at *France Travail* (allowing for temporary interruptions taking no longer than two weeks). 2 027 441 such unemployment spells are observed. In the following statistics, when job seekers have several unemployment spells, we describe the parameters of the most recent spell. On average, 427k job seekers are registered on a given week.

Table 2.2 describes the administrative categories of job seekers defined in terms of availability for starting a new job and of the type of job sought. Most job seekers are looking for a job and are immediately available (89.2%). The remainder is looking for a job but not immediately available, either because they already have a job (5.9%) or for other reasons (*e.g.* training, sickness or maternity leave). All job seekers will be considered for model training and evaluation: even though all of them may not be regular users of a recommender system in production, their labor market interactions should provide relevant data for training.

Data sources Job seekers that are immediately available and that claim unemployment benefits have a legal obligation to actively look for a job, and their search process is monitored by *France Travail*. When registering, they have to define with a caseworker the parameters describing what they would consider a "reasonable em-

ployment offer" (*Offre Raisonnable d'Emploi*) - that is, a job offer they would not turn down[2]. Job seekers may also define secondary job search parameters (*e.g.* in another location or another occupation), to help the institution provide recommendations. No legal obligations are associated to these secondary search parameters. Our main source of information on job seekers is extracted from the definition of the "reasonable employment offer" and secondary search parameters. All of these provide a desired occupation (in *France Travail*'s ROME[3] nomenclature), and requirements in terms of wage, geography, contract, part or full-time status, and number of working hours. Job seekers also provide basic socio-demographic information about themselves to the PES when registering.

France Travail also gives job seekers the possibility to showcase themselves and their skills on a platform (the *Profil de compétences*) where recruiters and caseworkers may browse a job seeker's profile (if it is published). Profiles on the platform roughly correspond to online resumes: they include a "business card" (or several of them) describing the job seeker as well as a more in-depth textual description, the provision of past experiences, of languages the job seeker speaks, driver's licenses he or she holds, and of skills. Skills may be provided in natural language or as entities in the ROME ontology, which provides a catalog of circa 12,300 standardized skills (*e.g.* "welding techniques", "tax system knowledge").

Since the use of the platform is non-mandatory, missing values abound: for instance, no "business card" exists for circa 55% of job seekers.

While textual data is sometimes available, it is often absent or low-quality, in contrast to the rich tabular data that is systematically collected on job seekers' administrative background and job search parameters. This situation differentiates *France Travail*'s setting from that of many online job platforms.

Descriptive statistics Tables 2.3, 2.4 and 2.5 report descriptive statistics on job seekers' qualifications and desired occupations, search parameters, and socio-demographic characteristics. 38% of job seekers had achieved tertiary education (while that was the case of 46.9% of the French population aged 25-64 in 2019[4]). 11% of job seekers are looking for executive positions. The most represented types of desired occupations are white-collar qualified (39%) or unqualified (18.7%) ones. 14.7% of job seekers seek jobs in retail and sales; 16.4% in "social, socio-educative and socio-cultural action" (37% of which are personal assistance jobs, *e.g.* care for

[2]Refusing two "reasonable" job offers may render job seekers ineligible to the reception of unemployment benefits. The "reasonable unemployment offer" is defined with respect to the job seekers' professional qualifications and skills. Job seekers do not have to accept offers that are part-time if they are looking for a full time job; nor offers with wages below the wage practiced in the region and occupation; nor offers that are in a job that is incompatible with their qualifications and skills. Thus, job seekers have incentives to declare job search parameters that correspond to jobs they would actually accept, but strategic considerations also come into play: if the parameters defining the "reasonable unemployment offer" are stringent, job seekers have more time and opportunities of considering alternative options without the threat of losing financial support.

[3]Répertoire Opérationnel des Métiers et Emplois. *France Travail*'s job ontology distinguishes 14 high-level sectors (*e.g.* "agriculture", "healthcare"), composed of 110 intermediate sectors (*e.g.* "woodcutting and pruning", "medical practitioner") and 531 detailed types of jobs. Each type of job is associated with a list of skills based on expert knowledge.

[4]Source: OECD France country profile of the OECD, Annex B.

Feature	Share
Highest level of education achieved	
5+ years of higher education	11.9%
3 or 4 years of higher education	11.5%
2 years of higher education	14.6%
General secondary education (“baccalauréat")	24%
Vocational secondary education (“CAP / BEP")	25.5%
Lower levels of education	9.2%
Missing	3.3%
Qualification	
Missing	5.2%
“Manoeuvres"	2.8%
Blue-collar, unqualified	3.9%
Blue-collar, qualified	8.7%
White-collar, unqualified	18.7%
White-collar, qualified	39%
Technician	5.9%
Blue-collar supervision	5%
Executive	11%
Desired occupation	
Agriculture, green spaces, animal care	3.4%
Craftsmanship	0.8%
Banking, finance, real estate	1.5%
Retail, sales	14.7%
Communication, media and multimedia	2.3%
Construction	7.6%
Hotels, restaurants, tourism, leisure, animation	9.5%
Industry	7.7%
Installation and maintenance	3.8%
Healthcare	4.8%
Social, socio-educative and socio-cultural action	16.4%
Entertainment industry	1%
Support to firms	13.3%
Transportation and logistics	9%
Missing	4.3%

Notes: This table describes the distribution of the educational achievements, qualification and desired occupations of job seekers in Auvergne-Rhône-Alpes from 2019 to mid 2022 (n=1 210 854).

Table 2.3: Job seekers: desired occupations & qualifications

the elderly or children; 15.3% teaching jobs; 14% cleaning jobs); 13.3% in “support to firms" (among which, 36.2% for secretary or assistant jobs; 10% for IT; 10% for accounting and management). The median wage sought is 10.89 euros, close to the minimum wage across the 2019-2022 period. Job seekers tend to look for full-time jobs (76.1%), indefinite duration contract (61%), and the average accepted mobility radius is around 30 kilometers.

Pre-processing After pre-processing and one hot encoding relevant categorical variables, job seeker - more precisely, their unemployment spells - are represented in circa 500 dimension. 100 of these dimensions correspond to a singular value decomposition (SVD, [Dee+90]) of a TF-IDF representation of textual data describing the job seeker (“business card", professional experiences, training, skills). 50 other dimensions correspond to another SVD performed on skills (in the ROME ontology) provided by the job seeker in one’s skills profile, augmented by the skills

Desired hourly wage	Euros
Mean	13.32
Median	10.89
Desired contract	**Share**
Indefinite duration	61%
Definite duration	30%
Seasonal	5.5%
Interim	4%
Other contracts	0.5%
Part or full time	**Share**
Looking for part time job	23.9%
Accepted mobility	**Kilometers**
Mean	29.93
Median	30

Notes: This table describes the distribution of the desired hourly wages, contract types, and accepted mobility of job seekers in Auvergne-Rhône-Alpes from 2019 to mid 2022 (n=1 210 854).

Table 2.4: Job seekers: search parameters

Feature	Share
Socio-demographics	
Women	50.3%
Has children	41%
Sensitive Urban Area (CUCS / ZUS)	7.7%
Age (at registration)	**Share**
Below 25	24.4%
26-40	46.8%
41-55	22.2%
55+	6.6%
Department of residence	
Rhône	25%
Isère	15.1%
Haute-Savoie	11.5%
Loire	8.5%
Ain	7.1%
Puy-de-Dôme	6.9%
Drôme	6.5%
Savoie	5.8%
Ardèche	3.8%
Allier	3.6%
Haute-Loire	2.2%
Cantal	1.4%
Other	2.6%

Notes: This table describes the distribution of socio-demographic attributes of job seekers in Auvergne-Rhône-Alpes from 2019 to mid 2022 (n=1 210 854).

Table 2.5: Job seekers: socio-demographic characteristics

corresponding to job seekers' desired occupation [5]. The remaining dimensions can be roughly classified into: i) job seekers' qualifications; ii) job seekers' preferences; iii) socio-demographic variables; iv) past employment history and relationship to the PES; v) resume elements; vi) geographic information. Table 2.6 and 2.7 describe all job seeker features used in the algorithm in further detail.[6]

[5] Correspondences between skills and occupations are based on *France Travail*'s ROME ontology.

[6] This set of features was constructed by progressive enlargement based on models' validation set performances (with candidate features for enlargement chosen based on expert suggestions). An economist may be surprised at the absence of features known to predict return to employment, such as the duration of unemployment, level and duration of unemployment benefits, and the occupations' tightness ratio. These features required computations at the weekly or monthly level that reduced tractability, and had no significant effect on recall. Their lack of predictive power for job recommendation is attributed to the fact that they may act as second-order modifications on job seekers' interest among job ads (while they may crucially matter for the timing of finding a job).

Qualifications	
Target job sector	categorical (x14)
Target job	categorical (x110)
Number of years of experience	numeric
Maximum level of qualification	categorical (x10)
Department	categorical (x13)
Vocational training field	categorical (x27)
Skills (SVD)	numeric (x50)
Driving licences	categorical (x22)
Number of languages spoken	numeric
Means of transportation	categorical (x5)
Latitude	numeric
Longitude	numeric
Preferences	
Reservation wage (euros / hour)	numeric
The job seeker is looking for a full-time job	binary
Target type of contract	categorical (x13)
Maximum commuting time	numeric
Maximum (and Minimum) number of work hours per week	numeric

Notes: This table enumerates the features describing job seekers in terms of qualifications and preferences given as input to the Muse algorithm, along with their type (numeric, categorical) and dimension. The distinction between "qualifications" and "preferences" will be used in Chapter 7.

Table 2.6: Job seeker features (1): qualifications and preferences

Socio-demographic variables	
Number of children	numeric
Job seeker lives in a QPV area	numeric
Past employment history	
Number of unemployment periods since 2018	numeric
Reason why the job seeker registered at PES	categorical (x15)
Type of accompaniment received from PES	categorical (x4)
Main obstacles assumed to slow return to employment	categorical (x4)
Resume	
Curriculum text (SVD)	numeric (x100)
Number of words in the curriculum text	numeric
Number of visit cards	numeric
Number of sectors considered by the job seeker	numeric
Geographic information	
Firm density within zip code	numeric
Unemployment rate within zip code	numeric

Notes: This table enumerates the features describing job seekers (aside from those relatives to qualifications and preferences, listed in Table 2.6) given as input to the Muse algorithm, along with their type (numeric, categorical) and dimension.

Table 2.7: Job seeker features (2): other variables

2.3 Job ads

Scope and limits of labor demand coverage Firms may post job ads on *France Travail*'s website[7], which can be browsed and applied to by job seekers. 2 205 647 job ads (pooling ads from the Auvergne-Rhône-Alpes region and adjacent French "départements") are observed from early 2019 to mid 2022. The number of job ads available at a given point in time is much smaller (67,720 on average a given week) than the total over the period, since job ads can be deleted if the firm has filled the position or stopped their recruiting process, and expire after a set duration. These job postings stem from 165 395 establishments (*i.e.* geographic sites of firms).

France Travail also gathers and displays job ads from "partner" institutions, such as large recruitment or interim firms (*e.g.* Adecco, Manpower, APEC). These partner-provided job ads represent 65.6% of all ads displayed on *France Travail*'s website at the national level. Nevertheless, these partner-provided job ads will not be used in the present study for two reasons. First, their format differs from ads directly posted at *France Travail*, making standardization a sizeable challenge. Secondly, the recruitment process for those ads is managed by the partner institutions rather than *France Travail* making it much harder to track associated applications and hires. Taking those ads into consideration in the training and recommendation process would constitute an empirically valuable extension of the present work, but is out of the scope of the present document.

Moreover, the union of *France Travail*'s job ads, and of "partner" ads does not constitute an exhaustive inventory of all region-wide job openings. For instance, firms may also broadcast job openings informally, on their own websites, internally only, through third party institutions that are not "partners" of *France Travail*.

Descriptive statistics When posting a job ad at *France Travail*, recruiters fill in rich tabular information on the advertised position. Lower (mandatory) and upper (optional) bounds for the job's wage are provided. Contract type and duration, occupation, working hours, required experience, education, driver's licences, spoken languages are specified. Both the job ad and the recruiting firm are described in natural language.

Table 2.8 provides descriptive statistics on job ads posted at *France Travail*. 67.1% of job ads contain no upfront educational requirements (either required or desired) - although this figure leaves out requirements that are implicit or specified in additional text. Only 13.6% of ads explicitly require higher educational achievements. The median lower bound for hourly wage, at 10.92 euros, is close to the hourly minimum wage (the gross minimum wage was 10.03 euros in 2019 and rose to 10.57 euros in early 2022). 48% of job ads offer indefinite duration contracts, and 78.4% of them offer full time jobs.

[7]They may do so standalone, or after having been contacted by *France Travail*'s firm-oriented caseworkers, whose missions include pro-actively contacting firms to seek to answer their recruitment needs. Recruiters may optionally receive *France Travail*'s help in the recruitment process for the advertised positions, *e.g.* for candidate selection (as evaluated in [ACG20]).

Required education level	Share
5+ years of higher education	1.6 %
3 or 4 years of higher education	4.4 %
2 years of higher education	7.6%
General secondary education (“baccalauréat")	5.8%
Vocational secondary education (“CAP / BEP")	12%
Lower levels of education	1.5 %
Missing	67.1 %
Qualification	**Share**
“Manoeuvres"	4.2%
Blue-collar, unqualified	5.5%
Blue-collar, qualified	10.6%
White-collar, unqualified	24.4%
White-collar, qualified	37.3 %
Technician	8.9%
Blue-collar supervision	4.5%
Executive	4.5%
Occupation	**Share**
Agriculture, green spaces, animal care	2.2%
Craftsmanship	0.2%
Banking, finance, real estate	1.6%
Retail, sales	14.6%
Communication, media and multimedia	0.6 %
Construction	9.3%
Hotels, restaurants, tourism, leisure, animation	12.3%
Industry	9.2%
Installation and maintenance	7%
Healthcare	5.9%
Social, socio-educative and socio-cultural action	17.9%
Entertainment industry	0.1%
Support to firms	10.9%
Transportation and logistics	8.1%
Wage	**Euros**
Lower bound (hourly), mean	11.89
Lower bound (hourly), median	10.92
Upper bound (hourly), mean	13.69
Upper bound (hourly), median	12.26
Contract Type	**Share**
CDI	48.3%
CDD	30.5%
SAI	4.6%
Interim	14.6%
Other	2%
Full-time	**Share**
Full-time	78.4%

Notes: This table provides descriptive statistics on job ads posted at France Travail in the Auvergne-Rhône-Alpes region and adjacent “départements" from 2019 to mid-2022 (n= 2 205 647).

Table 2.8: Job ads: descriptive statistics

Features	
Skills SVD	numeric (x50)
Text	numeric (x200)
Job sector	categorical (x14)
Job	categorical (x110)
Contract type	categorical (x12)
Weekly duration	numeric
Experience	numeric
Full time	boolean
Soft skills	categorical (x14)
Driver's license	categorical (x16)
Yearly wage (min)	numeric
Yearly wage (max)	numeric
Contract duration	numeric
Inferred wage lower bound	numeric
Pop. density within zip code	numeric
Firm density within zip code	numeric
Unemployment within zip code	numeric
Latitude	numeric
Longitude	numeric
Establishment size	categorical (x16)
Wage type	categorical (x5)
Missing yearly wage min	boolean
Missing yearly wage max	boolean
Education level	categorical (x11)
Type of education	categorical (x20)
Hourly min wage equivalent	numeric
Hourly max wage equivalent	numeric

Notes: This table enumerates features representing job ads provided as input to the Muse algorithm, along with their type (numeric, categorical) and dimension.

Table 2.9: Job ads: features

Pre-processing After pre-processing, job ads are represented in circa 500 dimension. 200 of these dimensions correspond to an SVD on the textual description of the job ad and firm, concatenated together. 50 dimensions correspond to an SVD on required and desired skills (in the ROME nomenclature), as well as those associated to the occupation[8]. The rest of features describe the job and firm. They include the occupation, required education, the type of contract, weekly duration, and wage descriptors. Establishments are characterized by their size, their location, and socio-demographic features of the location. The list of features used as inputs to the algorithm is provided in Table 2.9[9].

[8]In fact (and as on the job seeker side), the skills SVD is learned on the concatenation of the skills of job seekers and job ads, because they share a common nomenclature. On the other hand, the SVDs on textual data on both sides are fitted separately on job seekers and job ads because language varies between the two parties, as noted by [Sch+17].

[9]As for job seekers, this set of features was constructed by progressive enlargement based on models' validation set performances.

2.4 Interactions

We now describe the labor market interactions we shall leverage, namely applications (subsection 2.4.1) and hires (subsection 2.4.2). [10] We seek to recommend job ads rather than firms, and accordingly to define matches at job seeker - job ad dyad level (for training and evaluation downstream).

2.4.1 Applications

Institutional background *France Travail* enables and logs a variety of interactions between labor supply and demand. First, *France Travail*'s caseworkers may: i) recommend job ads to job seekers (if interested, job seekers may apply by themselves, or let the caseworker serve as an intermediary); ii) recommend job seekers to firms; iii) pre-select job seekers for job ads which rely on the PES's pre-selection service[11]. In all three cases, caseworkers have incentives to document applications they inter-mediated on the institution's platform - even more so if they end up leading to a match. Second, job seekers may spontaneously apply to job ads on the PES's website without any help from caseworkers. These applications may be made through the institution's interface, in which case they are logged. However, if job ads display recruiters' contact information, job seekers may contact the employer directly without using the PES's website to apply (they have no incentive to use it). In that case, the application is not logged, and we have no way of knowing with certainty if it took place at all. Third, recruiters may also directly contact job seekers (*e.g.* through the skills profile described above), possibly without going through the PES's interface if a job seekers' contact details are given directly in his or her skills profile.

Descriptive statistics In the following, we will focus on inter-mediation acts initiated by job seekers, which we will refer to as *applications*. We observe 1 292 694 applications involving 154 934 job seekers - *i.e.* 12.8% of the job seeker population. The sparsity of applications is 2×10^{-7} at the weekly level, and 4×10^{-7} at the aggregate level (pooling all time periods).

Differences between job seekers with applications and job seekers without applications are documented in Table 2.10. First, unsurprisingly, be it in terms of education, desired occupation, or qualification, job seekers who apply through the PES are less often described by missing values. Second, applicants are less often very qualified (4 percentage points fewer have a master's degree or PhD, 3.3 percentage points fewer look for executive positions); but more frequently look for white-collar qualified jobs (9 percentage points difference). They less often search for jobs in construction (3.5 pp less), in industry (1 pp less), and more often in support to firms (5.8 pp more).

[10] Clicks on job ads, collected when a job seeker is logged in on a personal account on the PES's website, would also be relevant, but were not systematically available for technical reasons (except for the year 2019), and are thus left out of the analysis (except in Chapter 4).

[11] Job seekers apply to these ads through the PES's system; the caseworker then selects which profiles are shown to the employer.

	Non-applicants (pp)	Applicants (pp)	Difference (pp)	p-value
Education				
5+ years of higher education	12.5	8.4	-4.1	0.000
3 or 4 years of higher education	11.4	11.7	0.3	0.000
2 years of higher education	14.1	18.0	3.9	0.000
General secondary education (“baccalauréat")	23.8	24.7	0.9	0.000
Vocational secondary education (CAP / BEP)	25.0	29.2	4.2	0.000
Lower levels of education	9.6	6.4	-3.2	0.000
Missing	3.5	1.5	-2.0	0.000
Job				
Agriculture, green spaces, animal care	3.6	2.4	-1.2	0.000
Craftsmanship	0.8	0.6	-0.2	0.000
Banking, finance, real estate	1.5	1.3	-0.2	0.000
Retail, sales	14.4	16.1	1.7	0.000
Communication, media and multimedia	2.4	2.0	-0.4	0.000
Construction	8.1	4.5	-3.5	0.000
Hotels, restaurants, tourism, leisure, animation	9.4	9.7	0.3	0.001
Industry	7.8	6.8	-1.0	0.000
Installation and maintenance	3.8	3.5	-0.4	0.000
Healthcare	4.9	4.6	-0.2	0.000
Social, socio-educative and socio-cultural action	16.4	17.0	0.6	0.000
Entertainment industry	1.0	0.5	-0.5	0.000
Support to firms	12.5	18.4	5.8	0.000
Transportation and logistics	8.8	10.8	2.0	0.000
Missing occupation	4.6	1.8	-2.8	0.000
Qualification				
Manoeuvres	2.9	2.0	-0.9	0.000
Blue-collar, unqualified	3.9	3.4	-0.5	0.000
Blue-collar, qualified	8.7	7.8	-1.0	0.000
White-collar, unqualified	18.8	18.7	-0.0	0.803
White-collar, qualified	37.8	47.1	9.3	0.000
Technician	5.8	6.3	0.5	0.000
Blue-collar supervision	4.9	5.2	0.3	0.000
Executive	11.3	8.0	-3.3	0.000
Missing qualification	5.8	1.5	-4.3	0.000

Notes: This table describes the distribution of education, desired jobs and qualifications for job seekers within the Auvergne-Rhône-Alpes region from 2019 to mid 2022 without any logged applications (Column 1, n = 1 055 920), for those with at least one application (Column 2, n = 154 934), the difference between the two populations for each attribute (Column 3) and the p-value of a test of equal means (Column 4).

Table 2.10: Characteristics of job seekers who apply through the PES (percentage points)

Contract types	
Indefinite duration	50.7 %
Definite duration	37.8 %
Interim	5.9%
Other	5.6 %
Distance (km)	
Mean	24.56
Median	10.06
Wages (mean, euros)	
Lower bound	11.37
Upper bound	12.67
Fit measures	
Geography	50.4%
Job type	26.26%
Wage	45.8%
Contract	44.6%
Duration	75.7%

Notes: This table describes applications (n=192694) in terms of job ads' characteristics and fit to job seekers' search parameters.

Table 2.11: Applications: descriptive statistics

Table 2.11 provides evidence on the characteristics of the job seeker - job ad pairs in applications. 50.7% of them are indefinite duration contracts, and 37.8% of them definite duration contracts (CDDs). The median distance of a job ad to a job seeker's post code is 10.06 kilometers. Table 2.11 also proposes measures of the adequacy of the job to job seekers' declared parameters. The average adequacy between job seekers' desired job and the job ad's occupation (at the ROME level) is 26.26%[12]. Job seekers' search parameters in terms of geography are satisfied in only 50% of applications; job seekers also apply more than half the time to jobs that do not match their search parameters in terms of wage and contract.

2.4.2 Hirings

Match definition Our information on hires comes from two complementary sources. First, *France Travail*'s logs of interactions may contain hiring dates when recorded interactions are successful. Second, *France Travail* has access to administrative data (the *Déclaration Préalable À l'Embauche*, henceforth DPAE) which record in which firm job seekers were hired in a nearly exhaustive fashion, as recruiters have legal obligations to declare new hires. However, matching these hires in firms to posted job ads is not trivial.

Based on these data sources, hires at the job seeker - job ad level are defined in the following fashion. First, *France Travail*'s interaction logs may contain non-empty hiring dates in the case of successful interactions. These are reliable in the

[12]This figure should be treated as a lower bound, and is heavily dependent on the pre-processing of job seekers' occupations. It does not account for the fact that job seekers may look for several jobs.

	Non-hired (pp)	Hired (pp)	Difference	p-value
Education				
5+ years of higher education	12.9	6.5	-6.4	0.000
3 or 4 years of higher education	11.6	10.5	-1.2	0.000
2 years of higher education	14.4	15.7	1.3	0.000
General secondary education ("baccalauréat")	23.4	27.3	3.9	0.000
Vocational secondary education (CAP / BEP)	24.8	29.5	4.7	0.000
Lower levels of education	9.4	8.2	-1.2	0.000
Missing	3.4	2.3	-1.2	0.000
Job				
Agriculture, green spaces, animal care	3.4	2.4	-0.0	0.969
Craftsmanship	0.8	0.6	0.1	0.001
Banking, finance, real estate	1.5	1.3	-0.5	0.000
Retail, sales	14.3	16.1	2.2	0.000
Communication, media and multimedia	2.5	2.0	-0.9	0.000
Construction	7.9	4.5	-2.0	0.000
Hotels, restaurants, tourism, leisure, animation	9.2	9.7	1.5	0.000
Industry	7.7	6.8	-0.1	0.356
Installation and maintenance	3.7	3.5	0.4	0.000
Healthcare	4.7	4.6	0.5	0.000
Social, socio-educative and socio-cultural action	16.5	17.0	-0.6	0.000
Entertainment industry	1.0	0.5	-0.4	0.000
Support to firms	13.4	18.4	-0.9	0.000
Transportation and logistics	8.8	10.8	2.2	0.000
Missing occupation	4.5	1.8	-1.6	0.000
Qualification				
Manoeuvres	2.8	3.0	0.2	0.000
Blue-collar, unqualified	3.8	4.2	0.4	0.000
Blue-collar, qualified	8.5	9.2	0.7	0.000
White-collar, unqualified	18.3	21.1	2.8	0.000
White-collar, qualified	38.0	44.6	6.5	0.000
Technician	5.9	5.8	-0.0	0.417
Blue-collar supervision	5.2	3.7	-1.4	0.000
Executive	11.9	5.3	-6.6	0.000
Missing qualification	5.6	3.0	-2.5	0.000

Notes: This table describes the distribution of education, desired jobs and qualifications for job seekers within the Auvergne-Rhône-Alpes region from 2019 to mid 2022 without any hires matched to job ads in the dataset (Column 1, n = 1 030 584), for those with at least one such hire (Column 2, n = 180 270), the difference between the two populations for each attribute (Column 3) and the p-value of a test of equal means (Column 4).

Table 2.12: Job seekers linked to hires on job ads posted at France Travail

sense that false positives are unlikely, but may contain false negatives (interactions may lead to a hire without the hire appearing in the records). Second, we assume that if a job seeker applied to a firm's job ad, and that he or she is hired in that firm based on the DPAE, the hire took place on that job ad. This reconstruction process may be flawed, especially for large firms, since a job seeker may have applied to some kind of job and have been hired on another. Third, if a firm posted a single job ad at *France Travail*, and a single job seeker is hired in that firm according to the DPAE, the hire is assumed to have taken place for the posted job ad. This induces a bias towards hires in small firms (compared to firms posting several ads at the PES), on top of the same disadvantages listed in the second type of match reconstruction (amplified by the fact that nothing ensures us that the job seeker was interested in a specific ad).

This definition process yields 285,992 hires. 38.4% of them originate from *France Travail*'s logs directly, 46.2% from the match of interaction logs and the DPAE, and 15.3% from the third assumption. The sparsity of hires is 5×10^{-8} at the weekly level, and 10^{-7} at the aggregate level.

Descriptive statistics Descriptive statistics of job seekers for which we observe hires as defined above and their matches are provided in Tables 2.12 and 2.13

Contract types	
Indefinite duration	43.5 %
Definite duration	40 %
Interim	9.7%
Other	6.8 %
Distance (km)	
Mean	18.2
Median	7.9
Wages (mean, euros)	
Lower bound	11.16
Upper bound	12.44
Fit measures	
Geography	52%
Job type	30.5%
Wage	48.2%
Contract	41.2%
Duration	77.2%

Notes: This table describes hirings in terms of job ads' characteristics and fit to job seekers' search parameters.

Table 2.13: Hirings: descriptive statistics

respectively. Job seekers which we manage to link to hires on job ads posted at *France Travail* have less often spent five years in higher education (-6.4 percentage points), and look less often for executive positions (-6.6 percentage points), than job seekers for which this is not the case[13]. 43.5% of matches linked to *France Travail* job ads are indefinite duration contracts, and 40% of them definite duration contracts (CDDs). The median distance of a job ad to a job seeker's post code is 7.9 kilometers. The job seekers' desired job and the job ad's occupation match in 30.5% of cases[14]. Job seekers' search in times of geography are satisfied roughly half the time; they often find contracts with a duration that fits their search criteria (77%); but their demands in terms of wages and contracts are met less than half of times. In comparison to applications (Table 2.13), hires are less often of indefinite duration, slightly closer geographically, and around the same average wage levels (this should not be over-interpreted since the population of applicants and hired job seekers differ).

Representativity Hires as defined above should not be considered representative of all matches on the labor market. They should be thought of a subset of the French labor market involving both job seekers and firms that search for jobs

[13]Our interpretation is that executives or highly qualified job seekers may resort less often to *France Travail*'s services when looking for a job compared to other platforms and search channels (as suggested by statistics on applicants in Table 2.10). Job ads posted at *France Travail* may also be less relevant to them (as suggested by Table 2.8). As a result, the matches these populations find might take place out of the scope of our data more often.

[14]As noted above, these figures should be treated as lower bounds, are heavily dependent on the pre-processing of job seekers' occupations, and do not account for the fact that job seekers may look for several jobs.

through *France Travail*, and bear the mark of the work of caseworkers. One may contrast the observations we observe with overall hires observed in the DPAE - which include establishments that do not use *France Travail* as a recruitment channel. For comparison, we can look at all hires involving jobs seekers from the sample, from 2019 to mid-2022. 76% of job seekers (920 364 among 1 210 854) were involved in at least one DPAE throughout the period. The hirings in the DPAE involve 2 871 848 job seeker - firm pairs - more than 10 times the number of job seeker - job ad pairs described above. Taking the first contract between the pair by default when several exist, we find 42.8% of these DPAE-logged contracts to be short-term duration contracts, 31.4% to be indeterminate employment ones, and 25.7% to be interim contracts.

Discussion: choice of the job seeker - job ad dyad level In the following, recommender systems will be learnt at the job seeker - job ad dyad level (typically from hires), and recommend job ads to job seekers. A justification of this choice is in order given data limitations listed above, especially uncertainty about hires at the job seeker - job ad dyad level. Another option would have been to *learn from job seeker - firm pairs, and recommend firms* as in [Beh+22]. Such a choice would enable a broader use of DPAE data and avoid any guesswork as to what counts as a "hire". It also comes with important downsides: i) recommending a firm is vague and uninformative, especially for large firms with different kind of job openings; ii) firms may not necessarily have open positions (at *France Travail* and in general), iii) their current open positions may be irrelevant for the job seeker.

Chapter 3

The MUSE algorithm

The present chapter presents our proposed contribution to cold-start, sparse recommendation (section 3.1), named MUSE (for "MUlti-head Sparse E-recruitment"). In Section 3.2, the approach is validated on RecSys 2017 challenge data (for the sake of public benchmarking), and on *France Travail* data.

3.1 Proposed approach

Overview MUSE is a two-tiered neural architecture tailored for job recommendation. The first tier, illustrated in Fig. 3.1, aims to enforce the scalability of the approach by leveraging a "two-tower" structure. It uses the elementary descriptions of the job seeker x and the job ad y, and computes a fast score MUSE.0(x, y) based on the dot product between their respective embeddings. This score is exploited to rank and filter all but the top 1,000 job ads, narrowing the search for the second tier MUSE1 - illustrated in Fig. 3.2 -, enabling the use of a more expressive model and of more intricate features[1].

3.1.1 MUSE.0: candidate retrieval stage

MUSE.0 models three facets relevant to job recommendation, respectively concerned with competences and skills, geographical, and general aspects. The faceted match of job seeker x and job ad y is sought as:

$$s_0(x, y) = \langle \phi_0(x), \psi_0(y) \rangle$$

[1]In terms of its development process, MUSE was developed on the *France Travail* dataset after establishing a boosted tree ensemble baseline (XGB) on the data [CG16]. Feature selection was largely based on the impact on recall of iterative feature addition to XGB. MUSE started as a single, standalone embedding-based model (the "general" MUSE.0 module below) which under-performed compared to XGB. Based on inspection of the models' errors, XGB's advantage seemed to be linked to a better use of geographic information and to its use and definition of pair-level features (which cannot be provided as inputs to a two-tower structure). These findings led to the addition of the geographic module in MUSE.0, and, due to the success of this fused-embedding strategy, to the addition of the skills module. They also suggested the addition of second tier MUSE.2 to work around limits of the two-tower structure, enabling the architecture to take as input and learn pair-wise elements (*e.g.* using multiplicative interactions) while maintaining scalability.

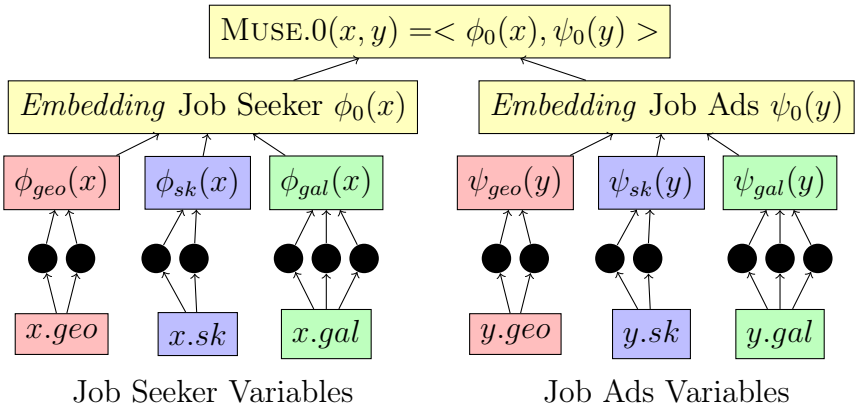


Figure 3.1: MUSE.0 architecture: three embeddings are defined to model geographical, skills and general aspects of job seekers (left) and job ads (right), and compute the hiring score.

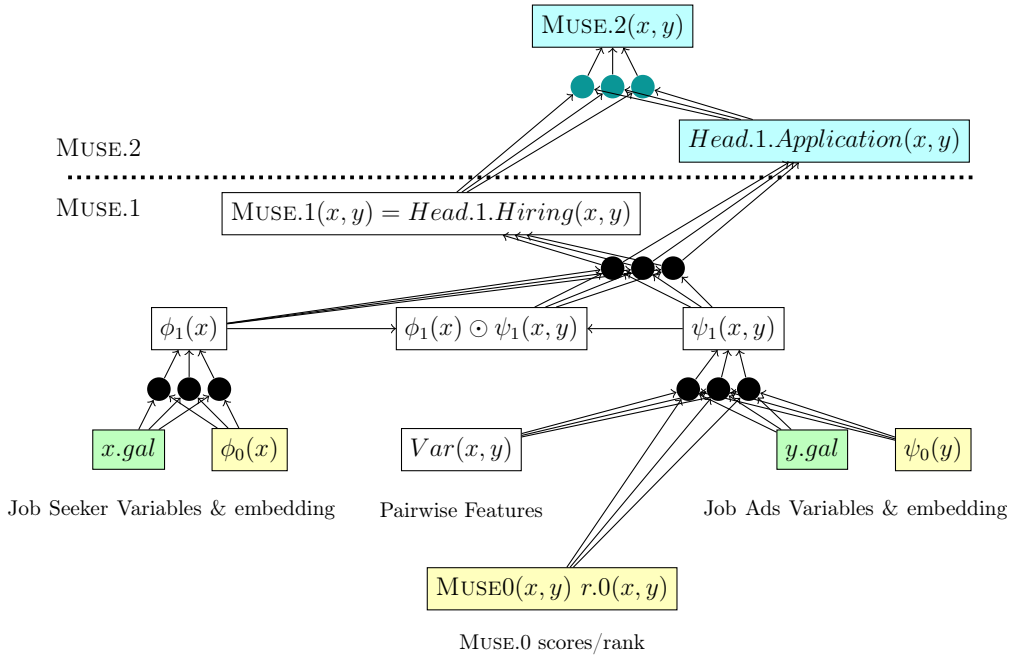


Figure 3.2: MUSE.1 (below dashed line) and MUSE.2 architectures. MUSE.2 includes a second head to model applications, and a top head, leveraging both the standalone hiring and the application scores to predict the overall hiring score.

where embeddings ϕ_0 and ψ_0 are trained using a triplet loss [WS09]. Noting (x, y, y') a triplet[2] made of job seeker x, their match y and another job ad $y' \neq y$,[3] the loss is defined as:

$$\mathcal{L}(\phi_0, \psi_0) = \sum_{(x,y,y')} [\langle \phi_0(x), (\psi_0(y) - \psi_0(y')) \rangle + \eta]_+ \tag{3.1}$$

with $[x]_+ = \max(x, 0)$ and $\eta > 0$ a margin factor. Intuitively, for a given triplet x, y, y', the minimal loss (of zero) is achieved if the score $s_0(x, y)$ associated to the positively labeled pair is separated by a margin at least η on the real line from the score $s_0(x, y')$ associated to the negatively labeled pair. [4]

ϕ_0 and ψ_0 are defined by concatenating three embeddings, respectively reflecting skills, geography and general information. The separation of these aspects is motivated by the evidence on hirings provided in section 2.4.2 (Table 2.13), which highlights the role of geography in the matching process (a median of 7.9 kilometers between job seekers and their future jobs), and of the importance of going beyond the standard ROME job ontology since few job seeker - job ad pairs are exact matches in terms of the ontology's job definitions. The gains in separate treatment of these aspects will be vindicated by ablation studies in Section 3.2.

The skill matching module (ϕ_{sk}, ψ_{sk}). This module takes as input the job seekers' and job ads' skills sets in the ROME ontology, represented by multi-hot encodings of dimension circa 12,300[5]. The job seekers' (resp. ad's) skill sets are defined as the union of the skills explicitly indicated (resp. required or desired) by the job seeker (resp. job ad), and those associated to the job seeker's desired occupations (resp. the ad's occupation) in the ontology. Thus, the input representation can be thought of as a joint representation of job seekers' and ads' jobs and skills. Embeddings ϕ_{sk} and ψ_{sk} are learned using a triplet loss.

The geographical matching module (ϕ_{geo}, ψ_{geo}). This module is based on a tiled representation of the locations, taking inspiration from kernel density estimation and matrix factorization [Lia+14]. Formally, given a reference grid paving the regional territory with points g_i, the geographical representation of a job seeker

[2] One could also have considered sampling negative job seeker examples x' and learning at the level of a quadruplet $x - y - y' - x'$, which would be a first step towards bilateral recommendation. However, the reverse problem of recommending job seeker profiles to recruiters is not necessarily symmetric in practice (*e.g.* due to practical / legal / ethical constraints on data usage). In the sole perspective of job recommendation, it is unclear whether such quadruplet-level training would improve recommendations. Furthermore, symmetry between the treatment of job seekers and job ads is broken at the re-ranking stage (MUSE.2).

[3] y' is uniformly sampled among the job ads *available during the match week*. More sophisticated negative sampling strategies have been considered with no improvement. This is rather surprising given common practices in the deep metric learning literature, but these findings nevertheless align with those in [Sch18].

[4] Experiments using other losses, *e.g.* the logistic one at the pair level, were also conducted. The choice of the loss function (among classical ones) seems to have limited impact.

[5] Rather than the 50-dimensional SVD embedding described in Chapter 2, which is however provided to the general module described below.

(resp. job ad) situated at $x.geo \in \mathbb{R}^2$ (*i.e.* a latitude, longitude tuple) and supplied as input of the geographical module is set to:

$$G(x.geo) = \{\exp^{-\omega \cdot d(x.geo, g_i)}\}_i$$

with d the geodesic distance and $\omega > 0$ controlling the granularity of the representation (the speed at which points far from g_i tend to a representation of zero at g_i). The g_i's may be thought of as a way to perform "soft" encoding of post codes on a discretized grid, in which post codes by (a transformation of) their distances to the grid points. ψ_{geo} is set to the identity [6], *i.e.* the geographic score takes the form

$$\langle \phi_{geo}([G(x.geo), x.geo]), G(y.geo) \rangle$$

Embedding ϕ_{geo} is learned on the top of the tiled representation using a triplet loss. The only difference with respect to equation 3.1 lies in the negative sampling strategy, as job ad y' is uniformly selected among the job ads contemporary of y *and situated farther away* from job seeker x.

Let us compare our approach to that of [Zha+21], which also incorporates geolocation embeddings in a two-tower fused-embeddings candidate retrieval structure. The authors encode latitude and longitude of ads and job seekers in their Cartesian coordinates (in dimension 3). The dot product between the Cartesian coordinates corresponds to an approximation of the geographic distance between the two points, ensuring the representation's compatibility with an inner-product based architecture. Their approach has the merit of using only three dimensions to embed geography. However, their modeling choice imposes that their model's global scores are partially linear in the geographic distance $d(x, y)$, *i.e.* have the form

$$\langle \phi_0(x), \psi_0(y) \rangle + w_d d(x.geo, y.geo)$$

with w_d a scalar weight. Our approach avoids this functional form assumption at the cost of a larger embedding. Moreover, by its choice of negative sampling, this module is also able to reflect the fact that the impact of the distance of a job seeker to a job depends on other factors (public transportation; traffic jams) than the distance in kilometers: it is not invariant by translation.

The general matching module (ϕ_{gal}, ψ_{gal}). The general matching module takes as input the 500-dimensional vectors with all information related to job seekers and job ads described in Section 2.2 and 2.3 (see in particular Tables 2.7) and 2.9). Embeddings ϕ_{gal} and ψ_{gal} are likewise learned using a triplet loss.

The training schedule. The parameters to be trained are the neural network weights which parametrize the embeddings, *i.e.* the weights of $\phi_{sk}, \phi_{geo}, \phi_{gal}$ and $\psi_{sk}, \psi_{geo}, \psi_{gal}$. Each of the three modules is pre-trained standalone in a first phase to provide appropriate initialization. In a second phase, all three modules are jointly trained and fine-tuned in a second phase using stochastic gradient descent

[6] This choice led to improved results compared to learning an embedding for $y.geo$.

with Adam [KB14][7]. More details on hyper-parameters are provided in Appendix A. Overall, MUSE.0 yields a scalar matching score:

$$\text{MUSE.0}(x, y) = \sum_{m \in \{sk, geo, gal\}} \langle \phi_m(x), \psi_m(y) \rangle$$

3.1.2 MUSE.1 and MUSE.2: re-ranking stage

MUSE.1 As said, the MUSE.0 score is used to filter the job ads considered for each job seeker. The recall@1,000 of MUSE0 is above 80%, making it possible to only consider the top 1,000 job ads for each job seeker with a limited loss in recall. MUSE.1, refining the ordering of the top 1,000 job ads, uses more complex features $Var(x, y)$ depending on both job seeker x and job ad y[8], which would not be possible for scalability reasons if all available job ads were considered, and a more elaborate architecture.

MUSE.1 takes as input the description of x and y (*i.e.* the inputs of MUSE.0's General module), the crossed features $Var(x, y)$ and the information provided by MUSE.0 - *i.e.* the latent description $\phi_0(x)$ and $\psi_0(y)$, the score MUSE.0(x, y) and the rank of y according to MUSE.0(x, y). Overall, the recommendation score learned by MUSE.1 reads:

$$\text{MUSE.1}(x, y) = \text{MLP}(\phi(x), \psi(x, y), \phi(x) \odot \psi(x, y))$$

where MLP denotes a multi-layer perceptron, and ϕ, ψ are respectively job seeker and job ad embeddings. The term $\phi(x) \odot \psi(x, y)$ enables the automated learning of user-item interactions relevant to the recommendation problem using multiplicative interactions in the spirit of [Guo+17]. MUSE.1 is trained by minimizing a cross-entropy loss, predicting whether a pair corresponds to a hire or not:

$$\mathcal{L} = \sum_{(x,y,y')} \log(\text{MUSE.1}(x, y)) + \log(1 - \text{MUSE.1}(x, y')) \tag{3.2}$$

The sampling of negative pairs is done uniformly at random among MUSE.0's top-1000 selection. The network's weights (*i.e.* those defining ϕ, ψ and the MLP) are trained using Adam [KB14].

MUSE.2 As said, a critical difficulty in the *France Travail* framework is the extreme sparsity of the interaction matrix in the dataset (a single hire being generally reported for the hired job seekers, and 0 for the others). In the spirit of multi-task learning, to exploit information in application behavior that may be relevant to predict hirings, a multi-head MUSE.2 architecture is considered to enable information sharing between the hiring and the application interaction matrices (taking the definition of applications provided in section 2.4.1). A first head aims to predict

[7] While non-zero, the gains associated to this second phase are relatively limited compared to simply concatenating the three pre-trained modules without joint training.

[8] Vector $Var(x, y)$ measures the adequacy of an (x, y) pair *re* the distance, skills, occupation, education, experience, contract type, spoken languages, driving licenses and wages.

the hirings; a second head aims to predict the applications; a third head, aimed to predict the hirings, is learned on the top of both first and second heads, likewise using a cross-entropy loss (Eq. 3.2)[9].

3.2 Validation: experiments *in silico*

This section's goal is to comparatively assess the performance of MUSE in terms of both performance and inference time. The single head (MUSE.1) and the multi-head (MUSE.2) architectures are compared and the impact of the different modules is assessed using ablation studies.

3.2.1 Experimental settings

Datasets

Xing As said, one of the datasets most relevant to job recommendation was provided by the social network Xing for the ACM Recsys 2017 challenge[10]. It involves 1.5M job seekers, 1.3M jobs and 30M interactions, recorded from Nov. 2016 to Jan. 2017 in Germany, Austria and Switzerland. As said, the source dataset was no longer made available on the competition's platform at the time of the writing of this work. Accordingly, we rely on a pre-processed version of the data distributed by [VYP17b]. After thorough anonymization and pre-processing carried out by the competition organizers and/or [VYP17b], job seekers and job ads are represented in the provided data as vectors of dimension respectively 831 and 2,738. The interaction matrix reports 6 levels of interaction, 4 of which (click, bookmark, reply, recruited) are treated as "hiring". The fifth level (impressions) is treated as "applying" and used for the MUSE.2 training.

The MUSE.0 architecture is not applicable as is on the RecSys dataset, since the geographical information and specific skill-related features can not be distinguished among the unnamed features[11]. Accordingly, the MUSE.0 architecture assessed on the RecSys dataset only includes the general module. The MUSE.1 architecture is trained from the only "hiring" interactions. The multi-head MUSE.2 architecture is trained end-to-end from the "hiring" interactions (first head and top head) and from the "application" interactions (second head).

The same training/test split procedures followed in [VYP17b] are used, including: i) a warm start scenario (426K interaction pairs), where users and items involved in the test set are also present in the training set; ii) a user cold-start scenario

[9] Recombining the two heads in final layers yielded slightly improved performances compared to a shared architecture with two separate heads (standard for multi-task learning), and from simply pooling applications and hirings in the MUSE.1 architecture.

[10] http://www.recsyschallenge.com/2017/

[11] [VYP17b] also provide pre-trained latent matrix factorization vectors (which they procede to use as part of the input provided to their proposed algorithm, DropoutNet for training). These may provide relevant behavioral information, and information on localization as a job seeker might tend to interact with nearby job ads. Nevertheless, we do not adapt MUSE to use these latent vectors as input or additional sources of information, as similar-quality latent vectors cannot be constructed in the main *France Travail* setting of interest.

(159K pairs) where 42,153 test users have no interactions in the training set. The difference in sparsity with the *France Travail* data (10^{-5} for interactions on the Xing dataset, against 10^{-7} for applications and 10^{-8} at the hirings level in *France Travail* data) is worth highlighting. While we report results in the warm start scenario for the sake of completeness, the results in cold start are of primary interest, since this is the case MUSE aims to address.

France Travail The *France Travail* data has been described above (Chapter 2). Weeks are split between train and test in 85%-15% proportions. The measure of performance is the recall@k, computed on hires. Ranking is done at the weekly level (*i.e.* among job ads available the week of the match).

Baselines

On the *France Travail* data, the machine learning baseline used for comparison is an ensemble of boosted trees [CG16], inspired by [VYP17a], winner of the Recsys 2017 challenge. It will be denoted XGB in the following. On the *France Travail* data, XGB is provided with the description input of the general MUSE.0 module ($x.gal$ and $y.gal$) plus the cross-features $Var(x, y)$ also used by MUSE.1 and MUSE.2 for a fair comparison[12].

The baseline on the Xing dataset is DropoutNet [VYP17b] (described in Section 1.2), that exploits both the job seeker and job ad description and their latent description extracted from the interaction matrix. Other algorithms, *e.g.* [Zha+21], that heavily rely on textual and geographical information, do not apply on the considered datasets[13].

3.2.2 Results

The reported computational times are obtained on Intel(R) Xeon(R) Silver 4214Y CPU @ 2.20GHz, with 187 GB RAM and a Tesla T4 GPU. Experiments on the *France Travail* dataset are conducted on a secure platform. More detail about the experiments is provided in Supplementary Material (Appendix A).

The results report the recall indicator and the computational time. Significantly best results (with 95% confidence with respect to the second best result) are denoted "*" in all tables.

The RecSys dataset

Table 3.1 reports the recall@100 and computational time of DropoutNet and the MUSE algorithms[14] along the two considered scenarios (warm and user cold start).

[12]Since the original Xing challenge data (and online test set) are no longer available, and that features are not named in the [VYP17b] version of the Xing dataset, we are unable to reproduce [VYP17a] on the Xing dataset directly.

[13]We do not assess [VYP17b] on the *France Travail* data due to its emphasis on warm start on a section of the training data.

[14]Recsys experiments replication code is provided at https://gitlab.com/solal.nathan/vadore_ijcai.

Table 3.1: Comparative results of Muse and DropoutNet on the RecSys dataset: recall@100, overall training time and recommendation time per job seeker (in seconds).

Recall@100	DropoutNet	Muse.0	Muse.1	Muse.2
Warm start	41.2*	19.1	26.3	25.8
User cold-start	23.1	17.4	24.2*	24.4*
Training time	>10h	2.7h	1.25h	8.3h
Recom. per j.s.	0.001"	0.002"	0.013"	0.016"

The warm start recommendation scenario considers test job ads and job seekers present in the training set, allowing DropoutNet to directly exploit the pre-trained matrix factorization of the collaborative filtering matrix, referred to as *CF-based representation*. As noted by [VYP17b], taking the scalar product of the job seeker and job ad representations in the *CF-based representation* even outperforms DropoutNet in warm-start mode (recall@100=42.6%). In warm-start mode, DropoutNet very significantly outperforms all Muse variants, while Muse.1 notably improves on Muse.0. This performance gap between Muse and DropoutNet is attributed to the fact that Muse does not use the *CF-based representation* as input.

In the user-cold scenario, DropoutNet proceeds by gearing together content-based embeddings called *content-based representation* trained to predict the score, as well as a reconstruction of the *CF-based representation* based on user content.

As could have been expected, the recall@100 in the cold-start scenario is degraded compared to the warm-start one. The gap is very significant for DropoutNet (from 41% to 23%) and less so for Muse (from 26% to 24% for Muse.2).

The significant improvement of Muse.1 compared to Muse.0 in both scenarios is explained from the fact that Muse.1 builds upon the pre-selection of the top 1,000 job ads enabled by Muse.0 (the recall@1,000 of Muse.0 is 87%). This filter allows for a refined negative sampling in training mode, selecting job ads y' better suited on average to the job seeker x than random job ads. In inference mode, the filtering of the top 1,000 candidate job ads is key to the low computational cost.

Interestingly, Muse.1 and Muse.2 slightly but statistically significantly outperform DropoutNet in user cold-start mode. A tentative interpretation for this fact is that both Muse.1 and Muse.2 exploit the score and rank associated with a pair (x, y) by Muse.0: this information expectedly gives some hint into the global structure of the job market, though in the perspective of the job seeker only. Further work will investigate the use of a better exploitation of the Muse.0 output, e.g. considering also the rank of x for y based on Muse.0(x, y).

The fact that Muse.2 does not improve on Muse.1 suggests that the RecSys "application" matrix (gathering only the "impression" interactions) does not yield a sufficiently diversified information about the job seekers' preferences compared to the "hiring" matrix (gathering all other interactions).

Experimental results on the *France Travail* data

Table 3.2 reports the recall@$\{10, 20, 100, 1000\}$ and computational time of XGB and MUSE on the *France Travail* dataset.

Validation *w.r.t.* XGB The main finding is that all MUSE variants but MUSE.0 significantly outperform XGB wrt recall@10, 20 and 100, with an inference runtime lesser by two orders of magnitude.

Impact of the two-tier structure These good performances in both terms of recall and runtime are explained from the filter built on the top of the MUSE.0 score. On one hand, the recall@1000 of MUSE.0 is circa 82%, upper bounding by construction the recalls of MUSE.1 and MUSE.2 (though not in a significantly detrimental way). On the other hand, the filter based on the MUSE.0 score contributes to the quality of the learned model, *re* the description of the data and the algorithm itself. At the level of the description of the (x, y) pairs, the filter enables to consider the expensive $Var(x, y)$ features (reminding that these features are also provided to XGB for a fair comparison). At the level of the algorithm and the learning trajectory, the filter also contributes to a more educated negative sampling, as job ads y' are now selected among the top 1,000 jobs suited to x.

MUSE.1 significantly improves on MUSE.0 for all recall indicators. It performs on par with the first head of MUSE.2 (also trained to predict the hiring interactions). Note that the second head of MUSE.2 (trained to predict hiring and application interactions alike) is only slightly outperformed by the first head of MUSE.2 regarding its recall on the hiring interactions (recall@10 = 28.4, vs 29.1 for the first head). The key result is that the top head of MUSE.2 (built on the top of the first and second head and trained to predict the hiring interactions) manages to improve on MUSE.1 by about 2 percentage points for all recall levels regarding the hiring interactions. A tentative interpretation for this improvement is that the internal representation (shared by both heads of MUSE.2) is more representative of the job seekers and job ads than that of MUSE.1, since it leverages multi-tasking to learn from more data. MUSE.2 also improves (in terms of recall on hires) compared to MUSE.1.Applications, which corresponds to using the single-head MUSE.0 structure but pooling both hires and applications for learning.

Table 3.2: Comparative results of Muse and Xgb on the PES dataset: recall@$\{10, 20, 100, 1000\}$, overall training time and recommendation time per job seeker (in seconds).

Recall@	Xgb	Muse.0	Muse.1	Muse.1.Applications	Muse.2
10	26.83	22.88	28.3	28.0	30.1*
20	35.59	31.55	38.0	37.8	40.2*
100	58.88	53.80	61.7	62.1	63.2*
1000	86.47*	82.13	-	-	-
Train.	1.83h	7.7h	8.3'	38'	1.25h
Recom.	1.4"	0.0004"	0.018"	0.018"	0.02"

MUSE.0 ablation studies The merits of the MUSE.0 architecture are further investigated using ablation studies, aimed to determine the contribution of a standalone module (geographical, skills, general) to the recall performance (Table 3.3, left). The complementarity of the modules is also examined by removing a single module from the overall architecture (Table 3.3, center: all modules but one).

These results confirm the importance of the geographical module (standalone recall@100 circa 15%; loss in recall@100 circa 14% when omitted). The skills module has a lesser impact (standalone recall@100 circa 4%; loss in recall@100 circa 2% when omitted). More surprising is the impact of the general module (standalone recall@100 circa 34%; loss in recall@100 circa 7% when omitted). Its standalone performance suggests that it contains a larger share of the data information compared to the other modules. On the other hand, the moderate loss suffered when removing the general module suggests that this information is partially redundant with that of the other modules (note that the skill module also has access to the occupational profile of job seekers/job ads). Finally, the overall performance of MUSE.0 (recall@100 = 53.8) is close to the sum of the performances of its modules ($15.43 + 34.79 + 4.80 = 55.02$), demonstrating their complementarity.

Table 3.3: Muse.0: Impact of the three geographical, skills and general modules on the recall@100 through ablation studies. Left: module standalone. Right: Muse.0 without this module.

	Single module			All modules but one			Muse.0 (all modules)
	Geo	Gal	Sk	Geo	Gal	Sk	
R@100	15.43	34.79	4.80	39.97	47.28	51.96	53.80

Heterogeneity in terms of recall Tables describing the recall@10's heterogeneity broken down by selected subgroups are provided in Appendix B.

At the job seeker level, the most educated job seekers (five years or more years of educational achievements) have a significantly lower recall than the overall population (23.5 compared to a population mean of 30.4). Similar findings hold for the job seekers' level of qualification: the recall@10 for executives is 22.6.

At the pair level, the algorithm's performance is better for matches that are close geographically than for those farther away (for instance, the recall is 47.75 when a job seekers' future workplace is in same zip code, 39.7 when it is less than 5 kilometers away, against 30.4 on average). Pairs in the same occupation (at the level of the ROME nomenclature of granularity 14) are also easier to correctly rank than those that aren't (42.9 and 16.89 recall@10 respectively).

These differences could be interpreted in terms of the intrinsic difficulty of the recommendation problem: ensuring high recall on a population of job seekers who find jobs in a narrow geographic and occupation radius is easier than for job seekers who are mobile and less predictable in terms of occupation choice. Another interpretation could be in terms of statistical biases: in that case, algorithmic development reweighing minority classes, or tailored to some sub-populations, may be able to improve on a one-size-fit-all algorithm.

These results shouldn't be over-interpreted, as they do not fully answer the main question of interest - identifying *for whom* the algorithm performs best or worse. First, because recall (on hires) is not necessarily an ideal measure for algorithm performance (as will be discussed in Chapter 4), and because this quantity can only be assessed on job seekers who have been hired (who differ from the population of job seekers as a whole, as described in Section 2, Table 2.12). Second, even though one's future job may not be ranked first, the ads ranked above said job could also be relevant. Thus, we can not fully conclude by comparing two populations in terms of recall that the algorithm is less relevant for a population than another, since recommendation relevance also depends on the size of the pool of relevant job ads for a category of job seekers.

3.3 Partial conclusion

This chapter described MUSE, an algorithmic approach to cold-start recommendation, and bench-marked it in terms of an offline metric of interest (the recall@k on hires) with respect to the state of the art. These results will be complemented by the results of field experiments collecting MUSE's assessment by job seekers (Chapter 5). Using MUSE variants as a backbone, we will proceed to question whether the training objectives and metrics used here where the right ones - in terms of value alignment with job seekers' interests (Chapter 4), of accounting for congestion (Chapter 6), and discuss some underlying fairness issues (Chapter 7). Before moving on to these topics, beyond the input data limitations in the *France Travail* setting noted in Chapter 2 (the main ones being: not using "partner offers"; selection bias when training from hires; representativity of hires used for training compared to the global labor market; lack of access to clicks), let us discuss some perspectives for further work to improve on MUSE.

First comes the issue of scaling the model nation-wide: should this be done using separate region-wide models, pooling all nation-wide data for training, or might intermediate architectures enabling domain adaptation and multi-tasking be relevant?

Second, it is unclear whether a "one-model-fits-all" approach as adopted here can deliver appropriate results for all kinds of job seekers. Job seekers with different backgrounds may be sensitive to different aspects of jobs, such as geographic proximity compared to close fit to one's own profile and qualifications. [Sch+17] contrasted the models learnt from two datasets, one relative to PhD graduates and another relative to job seekers with lower qualification, showing for instance that geographic distance played a different role. While the MUSE architecture formally enables a different weighting of different fit aspects for different job seekers' profiles, alternative architectures - such as a a divide-and-conquer approach, or using higher-level interactions and multiplicative architectures - may yield further improvements in that regard.

Third, due to the availability of quality tabular data while text quality on job seekers' side greatly varied, the modeling of textual information in our architecture was minimal (a TF-IDF followed by singular value decomposition plugged in among

other input features). Advances in natural language processing could be leveraged to improve different stages of the recommendation process (improved representation of textual data, missing data completion based on textual complements, skill extraction, generating fit assessments of job seeker - job ad pairs for MUSE.2).

Fourth, due to data limitations, MUSE was not tailored to account for other sources of data which, if available, might greatly enrich recommendations - for instance, using job seekers' previous clicks, which would call for the use of collaborative filtering elements in MUSE (and adaptations for the cold start case).

Finally, the MUSE architecture does not take into account how issuing relevant individual recommendations might depend on labor market context (the pool of available job ads and of other job seekers). Listwise rather than pair-wise LTR models, graph-based models or reciprocal recommendation approaches might be appropriate to tackle this challenge.

Part II: Job recommendation beyond accuracy

Chapter 4

Value alignment

So far, we have taken for granted the target metric for algorithm evaluation (the recall@k measured on hires in Chapter 3). Yet, this choice of target for the learning process may be questioned, in the broader context of the variety of algorithms deployed by PESs [Bro23; Gut+19] or proposed in the machine learning literature [FC21; DB21; Mas+22]. These ML algorithms are adapted to different observational contexts, and often primarily aim to predict the success of a match (the likelihood of a click on an ad, applying for a job, of being hired). Regardless of the specific goal, the core objective remains the same: establishing a measure of closeness between job openings and job seekers. Job recommendations are based on identifying the ads that are the best match *in some sense* for each individual.

In this chapter, we address three main questions. First, we examine the variability of results generated by different recommendation algorithms. Are the job rankings produced by these algorithms similar? How important are the differences? If significant variations exist, our second question concerns the goal that these algorithms should aim to optimize: what objective best aligns with that of job seekers[1]? Our third question centers on identifying the needs of job seekers an algorithm should meet. Should it replicate job seekers' behavior, enabling them to carry out searches more efficiently than they could themselves, or should it uncover relevant job opportunities that job seekers might overlook [BKM19; Alt+22]?

We address the first questions by a study of two algorithms. The first is an expert system emphasizing the fit between job seekers' search parameters and job ads, measured by a matching score denoted $\mathcal{U}$ ranging from 0 (no search criteria met) to 1 (all search criteria met). The other one is a machine learning algorithm that predicts hirings based on job seekers' and job ads' characteristics (more precisely, the MUSE.0 algorithm presented in Chapter 3). We first show that this algorithm's predicted scores are indeed related to an applicant's chances of being hired, and calibrate it into a hiring probability $\mathcal{P}$. For each job seeker i, we identify the best ads according to $\mathcal{U}$ and to $\mathcal{P}$, along with their respective scores or probabilities. The differences are striking. For more than half of the job seekers, their top-$\mathcal{U}$ ad has a $\mathcal{P}$-rank over 381; conversely, the top-$\mathcal{P}$ ads' $\mathcal{U}$-ranks are greater than 3093 for

[1] Our discussion will focus on alignment with job seekers' objectives. Note that the labor market is a multi-stakeholder setting where the interests of other social groups or institutions (recruiters, a PES, society) may not necessarily align with those of individual job seekers.

more than half of the population. Moreover, the chances of being hired with the top-$\mathcal{U}$ ads (1.5% on average across job seekers) are much lower than the chances of being hired with the top-$\mathcal{P}$ ads (6%). Similarly, the best job ads in terms of hiring opportunities are often less aligned with search criteria (median $\mathcal{U}$ scores of less than 0.5 for the top-$\mathcal{P}$ ads, compared to almost 1 for the top-$\mathcal{U}$). In essence, the rankings differ significantly, emphasizing that they each capture different dimensions of the search process.

The observed differences underscore the importance of thinking about the ideal goals that these algorithms should optimize. To address this second question we construct a straightforward theoretical model of a job seeker's application behavior incorporating concepts from [CS06; HHA10; GS21]. In this model, the expected utility plays a central role, encompassing two essential dimensions: the job's utility (U) and the application's success probability (p). The model distinguishes between an application's probability of success "as perceived by the job seeker" and its true success probability. It reveals that the actual success probability depends on the job seeker's utility and the match's value to the firm. There is no inherent reason to assume that the perceived and true success probabilities of an application are the same.

Different recommender systems can be positioned in relation to this expected utility objective. Algorithms focusing on matching job ad characteristics with search parameters, such as the expert systems used by several national PESs, primarily consider the utility a job seeker would derive from a job opening. Algorithms based on hiring prediction primarily emphasize just one of the two fundamental components of expected utility. Meanwhile, algorithms centered on job applications align with expected utility as perceived by job seekers, approaching objective expected utility only if job seekers accurately estimate their hiring probabilities.

Our second empirical finding supports the application decision implied by our model, confirming that both job utility (U) and the likelihood of being hired (p) contribute to the decision to apply for a job. We analyze job postings that job seekers chose to apply to among the ones they initially clicked on, and our results indicate that both $\mathcal{U}$ and $\mathcal{P}$ have a significant impact on this decision. This finding is crucial as it underscores that relying solely on one of the two algorithms would overlook an essential aspect of the decision-making process. It reinforces the idea that effective algorithms should be grounded in a representation of job seeker preferences and behaviors. Currently, both algorithms, based on $\mathcal{U}$ and $\mathcal{P}$, are guided primarily by data or available statistical methods, but they lack a representation of search behavior.

Our third question concerns the job seekers' needs that a recommendation algorithm should address. The literature suggests two underlying ideas. First, an algorithm can efficiently identify relevant job openings that a job seeker might have uncovered on their own, lowering search costs [LHR23]. Second, it can help job seekers *discover* job opportunities they might otherwise miss [BKM19; BKM22; Alt+22]. Identifying a relevant job posting involves assessing whether the job aligns with the seeker's preferences and the likelihood of a successful application. These evaluations demand effort and determine the number of job applications. The first idea implies that the algorithm's role is to reduce these costs for job seekers, en-

couraging them to apply to more positions. The second idea acknowledges that assessing key variables associated with job openings can be challenging, especially in estimating application success probabilities, corresponding in our model to real and perceived success probabilities. Job seekers may make errors in choosing which jobs to apply to, particularly concerning their chances of success [CK20]. This is consistent with the economic literature on job search, suggesting that there are often overlooked job postings in related fields. In this context, the algorithm's role is to broaden the search to include relevant job openings that job seekers may not have explored.

Our third empirical finding explores job seekers' needs by comparing recommended job ads with their actual applications, revealing substantial heterogeneity. While some job seekers widen their search beyond initial choices to improve success chances, many have yet to explore the full range of possibilities. We also evaluate the potential benefits of the best recommendations from different recommender systems (based on $\mathcal{P}$, $\mathcal{U}$, or their product $\mathcal{PU}$) compared to where job seekers actually apply. This highlights substantial but highly heterogeneous gains across the population. However, gains relative to actual applications using the $\mathcal{P}$ or $\mathcal{U}$ criteria are sometimes associated with losses relative to the symmetric criterion ($\mathcal{U}$ or $\mathcal{P}$). This is not the case when using expected utility ($\mathcal{PU}$) as the guiding principle. In this context, recommender systems prove valuable in helping job seekers uncover and recognize relevant opportunities they might otherwise miss.

In summary, our study underscores the importance of identifying essential quantities in creating a high-performance algorithm, specifically job utility (U) and the likelihood of a successful application (p). The latter should be derived from available data, a task that ML tools excel at, especially in predicting job application success. However, identifying job utility is challenging due to the lack of direct observation.

Another prerequisite involves understanding how job seekers assess their hiring prospects. Our findings reveal that recommender systems can help fulfill the need for discovering new opportunities, addressing the challenge job seekers face in accurately assessing key parameters describing job openings. Replicating observed job seeker behavior, such as their applications for jobs, may inadvertently replicate their errors.

Related literature This paper relates different strands of the economics literature. The first, in the context of online job search [KM14; Kir22], is related to the impact of recommendations on frictions in the labor market. Several studies show that (automatic) suggestions designed to extend job seekers' search perimeter to alternative occupations have an effect on interviews [BKM19; LRB20] and future job outcomes [BKM22; Alt+22; Beh+22]. However, some find only modest effects [LHR23]. Our contribution in this respect is to study a realistic ML algorithm relying on extensive data. One key insight of our analysis is that recommender systems often focus on a precise objective - typically improving the chances of a match - which may differ from the job seekers' objectives. This disconnection between the two can result in substantial losses, as some individuals may focus their search on vacancies far from their preferences following the recommendations. Additionally,

our analysis stresses the importance of accounting for behavioral aspects of job search [Bab+12; CK20; Alt+18], such as biased expectations of the chances of success of different strategies and/or perceptions of the market [MST21; MS23]. In an experiment, [Fie+23] show that lowering users' psychological cost of initiating job applications has a strong effect on applications. These psychological costs might prevent job seekers from applying to high-return vacancies. The paper is also related to the importance of preferences for various job attributes and how ignoring their heterogeneity can lead to frictions [MP17; BC22; FNO22]. By questioning how an algorithm's objectives align with its end users', our enquiry is inspired and tied to literature on the so-called value misalignment problem in economics and computer science [ZH20; KMR23; Kas24].

4.1 Two job recommender systems

Our sample contains job seekers registered at *France Travail* and the available job ads on the PES's website in the former French Rhône-Alpes region from ISO weeks 1 to 48 of 2019. The number of unique job seeker search sessions (resp. job ads) is 1,181,902 (resp. 516,776); on average, 610,986 job seekers (resp. 129,642 job ads) are active a given week. We observe 75,744 matches (hires) in the data. Observations from week 1 to 43 of 2019 are used as a training set (representing 66,914 matches); while weeks 44 to 48 (representing 8,830 matches) are used as a test set.

Denote by $M^*_{i,j} \in \{0,1\}$ the latent variable which takes value 1 when there is a match for a pair job seeker-firm (i,j), conditional on i having applied to j. The observed hiring dummy between i and j is $M_{i,j} = M^*_{i,j} A_{i,j}$, where $A_{i,j} = 1$ if job seeker i applied to job posting j, and 0 otherwise.[2]

Definition of $\mathcal{P}$ The backbone algorithm for hire prediction is (a slightly older version of) MUSE.0, using features described in Appendix C.1. The model's performance on a test set achieved a recall@100 of 57.5. Inspired by [Che+18]'s approach, we also check whether the algorithm's predictions $S_{i,j}$ can predict hiring, by checking whether $\mathbb{P}(M_{i,j} = 1 | S_{i,j}, A_{i,j} = 1) \neq \mathbb{P}(M_{i,j} = 1 | A_{i,j} = 1)$. We study the history of applications made by job seekers to vacancies, viewing the chronologically ordered sequence for an individual i_0, $1(i_0), 2(i_0), \ldots, j^{max}(i_0)$ as a sequential search model and analyzing it as a discrete duration model (see, *e.g.* [TS16]), where we model the hazard rate as a known, assumed logistic[3], transformation Λ of the score

$$\mathbb{P}\left(M_{i,j(i)} = 1 | M_{i,1(i)} = 0, \ldots, M_{i,j-1(i)} = 0\right) = \Lambda(\alpha_{r(i,j(i))} + \beta S_{i,j(i)}), \tag{4.1}$$

where $r(i,j)$ is the rank of the vacancy j among the set of applications, and $S = (S_{i,1(i)}, \ldots, S_{i,j^{\max}(i)})$ denotes the sequence of scores of the job ads.

[2]This sample also contains clicks on job postings and applications. The latter are not used for the training of MUSE.0, but later for the calibration of the hiring probability and the empirical validation of model (4.6) in Section 4.3.

[3]The logistic form is a strong assumption given that the triplet ranking loss used to train MUSE.0 (equation 3.1) does not guarantee that scores are comparable between job seekers.

Under strong assumptions (see Appendix C.2), including selection on observables and the score being a sufficient statistic, we can interpret probability $p(i,j) = \mathbb{P}\left(M_{i,j} = 1 \,|S_{i,j}, A_{i,j} = 1\right)$ as $\mathbb{P}(M^*_{i,j} = 1|S_{i,j})$. Bearing in mind that the purpose of the score is solely to rank job postings for each job seeker, such a procedure transforms it into a hiring probability (while maintaining identical rankings).

The estimated coefficient of β, of value 0.061, is significantly positive at the 1% level. This finding is robust to the different specifications including application and interview ranks effects (resp. 0.038 and 0.047) (see Table 4.1). Overall, this validates the content of the MUSE.0 score $S_{i,j}$ in terms of its potential to reflect the hiring chances. From now on, we only consider the transformation of the score $\mathcal{P}(i,j) := \Lambda(0.061\ S_{i,j} - 4.113)$, which is a signal on the hiring probability $p(i,j)$.

Table 4.1: Estimates of the calibration model parameters

Method	(1)	(2)
Score $S_{i,j}$	0.061*	0.038*
With application rank	No	Yes
AIC	28,040	25,116

Notes: On a half of the job seekers present in the test sample (weeks 44-48 of 2019): 79,097 applications, 3,469 matches, 34,255 job seekers. Significance levels: 1% : *. A robustness check including dummies for the ranking of the application j in the list of applications of job seeker i is provided.

Definition of $\mathcal{U}$ *France Travail* has developed a matching algorithm based on WCC Elise (see [Gut+19]), which is used to suggest relevant vacancies to job seekers. Each criterion is associated with a weight w_k and the final matching score is the weighted sum of each single fit between the criteria of applicants and the job ad's content (each fit measure $c_k(i,j)$ takes values between 0 and 1). The score used at the PES involves some nonlinearities ignored here for simplicity. The simplified version we use is:

$$\mathcal{U}(i,j) = \sum_{k=1}^{K} \frac{w_k}{\sum_{k=1}^{K} w_k} c_k(i,j), \tag{4.2}$$

where the set of weights $\{w_k\}_{k=1,\dots,K}$, presented in Table 4.2, is the same for the whole population.[4] The weights are determined by expert knowledge but the score's relevance to the description of job seekers' utility will be verified empirically in the following (Section 4.3).

[4] [Fie+23] also use a similar specification and characteristics to evaluate the value of a vacancy from the point of view of job seekers, see their Table 3 for details.

Criterion	Given weight w_k
Occupation	1000
Skills in occupation	1000
Geographic mobility	300
Reservation wage	200
Diploma	100
Working hours	100
Driving license	100
Languages	100
Years of experience in occupation	100
Duration and type of contract	10

Table 4.2: Weights defining $\mathcal{U}$

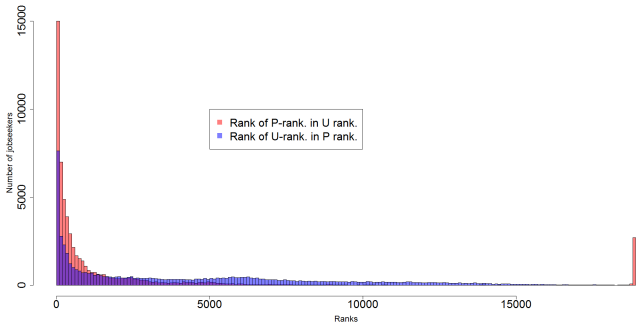
4.2 Do the two recommender systems recommend similar job ads?

For the sake of tractability, we further restrict the sample and focus hereafter on the subsample of job seekers and ads whose main sector is transportation and logistic in the former French Rhône-Alpes region from ISO weeks 44 to 48 of 2019. This sector contains 60,299 job seekers and 18,873 job openings. We will refer to the $\mathcal{P}(i,j)$-based rankings of job ads as *$\mathcal{P}$-rankings* or hire-based rankings, and, those based on $\mathcal{U}$ as *$\mathcal{U}$-rankings* or preference-based rankings.

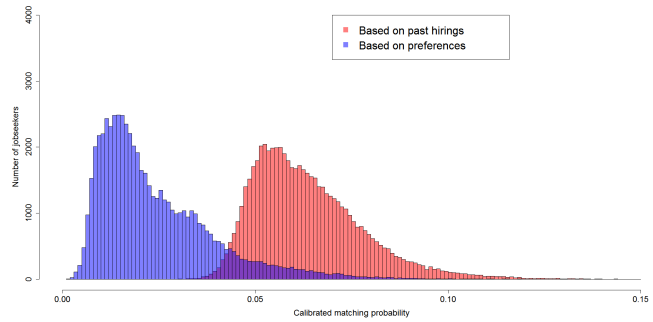
Firstly, we investigate the proximity of the $\mathcal{P}$- and $\mathcal{U}$-rankings, revealing substantial differences. Secondly, focusing on the quantitative dimension of the scores, we show the best recommendations according to the $\mathcal{U}$ and $\mathcal{P}$ rankings would yield substantially different $\mathcal{U}$ and $\mathcal{P}$ scores, underscoring the quantitative relevance of these ranking differences in terms of outcomes.

Rankings $r^{\mathcal{P}}$ and $r^{\mathcal{U}}$ are very different. We compare for each job seeker i the optimal ad based on the $\mathcal{P}$-ranking, denoted $j^{\mathcal{P}}(i)$, and the optimal job ad following the $\mathcal{U}$-ranking, denoted by $j^{\mathcal{U}}(i)$. We first compare the respective ranks of these optimal ads: the rank of $j^{\mathcal{P}}(i)$ in the $\mathcal{U}$-ranking: $r^{\mathcal{U}}(i, j^{\mathcal{P}}(i))$, and symmetrically the rank of $j^{\mathcal{U}}(i)$ in the P-ranking: $r^{\mathcal{P}}(i, j^{\mathcal{U}}(i))$. Figure 4.1(a) shows the distribution of these ranks. Top-$\mathcal{U}$ and top-$\mathcal{P}$ ads match only for a small minority of job seekers. For most, the ranks considered are very large. The median of $r^{\mathcal{U}}(i, j^{\mathcal{P}}(i))$ is 381 (top 2%) and that of $r^{\mathcal{P}}(i, j^{\mathcal{U}}(i))$ is 3,093 (top 16%). The two rankings are thus substantially different.

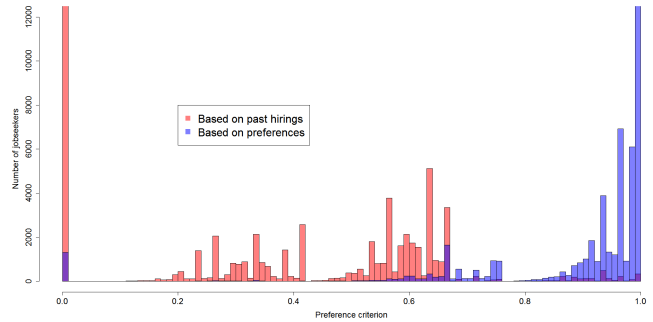
Differences in $\mathcal{U}$ and $\mathcal{P}$ scores between top-$\mathcal{P}$ and top-$\mathcal{U}$ job ads Figure 4.1(b) shows the distribution of the hiring probabilities for the two job ads: $\mathcal{P}(i, j^{\mathcal{P}}(i))$ and $\mathcal{P}(i, j^{\mathcal{U}}(i))$ along with their difference. The median value of the maximum hiring probability for each individual $\mathcal{P}(i, j^{\mathcal{P}}(i))$ is 0.06, sharply contrasting with the hiring probability for the optimal ad according to the adequacy criterion (0.015). The difference between the hiring probabilities is substantial, with



(a) Distributions of the ranks of the best recommendations



(b) Distributions of hiring probabilities



(c) Distributions of preference score

Notes: 60,299 job seekers whose main sector is transportation and logistic (Rhône-Alpes region, ISO weeks 44-48 of 2019). 18,873 job ads are available at that period in this sector. *Upper panel*: Distributions of the ranks of the best $\mathcal{P}$ and $\mathcal{U}$ recommendations in each other's rankings. The bunch at the right gathers top-$\mathcal{P}$ ads ranked after 18,800 according to $\mathcal{U}$ as they have a preference score of 0. *Middle panels*: Histograms of the hiring probabilities for the best recommendations in both systems. *Lower panels*: Histograms of the preference score for best recommendations in both systems.

Figure 4.1: Comparison of the best recommendations in the two rankings: ranks, hiring probabilities, and preference score

a median value of 0.04. Although the probability of hiring from the best ad in the $\mathcal{P}$-ranking is higher than the probability of hiring from the $\mathcal{U}$-ranking, it is worth noting that this probability in absolute terms is not so high; we will come back

to this when we shall study the job ads to which job seekers apply. Even more pronounced differences arise in the matching scores $\mathcal{U}(i, j^{\mathcal{U}}(i))$ and $\mathcal{U}(i, j^{\mathcal{P}}(i))$. As shown in Figure 4.1(c), the distribution $\mathcal{U}(i, j^{\mathcal{U}}(i))$ has a substantial mass at 1 (median 0.98), indicating that for many job seekers there are ads that meet all their criteria. Conversely, for the optimal job ad according to the hiring probability, there is a significant mass at zero (median 0.46). Figure 4.1 thus shows that a switch from $j^{\mathcal{U}}(i)$ to $j^{\mathcal{P}}(i)$ would be likely to improve job-finding chances substantially, but might compromise the suitability of the job concerning the job seeker's preferences.

4.3 Does a recommendation algorithm dominate the other regarding job seekers' objective?

The important question is whether these algorithms, designed to nail different objectives, align with the job seekers' (JS) objectives, a problem known as value misalignment. To investigate this question, we consider a simple model of job seekers' application behavior derived from [HHA10].

The model has two stages. In the first stage, job seekers identify vacancies and decide to apply. There are two sources of imperfect information about the job: 1) job seekers do not know whether the job is the right one for them, and 2) they do not know whether they are a good fit for the company. In the second stage, more information is revealed during interviews, with job seekers selecting from their applied vacancies and companies choosing from the received applications.

This section outlines the main features of the model and its first stage, and the following section describes the second stage.

A matching model with an application stage Consider a market of I job seekers and J firms of observed types $x \in \mathcal{X}$ and $y \in \mathcal{Y}$.[5] Let the utility of a job seeker i of type x (resp. a firm j of type y) who is hired at a firm j of type y (resp. who hires a job seeker i of type x) be

$$U_{i,j} = \widetilde{U}(x, y) + w_{x,y} + \varepsilon_{i,y}, \tag{4.3}$$

$$V_{i,j} = \widetilde{V}(x, y) - w_{x,y} + \eta_{x,j}, \tag{4.4}$$

where $\widetilde{U}(x, y)$ and $\widetilde{V}(x, y)$ are the nontransferable parts of the utility depending only on observable types, $w_{x,y}$ is the wage posted by a firm of type y for a candidate of type x, and $\varepsilon_{i,y}$ (resp. $\eta_{x,j}$) is a mean-zero idiosyncratic taste shock of candidate i for firms of type y (resp. of firm j for a candidate of type x).[6] To simplify the notations, we note $U(x, y) = \widetilde{U}(x, y) + w_{x,y}$ and $V(x, y) = \widetilde{V}(x, y) - w_{x,y}$.[7]

[5] These types should be denoted x_i and y_j respectively, but we remove the subscripts for ease of notation.

[6] In general, this error term would depend on the index j of the firm and not only on its observed type y, but similarly to, *e.g.*, [CS16; Gal18], we make this restriction for simplicity.

[7] We normalize the outside options respectively to $\varepsilon_{i,0}$ and $\eta_{0,j}$.

To model the frictions arising from the application step, we assume that job seekers have ex-ante imperfect information about the utilities (4.3)-(4.4). Information about $\varepsilon_{i,y}$ and $\eta_{x,j}$ becomes available only after applying. Additionally, job seekers' expectations include irrelevant information $\delta_{i,y}$ about job posting of type y.[8] Job seekers' information set $\mathcal{I}_1$ is generated by x, y, and $\delta_{i,y}$. We assume for simplicity that we are in a *large market*, *i.e.*, that there are infinitely many jobs of type y and profiles of type x.

We consider a process in two stages, starting with a **Stage 1** where job seekers decide the types y of job openings they apply to. They incur a cost c to apply for a job and a cost r if there is *in fine* no match, encompassing psychological costs. The fact that i applied for a job at j is denoted by $A_{i,j} = 1$. Job seeker i of type x applies for the job in firm j of type y if its expected utility is greater than its reservation utility $U_{x,0}$ that he or she gets if there is no match:

$$\mathcal{E}(U_{i,j}|\mathcal{I}_1) \geq U_{x,0}. \tag{4.5}$$

The decision to apply is based on weighing the gains incurred in case of a hire against the costs of applying plus the costs of rejection or refusing the offer in the end. The set $\mathcal{C}_i \subseteq \mathcal{Y}$ denotes the types of job ads job seeker i considers in Stage 1.

We consider applications decisions in Stage 1 as independent across vacancies - see Appendix C.3 for sufficient conditions on $\delta_{i,y}$ ensuring this is the case. In this context, using (4.5) and $\mathcal{E}(U_{i,j}|\mathcal{I}_1, M^*_{i,j} = 1) = U(x, y) + \delta_{i,y}$, the first stage decision rule is:

$$\psi(x, y) := U(x, y) - U_{x,0} + r - \frac{c + r}{\pi(x, y)} \geq -\delta_{i,y}, \tag{4.6}$$

where $\pi(x, y)$ is the subjective hiring probability perceived by the job seeker. Let us discuss the latter now.

Subjective probabilities $\pi(x, y)$ may diverge from objective ones $p(x, y)$ if job seekers do not hold rational expectations, *i.e.*, have expectations that systematically differ from the realized outcomes and do not efficiently use the available information. Despite this, equation (4.6) highlights that there are two key factors entering the decision to apply, one related to the utility of the job $U(x, y)$ and one related to the probability of the match $\pi(x, y)$. Thus in this model they both matter for job seeker's application behavior.

Our interpretation of the two algorithms of Section 4.2 is that they both actually capture distinct parts of relevant information. In equation (4.2), $\mathcal{U}(i, j)$ is a signal about $U(x, y)$ and $\mathcal{P}(i, j)$ in equation (4.1) can be interpreted as a signal about the hiring probability, thus also connected to $\pi(x, y)$.[9,10]

[8] Alternatively, one can view $\delta_{i,y}$ as an error in the decision to apply in (4.6).

[9] Although we maintain this interpretation of $\mathcal{U}$ and $\mathcal{P}$ in the sequel, we might consider more broadly that $\mathcal{U} = \mathcal{U}(U, p)$ and $\mathcal{P} = \mathcal{P}(U, p)$. The important point is that U and p both matter, and that $\mathcal{U}$ and $\mathcal{P}$ represent two different combinations of them. The following section 4.4 will shed more light on the difference between U and p.

[10] In section 4.4, about stage 2 of the model, we derive the true expression of the hiring probability $p(x, y)$ and discuss its link with $\pi(x, y)$.

Do $\mathcal{P}$ and $\mathcal{U}$ truly matter for job seekers? We want to test whether the latter interpretation about the information contained in $\mathcal{U}$ and $\mathcal{P}$ is consistent with the application behavior described in equation (4.6).

To assess this, we consider the set of vacancies on which job-seekers have clicked and estimate a logit model with fixed effects for the binary decision of applying to a job opening. The model takes the form, based on (4.6):[11]

$$\mathbb{P}\left(A_{i,j}=1\,|\text{click}_{i,j}=1, c_k(i,j), \mathcal{P}(i,j), X_i, Y_j, f_i\right) = \Lambda\left(\alpha\sum_{k=1}^{K}\gamma_k c_k(i,j) - \frac{\beta}{\mathcal{P}(i,j)} + f_i\right), \tag{4.7}$$

where f_i is an individual fixed effect. We first consider fixed $(\gamma_k)_{k=1}^K$ and α is estimated. While we do this for $\gamma_k = w_k \big/ \sum_l w_l$ with either w_k chosen by the PES or uniform w_k, these fixed γ_k naturally raises questions about their relevance and their ability to truly reflect the preferences of job seekers.[12] Using the available data on applications we thus also estimate these weights. In such a case $(\gamma_k)_{k=1}^K$ are estimated and $\alpha = 1$ (see, *e.g.*, [HHA10; CHL23] for similar estimation in the context of the marriage market).

The estimation results are displayed in Table 4.3. The application probabilities increase with the utility score $\mathcal{U}(i,j)$ and decrease with the inverse of the hiring probability $\mathcal{P}(i,j)$. This confirms that $\mathcal{U}$ and $\mathcal{P}$ capture different dimensions of the search process[13], and suggests that relying on either $\mathcal{U}$ or $\mathcal{P}$ alone would miss part of the decision process and deviate from job seekers' objectives. It prompts questions about the best use of $\mathcal{U}(i,j)$ and $\mathcal{P}(i,j)$. What is the relevant objective to train a recommender system? What about training an algorithm which would identify vacancies job seekers apply to, thereby reproducing their behavior? Addressing these questions hinges on establishing the link between the perceived hiring probability $\pi(x,y)$ and the true hiring probability $p(x,y)$ that we now focus on.

4.4 Should recommender systems reproduce job seekers' behavior?

Up to this point, we have not explicitly addressed the informational content of the perceived hiring probability. In this section, we close the previous model to derive the form of the hiring probabilities, and discuss the implications of potential disparities between perceived and actual probabilities. Our model has the following **Stage 2**. We consider that some information about the tastes of both sides is revealed during the interview. Then, the matching is based on the maximization of the utilities of both parties, conditional on the first stage interview. Job seeker i maximizes his or her utility over the set of vacancies $\mathcal{C}_i$ they applied for: $\max_{y\in\mathcal{C}_i}\{U(x,y)+\varepsilon_{i,y}\}$.

[11] See equation (9) in [HHA10] or [LRR21].

[12] An ongoing experiment [Ban+22] consists in amending such weighting of the preferences to substitute values chosen by the job seekers themselves.

[13] We emphasize in Section 4.2 that both $\mathcal{U}$ and $\mathcal{P}$ cannot be considered as two noisy measures of the same underlying score.

Table 4.3: Estimates of the model of application on job ads

	Estimate	Std. error	Estimate	Std. error	Estimate	Std. error	Estimate	Std. error
Utility score $\mathcal{U}(i,j)$ (α)	1.180**	0.155						
Unif. Utility score $\overline{U}(i,j)$ (α)			0.883**	0.137				
Sector occupation					-0.068	0.170		
Occupation					0.693**	0.199	0.626**	0.104
Skills					0.190†	0.114	0.191†	0.114
Reservation wage					0.254**	0.082	0.255**	0.082
Languages					-0.011	0.229	-0.011	0.229
Experience in occ.					-1.086**	0.340	-1.086**	0.340
Diploma					0.304*	0.119	0.305*	0.119
Driving license					0.111	0.097	0.110	0.097
Geographic mobility					0.666**	0.216	0.669**	0.215
Duration					0.146	0.098	0.146	0.098
Type of contract					0.015	0.068	0.015	0.068
Inverse of $\mathcal{P}(i,j)$ (β)	-0.030**	0.004	-0.031**	0.004	-0.028**	0.004	-0.028**	0.004
Avg. indiv. Fixed effects	-1.342		-1.372		-1.357		-1.355	
Nb. observations	17,865		17,865		17,865		17,865	

Estimation of equation (4.7) modeling applications as a fixed effect logit model
Notes: Our sample is the set of all applications for job seekers in the transportation and logistic sector during week 44 of 2019, leading to a hiring or not. Fixed effect estimation keeps 70,557 observations for 8,105 job seekers, and 869 of them applying at least once. Thus, 17865 observations are kept for estimation. Estimation of results for a logit panel without individual fixed effects are available in Appendix C.4. Results are robust to the different negative sampling strategies we considered. Significance levels: $< 1\% : **$, $< 5\% : *$, $< 10\% : \dagger$.

Firm j of type y chooses an offer among the set of types x of workers for which its type is maximizing their utility: $\max_{x\in\mathcal{X}} \{V(x,y) + \eta_{j,x}\}$.

The structural form of the hiring probability. Under further standard assumptions on the distributions of $\varepsilon_{i,\cdot}$ and $\eta_{\cdot,j}$, one can derive a closed form expression for the hiring probability $p(x,y)$. This expression is similar to the seminal one in [CS06] and [GS21],[14]

$$p(x,y) = \underbrace{p_{f,0}(y)e^{V(x,y)}}_{\text{Probability } y \text{ selects } x} \underbrace{p_{js,0}(x)e^{U(x,y)}}_{\text{Probability } x \text{ selects } y}, \tag{4.8}$$

where $p_{f,0}(y) = 1/(1 + \sum_{x'\in\mathcal{X}} e^{V(x',y)})$ and $p_{js,0}(x) = 1/(1 + \sum_{y'\in\mathcal{Y}} e^{U(x,y')})$ are the probabilities that a firm of type y and a job seeker of type x prefer to remain unmatched. The matches that occur depend on the utility U of the job for the job seekers and the value of the hire V for the firm. Moreover, the probability of hiring is directly related to the total surplus via formula (4.8).

There are thus two sources of uncertainty explaining p and π. The first is that job seekers have to assess the utility that they would get on the job, which might be well performed. The more important one, implying that p and π might be different, is due to the fact that job seekers have to assess V_{ij}, *i.e*, how one's profile will be valued by the firm.

Biased job seekers' perceptions about their chances to be hired on some vacancy $\pi(x,y) \neq p(x,y)$ entering in (4.6) may distort their applications. There might be under and over estimation of the hiring probability. An underestimation of the chances of recruitment $\pi(x,y) < p(x,y)$, may lead to *exclusion errors*, where job seekers discard ads y on which they consider have low selection chances

[14]See Lemma C.3.1 in Appendix C.3 which provides the exact form of this hiring probability.

$\psi(x,y)+\delta_{i,y}<0$, whereas these chances are in fact sufficiently high for a profitable application: $\widetilde{\psi}(x,y)+\delta_{i,y}\geq 0$ ($\widetilde{\psi}$ is the analogue of ψ replacing π with p). On the other hand, an overestimation of the chances of recruitment $\pi(x,y)>p(x,y)$ may result in *inclusion errors*, where job seekers apply to ads on which they consider have sufficient selection chances $\psi(x,y)+\delta_{i,y}\geq 0$ whereas the actual chances are low, making an application unprofitable $\widetilde{\psi}(x,y)+\delta_{i,y}<0$.

This means that job seekers might overlook or incorrectly select relevant job opportunities. Consequently, replicating job seeker behavior would reproduce these inclusion and exclusion errors. Conversely, learning from past hirings generates information on the hiring chances p for a given position. If well designed, such a recommender system could assist job seekers in *discovering* relevant job opportunities beyond their typical search area. For instance, based on $p(i,j)$ and $U_{i,j}$ one might consider the expected utility $p(i,j)U_{i,j}$ and rank job ads based on such a criterion rather than $p(i,j)$ or $U_{i,j}$.

Exploration or reproduction of job seeker's behavior? In this section, the focus broadens to the positions job seekers apply to. We consider a simple counting exercise: for each job seeker we examine the number of job postings with a higher score than the actual observed applications according to different recommendation algorithms. Let us stick with the two previous algorithms based on $\mathcal{P}$ and $\mathcal{U}$, and add a third one, reflecting the expected utility. It is obtained simply by considering the product $\mathcal{PU}$. This last ranking of job ads has the interest of being built from $\mathcal{P}$ and $\mathcal{U}$ and being close to the objective of job seekers, while avoiding the replication of their behavior and potential mistakes.

Table 4.4 presents the results, organized into several panels. The upper panel contains the quantiles of the numbers of existing vacancies with a higher score than where job seekers apply when considering each direction of exploration. The intermediate panel provides some features of the distribution of the different scores over the application set. In the lower part of the table, we consider the distribution of relative gains according to $\mathcal{P}$ (column 1), $\mathcal{U}$ (column 2), and $\mathcal{PU}$ (column 3) related to exploration in each of the dimensions $\mathcal{P}$ (upper panel), $\mathcal{U}$ (middle panel), and $\mathcal{PU}$ (lower panel).[15]

The main result is that, for each criterion, there are significant gains associated with exploration, but with notable heterogeneity. A segment of the population has numerous unexplored opportunities. For instance, concerning the expected utility criterion, for 25% of the population, there are at least 984 ads that would score higher than those the job seekers are currently applying for, while 25% have less than 17 such ads. This implies that some job seekers are already thoroughly exploring existing opportunities, while many have untapped possibilities. Similar patterns are observed for rankings based on $\mathcal{P}$ and $\mathcal{U}$, and it is noteworthy that, for the ranking based on $\mathcal{U}$, the opportunities seem even more abundant.

Unexplored job postings offer substantial gains compared to current applications. In the last panel, for example, gains in expected utility when searching in

[15]The information in the different columns is approximately linked by (1+col1)x(1+col2)=(1+col3).

Table 4.4: Recommender systems' ability to identify better job ads compared to application set.

	Applications	Job ads with better Matching probability ($\mathcal{P}$) than the application with best $\mathcal{P}$	Job ads with better Utility ($\mathcal{U}$) than the application with best $\mathcal{U}$	Job ads with better Expected Utility ($\mathcal{PU}$) than the application with best $\mathcal{PU}$
Nb. (quantile 0.75)	2	437	5,124	984
Nb. (median)	1	86	1,465	139
Nb. (quantile 0.25)	1	17	304	17
		Applications		
Maximum		0.044	0.399	0.018
Average		0.042	0.342	0.015
Minimum		0.040	0.332	0.013
Distribution of maximum gains from exploration among JS according to the three different criteria		Quantiles of percentage change wrt. applications		
Exploration according to $\mathcal{P}$				
Q25		20.8	-20.0	-3.3
Median		39.6	-62.3	-47.4
Q75		65.5	177.8	359.8
Exploration according to $\mathcal{U}$				
Q25		4.1	55.0	61.3
Median		-81.1	138.1	-55.1
Q75		-60.7	258.0	40.7
Exploration according to $\mathcal{PU}$				
Q25		3.5	49.4	54.7
Median		60.6	44.8	132.5
Q75		40.3	219.8	348.5

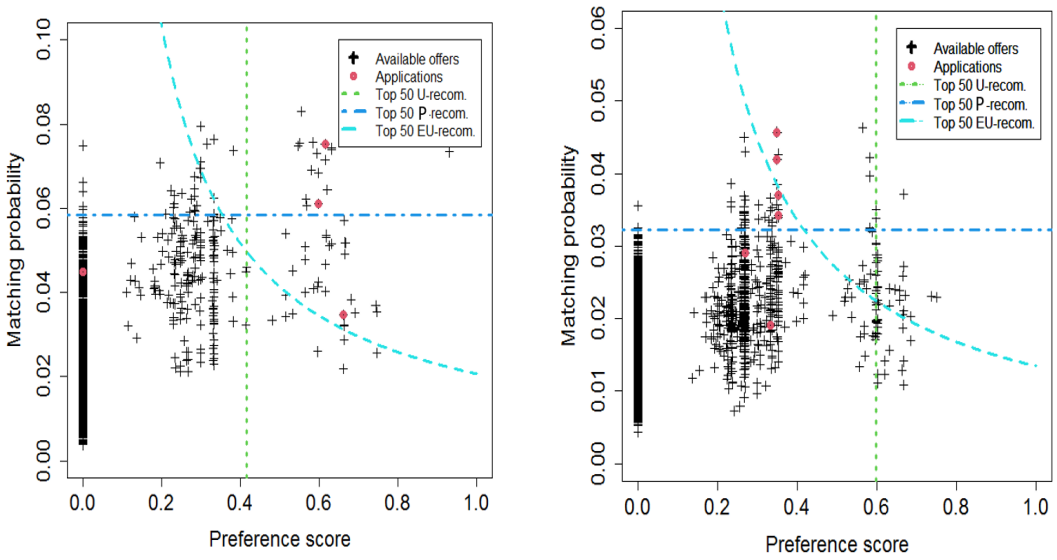
Notes: All quantities are defined at the job seeker level and then we report the median over the population of job seekers on our sample of applications for job seekers in the transportation and logistic sector during week 44 of 2019, leading to a hiring or not. This represents 9,965 applications and 5,252 job seekers. We consider the set of all 138,237 ads available this week. The recall on this subsample is respectively 33.89%, 54.93%, and 6.33% for the rankings based on $\mathcal{P}(i,j)\mathcal{U}(i,j)$, $\mathcal{P}(i,j)$, and $\mathcal{U}(i,j)$.

this direction would be at least 348.5% for 25% of job seekers and at least 54.7% for 75% of job seekers.

Another noteworthy result is that when searching in the directions of $\mathcal{P}$ and $\mathcal{U}$, gains are observed in both $\mathcal{P}$ and $\mathcal{U}$, but instances exist where gains in one dimension coincide with losses in the other, diminishing gains in expected utility. Note that recommendations from algorithms akin to those in [BKM19] or [Alt+22] can be seen as having the same spirit as exploration according to $\mathcal{P}$.

This exercise thus provides suggestive evidence of the idea that some job seekers are missing out on numerous job openings that would significantly increase their expected utility. Consequently, replicating the job seekers' search behavior would result in reproducing their appraisal errors. Such a recommender system would, therefore, overlook many ads that would be objectively relevant from the job seeker's perspective. Figure 4.2 provides a visual illustration of these findings: it represents the jobs considered by two job seekers in a utility-probability plane and shows the

impact of different recommender systems ($\mathcal{U}$, $\mathcal{P}$, and $\mathcal{PU}$) on their job selections.



Note: These are two job seekers of the logistic and transportation sector and their search behavior in week 44 of 2019. Black points are available job ads and red ones are the two job seekers' applications. The dark blue line represents the Top-50 cut-off corresponding to the hiring probability of the 50th best ad in the $\mathcal{P}$-ranking. The green line represents the Top-50 cut-off corresponding to the $\mathcal{U}$-rankings. The light blue line corresponds to $\mathcal{PU} = \underline{W}$, where $\underline{W}$ is the expected utility of the 50th best ad in the ranking according to the expected utility ("EU-recom").

Figure 4.2: Representation of the different selections between recommender systems and actual search behavior for two job seekers

4.5 Partial conclusion

Recommendation algorithms are rapidly gaining popularity, and many PES are planning to adopt them in the near future [Bro23]. However, there are various approaches to designing these systems [FC21], each following different paths and constraints, all rooted in training algorithms based on past observations to predict future matches, applications, or clicks. In this chapter, we explore the economic aspects of such algorithms, highlighting challenges in recommending jobs to job seekers.

Our main findings highlight that these algorithms, represented by two ideal cases, pursue objectives partially aligned with job seekers' goals, recommending substantially different vacancies. They rely on scores capturing fundamental aspects of job seekers' selection behavior, specifically job utility ($\mathcal{U} \leftrightarrow U$) and recruitment chances ($\mathcal{P} \leftrightarrow p$). Consequently, an algorithm exclusively focused on $\mathcal{U}$ or $\mathcal{P}$ would overlook valuable information for job seekers, a conclusion supported by our empirical analysis.

Our contribution underscores potentially important gains in exploring new opportunities in each direction ($\mathcal{U}$, $\mathcal{P}$, and $\mathcal{PU}$), although these gains are very het-

erogeneous across job seekers. Our analysis reveals that for a large share of job seekers, there is a significant number of jobs with higher expected utility ($\mathcal{PU}$) than the ones where job seekers apply and that the gains in expected utility would be quite substantial.

Consequently, designing a recommendation algorithm should be based on an objective that is close to that of job seekers, while avoiding the replication of their potentially biased behavior. Possible solutions include leveraging ML tools to identify the fundamental parameters P and U and constructing sets of recommendations from these two scores. Another option is to predict $U_{i,j}M_{i,j}$, which would identify the expected utility, rather than $M_{i,j}$ alone, which identifies the hiring probability. A crucial step involves determining job-related utility ($U_{i,j}$), while limiting misspecification. However, this step is complex as there is no straightforward way to quantify job-related utility, unlike for the recruitment $M_{i,j}$.

Chapter 5

Field experiments

As discussed in Chapter 1, assessing recommender systems “in the lab" (in terms of metrics such as recall) has limits. For instance, the recall (on hires) can only be computed on the population of job seekers who found a job in the past; even on this population, how this metric relates to job seekers' actual objectives is less than clear (as discussed in Chapter 4). This section describes the organization and results of two field experiments[1] , conducted in March 2022 and June 2023 (section 5.1 and 5.2 respectively). These randomized control trials aimed to assess the reception of Muse's recommendations by job seekers, as well as to answer questions about optimal communication around job recommendations[2]. Leaving a complete study of the trials' implications in terms of communication and value alignment for further work, the present chapter aims to present preliminary results relative to Muse's assessed relevance.

5.1 March 2022 field experiment

The first experiment, conducted in March 2022, set to answer several questions. First, are Muse's algorithmic recommendations relevant to job seekers? Recommender systems trained from past hires may not provide suitable job recommendation in the perspective of a PES. In particular, recommending a job too far from the job seeker's desires and profile might be considered offensive.[3] Moreover, the margins of improvement related to pursuing the development of ML tools rather than expert systems should be carefully assessed. Despite potential advantages, developing ML tools is a costly endeavor with drawbacks in terms of the traceability and interpretability of recommendations. A second concern is related to the value alignment issues discussed in Chapter 4. Are recommendations from different algorithmic sources assessed differently by job seekers in terms of fit to their search

[1]Both experiments were approved by the Institutional Review Board of the Paris School of Economics (PSE), and registered at the American Economic Association's Registry for Randomized Control Trials (https://doi.org/10.1257/rct.8998-1.3).

[2]Both interventions contained additional treatment arms providing complementary information about algorithms and job ads' rankings, which will not be discussed in the present chapter.

[3]The issue of People With Disabilities is particularly critical. Note however that the type of disability is unavailable for training or post-processing due to regulation policies.

and hiring probabilities? Could the combination of recommendations trained from hires, and proxies for job seekers' utility, improve on either of these recommendation algorithms standalone?

5.1.1 Experiment design

Algorithms The first algorithm considered is MUSE.0, which at the time of the experiment was the best MUSE version in terms of recall@k on hires. A second considered algorithm is a preference-based system (PBS) inspired from the proprietary *France Travail* expert system. PBS computes a weighted sum of criteria[4] measuring the adequacy between the job seeker's preferences and profile, and the job ad, using the same weights as the proprietary *France Travail* system [5]. PBS was used as the actual *France Travail* system was not accessible on a large scale at the time for technical reasons.

We also consider algorithms, denoted MIX hereafter, that seek to select recommendations at the Pareto frontier between the two systems, while seeking to work with ordinal quantities (ranks) rather than cardinal ones (scores). At a high level[6], MIX proceeds as follows: i) select a consideration set of job ads that are ranked highly by PBS, MUSE.0, or both; ii) among this consideration set, select the top ones according to MUSE.0; iii) reorder these by PBS score. The key hyperparameter to the approach is the share of job ads that are not discarded in step ii), determining how close the top MIX ads are to PBS and MUSE.0 original recommendations. Three versions of MIX are considered, where respectively a quarter (MIX-1/4), half (MIX-1/2), or three quarters (MIX-3/4) of the consideration set in i) is discarded at step ii). In other words, MIX-1/4 is closer to pure MUSE.0 recommendations than MIX-3/4, which is in turn closer to PBS recommendations.

The recommendation policies based on PBS and MUSE.0 significantly differ: the top-1 job ad recommended by MUSE.0 is included in the top-10 recommendations of PBS for only circa 15% of the job seekers; it does not appear among the top-100 recommendations of PBS for circa 64% of the job seekers.

Treatment groups Job seekers are randomly assigned to five treatment groups, corresponding to assignment to one of the five algorithms: PBS, MUSE.0, MIX-1/4, MIX-1/2, and MIX-3/4.

Surveyed population The eligible population are job seekers registered at *France Travail* in the Auvergne-Rhône-Alpes region, of administrative category A (*i.e.* available for a job and looking for one), aged over 18 years old, and having given the PES the permission to contact them. Randomization was stratified by desired

[4]Working hours, reservation wage, geographic mobility, type of contract; skills, diploma, languages, experience, and driver's license.

[5]More details are provided in Appendix D.1.1. Note in particular that in comparison to the criteria-based algorithm $\mathcal{U}$ studied in Chapter 4, the algorithm has a "filtering" behavior with respect to geographic distance.

[6]See Appendix D.1.1 for further details.

job type (14 modalities), the kind of accompaniment by the institution (3 modalities describing the job seeker's degree of autonomy), and geographic location (level of a French *département*, 12 modalities).

Survey protocol Job seekers are sent an email inviting them to complete a survey[7]. A link provided in the email directs them to the survey's cover page. The cover page provides them information on the survey's goals, as well as assurance that the information collected will be used for research purposes and have no impact on their treatment by *France Travail*.

If they accept those terms, job seekers are first shown two job ads (their assigned algorithm's top-2). Job ads are characterised by the firm, working conditions, wage, workplace (and distance), experience, experience, educational requirements, driver's licence requirements, and an overview of the job's and firm's textual description. Job seekers are asked to rate the two job ads (out of ten, on a continuous slider) in terms of i) global relevance, ii) their perception of their chances of being recruited, and iii) fit to their job search criteria. They may also optionally provide comments in natural language.

After rating the two job ads (which is mandatory to proceed in the survey), job seekers visualize a final page displaying ten job ads (their assigned algorithm's top-10, including the two previously seen ones). Job seekers do not have to rate ads on this page. They may click on the ads to view them on *France Travail*'s website (which provides further details on the ads, and allows job seekers to apply if they wish to). Clicks of job seekers on the ads are recorded.

Data The survey enables us to collect ratings and clicks, which we are able to merge with background information on job seekers gathered by the PES (as described in Chapter 2).

5.1.2 Analysis

In the following regression tables, three stars refer to $p < 0.001$, two to $p < 0.01$, one to $p < 0.05$, and none to a higher p-value.

Randomization, responses and attrition Completion rates on the survey are documented in Table 5.1. Altogether, 17.7% of surveyed job seekers completed the survey (*i.e.* rated two job ads and accessed the last page of the survey, regardless of whether they clicked or not on ads). The null hypothesis of equality of completion rates in the different treatment groups is not rejected in a linear regression (reported in the Appendix, Table D.1), leading us not to model differential attrition in terms of survey completion in the analysis.

Results Table 5.2 displays the results, across respondents to the survey, of the regression:

$$Y_i = \alpha + \sum_k \beta_k \{T_i = k\} + \epsilon_i$$

[7]Screenshots of the different stages of the survey are provided in Appendix D.1.2.

	N	Share (%)
Survey not started	38137	75.52
Survey started but no ads rated	2813	5.57
Survey started, only ad 1 rated	624	1.23
Survey completed, no click	6431	12.74
Survey completed, at least 1 click	2490	4.93

Table 5.1: Survey completion rates - March 2022 Field Experiment

with T_i job seeker i's received treatment, and Y_i the mean ratings of the top-2 job ads presented to job seeker i - in terms of relevance, hiring probability, and fit to job seekers' criteria. The PBS treatment serves as the reference category. Table 5.3 displays the results of the same regression with Y_i taken to be the number of clicks and the probability of clicking on at least one ad among the top-10 recommendations.

Altogether, recommendations' overall ratings do not vary much across algorithms. As expected, recommendations issued by MUSE, as well as the MIX variants closest to MUSE, are judged by job seekers to present higher hiring probabilities than those of the PBS baseline. On the other hand, the assessment of MUSE's recommendations in terms of fit to job seekers' search criteria are slightly lower than those of MUSE, albeit not in a statistically significant fashion.

	Overall rating	Hiring	Fit to job seekers' criteria
α	5.1630***	3.3569***	3.2768***
	(0.0595)	(0.0622)	(0.0628)
Muse.0	-0.0064	0.1870*	-0.0209
	(0.0837)	(0.0875)	(0.0883)
Mix-1/4	0.0412	0.2402**	0.1725
	(0.0841)	(0.0879)	(0.0887)
Mix-1/2	0.0393	0.2402**	0.1206
	(0.0827)	(0.0864)	(0.0051)
Mix-3/4	-0.0027	0.0602	-0.0162
	(0.0836)	(0.0874)	(0.8545)

Table 5.2: Ratings (top-2 ads) - March 2022 Field Experiment

Click-through rates are rather low for all treatments - between 0.4 and 0.5 clicks per person among the 10 recommendations. Algorithmic variants MUSE.0, MIX-1/4 and MIX-1/2 are associated with slightly improved probabilities of clicking at least once compared to PBS, although this effect is only significantly different from zero at the 10% threshold for MIX-1/2 (p=0.0542). All algorithmic variants except MIX-3/4 improve on PBS in terms of click numbers on the top-10 recommendations (+14% for MUSE.0, p=0.06; +18% for MIX-1/2, p=0.019). This is attributed to the limitations of PBS: while it is expected to query very relevant job ads when they exist, the quality of its recommendations might degrade when having to weight the

	Number of clicks	Clicked on at least one ad
α	0.4147***	0.2679***
	(0.0233)	(0.0108)
Muse.0	0.0611	00.0150
	(0.0328)	(0.0151)
Mix-1/4	0.0543	0.0174
	(0.0329)	(0.0152)
Mix-1/2	0.0759*	0.0288
	(0.0324)	(0.0149)
Mix-3/4	0.007	-0.0058
	(0.0327)	(0.0151)

Table 5.3: Click behavior (top-10 ads) - March 2022 Field Experiment

importance of different criteria when all of them can't be satisfied (using brittle, non-individual weights).

Table 5.4 provides results of an individual-level regression, as above, on having reported at least one of the two top job ads as having a rating of 1/10 (the worst possible score), and of having reported at least one of the two top job ads as less than 5/10. Compared to PBS, none of the assessed algorithmic variants significantly increased the rates of "poor" job ads (as so defined). Informally, when the job ads proposed by PBS or MUSE.0 are judged negatively, they get the same comments (e.g., "too far"; "I am not interested in this type of job anymore"), suggesting that the acceptability of MUSE.0 recommendations is similar to that of PBS.

	At least one ad rated 1/10	At least one ad rated lower than 5/10
α	0.1970***	0.6377***
	(0.0096)	(0.0115)
Muse.0	0.0072	0.0119
	(0.0135)	(0.0161)
Mix-1/4	0.0006	0.0175
	(0.0135)	(0.0162)
Mix-1/2	-0.0041	0.0117
	(0.0133)	(0.0159)
Mix-3/4	0.0047	0.0171
	(0.0134)	(0.0161)

Table 5.4: Prevalence of inadequate job ads - March 2022 Field Experiment

Partial conclusion The March 2022 experiment enabled us to compare MUSE.0 to the intuitive PBS expert system, as well as MIX variants mixing their two rationales. The main conclusions from the analysis were the following. First, different

rankings by plausible recommender systems were indeed perceived differently by job seekers, underlining the importance of algorithm design. Second, MUSE.0 seems at least as acceptable as PBS: its recommendations have higher click-through rates and assessed hiring probabilities, and do not generate increased rates of rejection (poor ratings) nor a significant decrease in assessed fit to job seekers' criteria. Third, the algorithms with the highest performance were found to be MIX variants, which balanced hiring probability and fit measures to search parameters to generate recommendations.

5.2 June 2023 field experiment

The second experiment, conducted in June 2023, had several goals. A first goal was to compare MUSE to *France Travail*'s current recommendation solution (SDR), and to the state of the art (XGB). A second goal was to assess whether the two-tiered structure of MUSE.2 translates to meaningful gains in job seeker satisfaction compared to standalone use of first tier MUSE.0. The third was to compare MUSE variants focused on predicting hires (MUSE.2) to approaches putting a higher weight on job seeker utility (MUSE.1.Applications and MIX), in the light of the questions about value alignment raised in Chapter 4.

5.2.1 Experimental design

Algorithms Seven algorithms are studied in the experiment:

1. MUSE.0: to i) enable a comparison to the March 2022 experiment, ii) assess if the costs of MUSE.2 in terms of algorithmic complexity and additional computation time are balanced by gains in job seeker satisfaction;

2. PBS: i) for comparison to the March 2022 experiment; ii) as it is a part of MIX-1/2;

3. SDR, based on WCC ELISE, which is *France Travail*'s current recommender system [8];

4. MUSE.2, as the best MUSE version in terms of recall computed on hires;

[8]Note that SDR leverages data from the "Offre raisonnable d'emploi" (see Chapter 2 for details) by default. This information is not necessarily as up to date or as much of a good fit to the job seekers' own wishes as additional data MUSE may leverage (*e.g.* the secondary search parameters mentioned in Chapter 2). Thus, the experimental setting can be considered to be slightly favorable to MUSE compared to SDR. However, there is no obvious way to correct for this "bias" in the evaluation setting. One way to do so would be to enable job seekers to choose which "métier recherché" can be used by the SDR in real time during the survey, which is out of the scope of what could be implemented due to pratical constraints. Moreover, note that SDR's design goals (retrieve a few, highly relevant ads corresponding to explicit search parameters) are very different from MUSE's (retrieve broader lists of recommendations based on a job seekers' profile as a whole and past hiring data). These two designs may be complementary rather than in competition in the broader scope of a PES's missions and services (each of these designs may be better suited to some application contexts).

5. MUSE.1.Applications: as defined in Chapter 3, it uses the same structure as MUSE.1 but is learnt from applications (and hires). It should place stronger emphasis on job seekers' perceived application success chances and utility than MUSE.2, possibly at the expense of actual success chances;

6. MIX-½, due to its success in the March 2022 field experiment. This version of MIX-½ is updated to leverage MUSE.2 instead of MUSE.0 (since MUSE.2 outperforms MUSE.0 in terms of recall);

7. XGB, as a representative of the state of art.

SDR-eligible sample Access to SDR during the experiment relied on an API provided by *France Travail*. A call to the API to retrieve SDR recommendations for a given job seeker could sometimes return strictly less than 10 job ads, if these were the only ones deemed relevant enough under the API's parameters (which could not be modified for the sake of the experiment). To enable comparison between populations (*i.e.* avoid comparing the effect of 5 recommendations by an algorithm to the effect of 10 recommendations from another one), SDR recommendations are only sent to the population of job seekers for whom at least 10 SDR ads may be retrieved. This population will be compared to the population in the other treatment groups *which would have received 10 recommendations from* SDR. This population will be referred to as the SDR-eligible sample in the following.

Treatment groups Job seekers are randomly assigned to seven treatment groups (each of size 10000), each corresponding to one of the seven algorithms.

Surveyed population Sampling is conducted uniformly at random among category A job seekers (*i.e.* available for a job and looking for one) in the Auvergne-Rhône-Alpes region which have given *France Travail* the permission to contact them.

Survey protocol Job seekers are sent an invitation email to complete a survey. A link in the email leads them to a landing page describing the survey's goals and terms.

After accepting to participate, job seekers proceed to view a first page showing them five job ads (their assigned algorithm's top-5), that they are asked to rate in terms of relevance on a five-point Likert scale. Rating all five ads is mandatory to proceed in the survey. At the bottom of the page, job seekers may also optionally indicate their interest in receiving more of the algorithm's recommendations.

On a second page, job seekers view the algorithm's top-10 recommendations (including the five shown on the first page), with clickable links to view the ads in more detail on *France Travail*'s website (on which they may also apply if they wish to). Clicks on the links are recorded.

An overview of the survey is provided in Appendix D.2.

5.2.2 Analysis

As above, three stars refer to $p < 0.001$, two to $p < 0.01$, one to $p < 0.05$, and none to a higher p-value.

Randomization, responses and attrition Tables 5.5 and 5.6 describe the size of populations that opened and completed the survey (*i.e.* reached the second page) per treatment group in the full sample and the SDR-eligible sample respectively. The survey completion rates range from 16.45 to 17.17 percent for the full sample, and from 14.17 to 15.45 percent in the SDR-eligible sample. Among comparable full-sample groups (*i.e.* leaving the SDR-assigned group aside), a formal test (a F-test considering the null hypothesis that all groups have the same rates as PBS) does not let us conclude in different group-wise opening and completion rate between the 6 groups. Among the SDR-eligible sample, we also cannot conclude (using a F-test with a null hypothesis of all rates being equal to SDR) in different group-wise open and completion rates among the 7 groups.

	Pbs	Muse.0	Muse.2	Muse.1.Applications	Mix-1/2	Xgb
Sent emails	9999	9999	9999	9999	10000	9999
Opened survey	2067	2000	2014	1997	2091	2015
Completed survey	1682	1653	1645	1648	1716	1648
Completion rate	16.82	16.53	16.45	16.48	17.17	16.48

Table 5.5: Response Rate by Treatment (full sample)

	Sdr	Pbs	Muse.0	Muse.2	Muse.1.Applications	Mix-1/2	Xgb
Sent emails	10000	5115	5073	5106	5121	5122	5158
Opened survey	1923	955	899	909	955	963	920
Completed survey	1539	776	719	732	791	782	764
Completion rate	15.39	15.17	14.17	14.33	15.45	15.27	14.81

Table 5.6: Response Rate by Treatment (Sdr-eligible sample)

Interest for recommendations Tables 5.7 and 5.8 display the results, on the population of job seekers who completed the survey, of a regression of measures of interest in recommendations (share of job seekers expressing interest for more recommendations, share of job seekers who found at least one ad relevant[9], number of ads found relevant, share of job seekers who clicked on at least one ad, number of clicks) on the treatment groups, in the overall population and the SDR-eligible one respectively. The reference treatment is PBS in Table 5.7, and SDR in Table 5.7.

On the full sample, MUSE.0 does not improve on PBS in a statistically significant fashion with respect to any of the considered measures of interest. The comparison of the two algorithms is altogether coherent with those of the March 2022 experiment (non-significantly higher number of clicks and similar or slightly lower average relevance for MUSE.0). XGB slightly outperforms PBS on all interest

[9]Relevance is defined here as ratings of 4 or 5 on the 5-point Likert scale.

	Interest for further rec.	$\geq$ 1 relevant ad	Number of relevant ads	$\geq$ 1 click	Number of clicks
Constant	0.5***	0.473***	1.052***	0.206***	0.322***
	(0.009)	(0.009)	(0.024)	(0.008)	(0.017)
Muse.0	0.01	0.009	-0.078	0.021	0.036
	(0.015)	(0.015)	(0.042)	(0.013)	(0.029)
Muse.2	0.069***	0.065***	0.093*	0.083***	0.173***
	(0.015)	(0.015)	(0.042)	(0.013)	(0.029)
Muse.1.Applications	0.091***	0.109***	0.251***	0.085***	0.16***
	(0.015)	(0.015)	(0.042)	(0.013)	(0.029)
Mix	0.047**	0.056***	0.061	0.054***	0.143***
	(0.015)	(0.015)	(0.041)	(0.013)	(0.029)
Xgb	0.029	0.041**	0.015	0.032*	0.073*
	(0.015)	(0.015)	(0.042)	(0.013)	(0.029)

Table 5.7: Interest for recommendations - June 2023 experiment, full sample

measures (although the difference is non-significant when it comes to interest for further recommendations and the number of ads found relevant). MIX also performs better than PBS for all outcome measures (in a statistically significant fashion, except for the number of ads found relevant). MUSE.1.Applications and MUSE have the highest means for all interest measures (with MUSE.1.Applications above MUSE for all measures aside from the number of clicks). For both algorithms, the difference with respect to PBS is statistically significant, and sizeable in magnitude (*e.g.* improvements of 18% and 14% in the share of respondents interested in further recommendations; of 53% and 50% in terms of number of clicks).

Results on the SDR-eligible population lead to similar conclusions. SDR outperforms PBS on all outcomes (differences are statistically significant for expressed interest and relevance), as well as MUSE.0 (the difference is significant for expressed interest). It has similar performances to MIX (except on likes at the extensive margin, for which MIX has significantly higher mean) and XGB. MUSE.2 and MUSE.1.Applications outperform SDR on all outcomes, with the differences being significant (except for MUSE.2 in terms of the number of relevant ads), and quantitatively sizeable. For instance, MUSE.1.Applications recommendations (resp. MUSE.2) lead to 13% more expressions of interest (resp. 10%), and to 30% (resp. 45%) more clicks.

	Interest for further rec.	$\geq$ 1 relevant ad	Number of relevant ads	$\geq$ 1 clicks	Number of clicks
Constant	0.564***	0.543***	1.332***	0.227***	0.374***
	(0.013)	(0.013)	(0.038)	(0.011)	(0.025)
Pbs	-0.059**	-0.045*	-0.288***	-0.014	-0.054
	(0.022)	(0.022)	(0.065)	(0.019)	(0.043)
Muse.0	-0.045*	0.01	-0.114	0.005	0.019
	(0.022)	(0.022)	(0.067)	(0.02)	(0.045)
Muse.2	0.059**	0.072**	0.031	0.077***	0.17***
	(0.022)	(0.022)	(0.066)	(0.019)	(0.044)
Muse.1.Applications	0.076***	0.12***	0.222***	0.075***	0.114**
	(0.022)	(0.022)	(0.065)	(0.019)	(0.043)
Mix	0.022	0.075***	0.066	0.036	0.077
	(0.022)	(0.022)	(0.065)	(0.019)	(0.043)
Xgb	0.012	0.032	-0.056	0.025	0.066
	(0.022)	(0.022)	(0.065)	(0.019)	(0.044)

Table 5.8: Interest for recommendations - June 2023 experiment, Sdr-eligible sample

Partial conclusion The June 2023 march experiment enabled us to compare MUSE variants to the institution's expert system and to the state of the art. The gains associated to the use of the proposed two-tier MUSE architecture, rather than simply relying on the first-tier MUSE.0 rankings, was vindicated. MUSE variants

MUSE.2 and MUSE.1.Applications, learned from hirings and applications respectively, were shown to outperform the institution's system (SDR) as well as XGB on a variety of indicators of interest. While MUSE.1.Applications slightly outperformed MUSE.2 on most indicators (albeit not necessarily in a statistically significant fashion), judging of the relative quality of both algorithms in terms of usefulness to job seekers requires more evidence. Indeed, it is unsurprising that an algorithm mimicking applications is more appreciated by job seekers; yet MUSE.2 recommendations may yet, perhaps, lead to more successful matches down the line.

The main perspective is the investigation of the effect of recommendations on downstream labor market behavior (applications, return to employment and employment quality), which requires larger sample sizes and continuous exposure of job seekers to recommendations. It shall be pursued in a large-scale experiment providing a stream of exposition to recommendations to job seekers on the long term, with an experiment design accounting for possible externalities. Other perspectives for further work include the study of possibly heterogeneous treatment effects of MUSE.1.Applications and MUSE with respect to SDR, in order to understand which job seekers may benefit from machine-learned recommendations and why; and improving the design of MUSE itself by leveraging the feedback collected from job seekers.

Chapter 6

Congestion-avoiding job recommendation

As noted in Chapter 1, job ads are *rival goods*: while recommender systems typically aim at independently recommending to each user the most desirable item *for them*, it is inappropriate to recommend the same irresistible job ad to many job seekers, as this would induce a congestion phenomenon at the population level and a poor eventual satisfaction at the individual level. More generally, in domains such as the labor market or online dating, referred to as *reciprocal recommendation settings* [Pal+21], an appropriate recommendation policy should globally take into account the populations of job seekers and job ads, and somehow connect both populations in a congestion-free way.

Taking inspiration from related works in recommender systems [Li+19; LBZ19; CHL19] and in econometrics [CS16; Gal18], this chapter investigates the coupling of optimal transport [Cut13; PC19] with recommender systems. The presented approach, referred to as *Congestion avoiding recommendation with Optimal Transport* (CAROT), learns matchings between the users (job seekers) and the items (job ads) populations, aimed to maximize some trade-off between the interestingness of the recommended items, and their sufficient diversity at the population level (as opposed to recommendation serendipity [KP17], aimed at the recommendation diversity at the individual level). The scientific questions considered thus regard: i) how to measure and algorithmically prevent congestion; ii) how to assess the trade-off between the mainstream recommendation performance indicator, that is, recall, and congestion.

Our contributions are threefold. Firstly, congestion avoidance is formalized within the optimal transport framework (Section 6.2). Secondly, the CAROT algorithm proposed to tackle this problem is agnostic regarding the data distribution (as opposed to the assumptions in [Li+19; LBZ19; Gal18; CS16]), and is less computationally demanding than *e.g.* combinatorial optimization approaches [Xia+19]. Thirdly, the merits of CAROT are empirically demonstrated on public data on marriages (for comparative evaluation with [Li+19]), and on the *France Travail* data. The experimental results demonstrate the robustness of the approach *w.r.t.* the state of the art, and yield some unexpected lessons about the interactions of the recall and congestion indicators.

6.1 Related work

This section reviews other congestion-related approaches to recommendation, and introduces the so-called optimal transport setting for the sake of self-containedness (see [PC19] for a thorough survey).

Notations. As previously, let n (respectively m) denote the number of users (resp. items), with x_i (resp. y_j) the description of the i-th user (resp. j-th item). The boolean collaborative matrix $M_{i,j}$ is such that $M_{i,j} = 1$ if and only if user i selected item j.

Position of the problem. A recommender system usually learns a scoring function s such that the matrix defined from $s_{i,j} = s(x_i, y_j)$ maximizes the fit with the collaborative matrix M (expressed in terms of mean-square error or Kullback-Leibler divergence), possibly penalized with a regularization term [Agg16]. For convenience of notations, and with no loss of generality, it is assumed in the remainder of this chapter that the items recommended to the i-th user are ordered by *increasing* $s_{i,j}$.

In a rival good setting, item j is subject to capacity constraint n_j: only the top n_j users selecting this item can be served. New optimization objectives and algorithms need be defined to accommodate such constraints.

Related works. An early approach facing reciprocal recommendation, [GMZ13] assign recommendations as the solution of a constrained optimization problem. [Xia+19] casts reciprocal recommendation as a (NP-hard) multi-objective optimization problem, where the additional objective accounts for satisfying the capacity constraints; it is tackled using greedy optimization.

In [BZK17], a Poisson model is used in real time to forecast the expected number of clicks on an ad. The ad's visibility is shifted up or down accordingly. The downside of this lightweight intervention, however, is that it requires real-time access to modify the ads' visibility.

Our closest neighbors are [CHL19]: inspired from decentralized economic models, they consider the scoring functions reflecting the mutual utility of x_i w.r.t. y_j, and use an optimal transport approach (see below) to alleviate congestion.

Since the presentation of this Chapter's key results in a 2021 workshop [Bie+21], further approaches leveraging tools from optimal transport have been proposed. In economics, the ideas presented [CHL19] were improved (by considering global assignment plans rather than local, individual-level approximations in a "local market"), tested in the field and published in in [CHL23]. [Tom+23] proposed to leverage optimal transport in a recommender system setting with an overall approach close to ours. [Mas+23; Mas+24] propose an in-processing approach to congestion-avoiding recommendation (using an optimal transport term in the within-batch training loss of a neural recommender system), and have favorably compared their approach to the one presented here.

Computational optimal transport. Optimal transport (OT) aims to map some (continuous or discrete) distribution μ onto another distribution ν. In the following, μ (respectively ν) stands for the uniform discrete distribution on the set of n users (resp. on the set of m items). Denoting $\Gamma(\mu, \nu)$ the set of measures such that their marginals with respect to first and second arguments respectively are μ and ν, letting $C_{i,j}$ be the cost of mapping i onto j, the OT problem aims to find a joint distribution γ^* in $\Gamma(\mu, \nu)$ s.t.:

$$\gamma^*(C) = \arg \min_{\gamma \in \Gamma(\mu,\nu)} \sum_{i=1}^{n} \sum_{j=1}^{m} \gamma_{i,j} C_{i,j} \tag{6.1}$$

A tractable relaxation of the above optimization problem is proposed by [Cut13] by regularization with an entropic term:

$$\gamma^*(C, \varepsilon) = \arg \min_{\gamma \in \Gamma(\mu,\nu)} \sum_{i=1}^{n} \sum_{j=1}^{m} \gamma_{i,j} C_{i,j} + \varepsilon \sum_{i=1}^{n} \sum_{j=1}^{m} \gamma_{i,j} (\log(\gamma_{i,j}) - 1) \tag{6.2}$$

with ε the regularization weight. When $\epsilon \to 0$, the solution converges to a solution of Problem 6.1 with maximum entropy; whereas when $\epsilon \to \infty$, the solution tends to $\mu \otimes \gamma$, *i.e.* a uniform coupling that does not take cost C into account. Crucially, the solution to optimization problem 6.2 is unique and takes the form:

$$\forall (i, j) \in [[n]] \times [[m]], \quad \gamma_{ij} = u_i K_{ij} v_j$$

for two scaling variables $(u, v) \in \mathbb{R}_+^n \times \mathbb{R}_+^m$, and $K_{ij} = \exp(-C_{ij}/\epsilon)$, meaning that it can be parametrized by $n + m$ variables. Moreover, u and v satisfy:

$$\text{diag}(u) K \text{diag}(v) 1_m = \mu, \qquad \text{diag}(v) K^T \text{diag}(u) 1_n = \nu$$

To find u and v satisfying this equation (the so-called matrix scaling problem), one can resort to Sinkhorn's algorithm, which proceeds by initializing $v^{(0)}$ by an arbitrary positive vector (say 1_m), and iterating:

$$u^{(l+1)} = \frac{\mu}{K v^{(l)}}$$

$$v^{(l+1)} = \frac{\nu}{K^T u^{(l+1)}}$$

We refer the reader to [PC19], Chapter 4, for proofs of the above statements, and a thorough introduction to entropic-regularized optimal transport (including convergence analysis and computational tricks).

Discussion A distinct line of work, for instance in econometrics [GS22; CS16], but also in recommendation ([Li+19], which is the main baseline used for comparison in this chapter) assume the observed collaborative matrix M to be the optimal solution of an OT plan based on some matching cost C [CS16; GS22; Li+19; LBZ19]. They infer C from training data, and use the estimated cost model to build matchings on other data.

In econometrics, [DG14; CS16; GS22] formalize matching markets (*e.g.* dating) as the result of an optimal transport process and infer the individual matching utilities $C_{i,j}$ from the observed matching. Along the same lines, [Li+19] infer the cost $C_{i,j}$ such that the optimal solution of the OT problem (Eq. (6.2)) is as close as possible to M in the sense of the KL divergence. Another approach, motivated by applications in bioinformatics, is proposed by [LBZ19]. The observed matchings among sets of RNA sequences, referred to as clusters, is exploited to infer the cost of transport among these clusters and derive new matchings.

In the context of the job market, it is however debatable whether the actual matching, *i.e.* the observed collaborative matrix M, should be viewed as the solution of an optimal transport plan. This is only the case under strong structural assumptions, *e.g.* the transferable utility assumptions of [CS06]. We remain agnostic on the data-generating process, and instead view optimal transport as a post-processing tool. Accordingly, the proposed approach will be structured along two phases: learning the matching cost function C from M (without assuming M to be an OT solution), and using C within an OT process.

6.2 Overview of CAROT

Let s_{ij} in $\mathbb{R}$ denote the sought recommendation score of the j-item for the i-th user.

Performance criteria. Beside the standard recall@k indicator, measuring the fraction of users for which the actually preferred item is ranked among the top-k recommendations, we define the notion of *item market share* $MS_k(j)$ of item j as the fraction of users i such that j is among the top k items recommended to i (with $k < m$). Informally, the congestion is minimized if minus the entropy of the market shares is minimized (all the more so as k goes to 1):

$$\text{Congestion@}k(s) := \sum_{j=1}^{m} MS_k(j) \log\left(MS_k(j)\right) \tag{6.3}$$

For the sake of normalization, the congestion indicator is mapped to $[-1, 0]$. Perfect congestion avoidance is obtained for equal market shares of the items, with -1 as optimal value.

Enforcing congestion avoidance with OT. As said, optimal transport is applied based on a matrix of pairwise costs $C_{i,j}$, denoting how costly it is to match i and j. The key algorithmic design question then becomes the definition of the cost function. Leaving end-to-end learning of $C_{i,j}$ for further work, we focus on choices of the form $C_{i,j} = g(s_{ij})$ or $g(r_{ij})$, where r_{ij} is the rank of j for i in terms of s_{ij}. g is a monotonous scalar function, hyper-parameter of the approach, such that the cost $C_{i,j}$ of transporting i toward j increases with the recommendation score s_{ij} (that is, as the relevance of matching i and j decreases), or with the rank r_{ij}.

A first key consideration is whether to use ranks or scores. Depending on the quality of the underlying scoring function s_{ij} and what it represents (if it is a proxy

of hiring probabilities for instance), directly using values of s_{ij} in the cost enables between-person comparisons in terms of match quality and better accounting for the distribution of the quality of recommendations. Using ranks r_{ij} on the other hand puts all job seekers on a formal foot of equality and is agnostic about the interpretation of s_{ij}'s (only within-individual rankings by s_{ij}'s matters, which is reassuring when s_{ij} is trained with a ranking loss), but may result in a loss of information. A second key consideration is whether g is to be chosen concave, convex or linear: depending on the choice of concavity or convexity, keeping a strong weight in γ on the very top recommendations compared to the very next ones becomes more or less crucial.

In the following, s_{ij} is capped to the score of the 1,000-th item recommended to each i, noted $s_i^{(1000)}$ ($s_{ij} \leftarrow \max(s_{i,j}, s_i^{(1000)})$ in the following). Four g functions have been considered, respectively linear, or exponential functions of s_{ij}, or rank-based, or NDCG-like.

$$C_{ij} = \min(\frac{s_{ij} - s_{min}(i)}{s_{i,1000} - s_{min}(i)} \times 100, 100) \qquad \text{(Linear / Id+)}$$

$$C_{ij} = \min\left(\exp(\log(10) \times \frac{s_{ij} - s_{min}(i)}{s_{i,1000} - s_{min}(i)}), 10\right) - 1 \qquad \text{(Exponential)}$$

$$C_{ij} = \begin{cases} 1 & r_{ij} = 1 \\ 2 & r_{ij} \in]1, 2] \\ 3 & r_{ij} \in]2, 10] \\ 4 & r_{ij} \in]10, 100] \\ 5 & r_{ij} \in]100, 1000] \\ 6 & r_{ij} > 1000 \end{cases} \qquad \text{(Rank-based)}$$

$$C_{ij} = 1 - \frac{\log(2)}{\log(1 + r_{ij})} \qquad \text{(NDCG)}$$

To enable the comparison of results obtained with same entropic regularization weight ε but different cost definitions, $C_{i,j}$'s are normalized so that $\sum_{i,j} C_{i,j} = 1$.

The Carot algorithm. Overall, Carot is a 2-step process: i) learning a scoring function s; ii) solving the optimal transport problem defined from $C_{ij} = g(s_{ij})$.

Carot: 1. Learning s. The two considered learning approaches for s are Xgb and neural networks (NN). Xgb is a recommender system based on gradient boosting [CG16; VYP17a], that can be efficiently trained by aggressively subsampling the negative pairs (i, j), at the expense of a lesser scalability in recommendation. NN corresponds to an earlier version of Muse.0; it is a neural net trained with a triplet loss whose architecture is tailored to the specifics of the domain (e.g., considering submodules devoted to geographic or skill-related information)[1]. As for Xgb, negative sampling is used to cope with the number of negative pairs. More details on the NN architecture, and on the hyper-parameters of Xgb and Muse.0 are provided in Appendix E.

[1]Details on the approach are provided in Appendix E.

CAROT: 2. Optimal transport. Depending on the regularization weight ε and the g function (with $C_{ij} = g(s_{ij})$), discrete distribution γ is trained by optimizing Eq. (6.2) using Sinkhorn's algorithm. Note that the extension of the approach to the general reciprocal recommendation case (*e.g.* where several positions are opened for the j-th job ad) is straightforward by making ν_j proportional to the capacity constraint of item j.

Eventually, the CAROT recommendation proceeds deterministically, ordering the j items recommended to user i in decreasing order w.r.t. $\gamma_{i,j}$.

6.3 Results

This section presents the empirical validation of the approach, conducted on two datasets: public mariage data (noted MAR), first introduced by [Li+19], for the sake of comparison with the state of art; and a proprietary dataset provided by *France Travail*.

The first goal of experiments is to assess the efficiency of the proposed approach in terms of trade-off between recall and congestion. The second goal is to investigate how the results depend on the hyper-parameters of the approach: s being learned using XGB or MUSE.0; entropic regularization weight ε ranging in $10^{-2}, \ldots, 10^{2}$; transport cost C_{ij} defined as $g(s_{ij})$ with g ranging in {Id, Exp, Ndcg, Rank}.

With each hyper-parameter setting is associated seven performance indicators: recall@k with $k = 1,\ 10,\ 100$, congestion@k with $k = 1,\ 10$, and coverage@k with $k = 1,\ 10$, indicating the fraction of items involved in top-k recommendation of at least one user. Additional results are reported in Appendix E.

6.3.1 MAR Dataset

Dataset description. The data include 2,475 men (respectively women), partitioned in 50 clusters. Each individual is described with 11 mostly ordinal features. The 1-to-1 matching is described at the individual level and the data also include the $M_{c,c'}$ collaborative matrix, reporting the fraction of matches between men from cluster c and women from cluster c'.

Table 6.1: Comparative results on MAR at the cluster level; average and standard deviation of the RMSE and MAE w.r.t. the cluster matrix M, over 5-fold CV. Results for CAROT correspond to $g = Id+, \varepsilon = 1$.

	Random	PMF	SVD	itemKNN	RIOT	γ^{NN}	γ^{XGB}
RMSE	10.71± 0.13	446.6± 9.86	441.4± 11.2	9.36± 0.12	9.12± 0.12	8.98± 0.17	8.89± 0.11
MAE	7.22± 0.06	251.3± 6.00	249.2± 5.71	6.30± 0.03	5.98± 0.10	5.80± 0.13	5.79± 0.12

Benchmarks. The baseline results on MAR are those of RIOT [Li+19], an SVD-based decomposition, and itemKNN. At the cluster level, the performance indicators include the RMSE (Root Mean Squared Error) and the MAE (Mean Absolute Error) between the collaborative matrix M at the cluster level and the estimated

Table 6.2: Comparative Results on MAR at the individual level: Recall, Coverage and Congestion.

	Algorithm	Recall (%) @1	Recall (%) @10	Coverage (%) @1	Coverage (%) @10	Congestion @1	Congestion @10
	s Random	0.16	2.27	63.32	100	-0.90	-0.98
CAROT-XGB	s XGBoost	7.93	27.88	48.55	98.69	-0.84	-0.94
	γ^{XGB},$g = Id+$,$\varepsilon = 1.0$	8.05	28.41	49.77	99.18	-0.85	-0.95
	γ^{XGB},$g = Id+$,$\varepsilon = 0.1$	8.01	27.02	72.73	100	-0.93	-0.95
	γ^{XGB},$g = Id+$,$\varepsilon = 0.01$	6.47	23.77	96.05	100	-0.98	-0.84
	γ^{XGB},$g = ndcg$,$\varepsilon = 1.0$	7.93	28.2	48.55	99.02	-0.84	-0.95
	γ^{XGB},$g = ndcg$,$\varepsilon = 0.1$	8.10	25.72	59.42	100	-0.89	-0.93
	γ^{XGB},$g = ndcg$,$\varepsilon = 0.01$	6.06	19.49	94.26	100	-0.98	-0.73
CAROT-NN	s NN	3.82	15.50	46.27	98.00	-0.83	-0.93
	γ^{NN},$g = Id+$,$\varepsilon = 1.0$	2.84	14.32	38.86	92.47	-0.80	-0.90
	γ^{NN},$g = Id+$,$\varepsilon = 0.1$	3.94	15.46	70.12	100	-0.92	-0.98
	γ^{NN},$g = Id+$,$\varepsilon = 0.01$	3.78	15.46	93.48	100	-0.98	-0.95
	γ^{NN},$g = ndcg$,$\varepsilon = 1.0$	3.82	15.63	46.27	98.73	-0.83	-0.94
	γ^{NN},$g = ndcg$,$\varepsilon = 0.1$	4.23	13.87	57.99	99.91	-0.88	-0.93
	γ^{NN},$g = ndcg$,$\varepsilon = 0.01$	2.89	11.60	93.44	100	-0.98	-0.72

recommendation matrix, measured using a 5-cross fold validation. CAROT is also assessed at the individual level, using the performance indicators defined above.

Results. On the marriage benchmark, tables 6.1 and 6.2 respectively display the comparative results obtained at the cluster[2] and the individual level.

At the cluster level, γ^{XGB} slightly but statistically significantly improves on RIOT w.r.t. both RMSE (8.89± 0.11 as compared to 8.98± 0.17) and MAE (5.80± 0.13 in contrast to 5.79± 0.12). γ^{NN} also slightly improves on RIOT. Other benchmarks (random, PMF, SVD and itemKNN) are outperformed.

At the individual level, XGB significantly outperforms NN in both terms of recall and congestion for all values of k.

γ^{XGB} is found to only improve the congestion at the expense of the recall: improving the congestion (from -.84 to -.98) is obtained by decreasing the recall@10 (from 28.4% to 23.7% at best, for $g = Id, \varepsilon = 10^{-2}$).

For γ^{NN}, the congestion can be significantly improved (from -.84 to -.98, for $g = Id, \varepsilon = 10^{-2}$) while preserving the recall@10 (circa 15.4%); however, the initial recall is significantly lower than for XGB, as said above.

The recall and the congestion indicators may not actually be antagonistic in the MAR problem: by construction, the sought collaborative matrix is a permutation. The main difficulty for this recommendation problem thus seemingly comes from

[2]The difference with [Li+19] is explained as a bug was found (and corrected) in the publicly available code for RIOT and other baselines, dividing the error by the number of folds in each iteration. The performance order is not modified by (correcting) the bug.

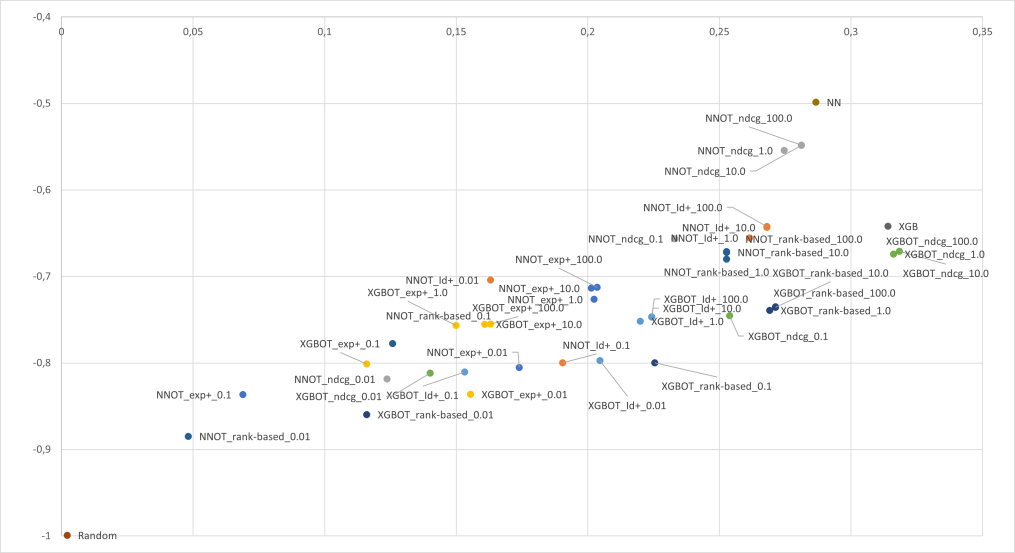
the small size of the dataset and the poor description of the individuals.

6.3.2 JOB Dataset

Dataset description. The training dataset includes circa 1,650,000 job seekers, 477,000 job ads and 43,000 matches (signed contracts) reported in *Ile-de-France* during the Feb.-Oct 2018 period. The description x_i (respectively y_j) of a job seeker (resp. job ad) is in $\mathbb{R}^{448}$ (resp $\mathbb{R}^{582}$). Function s is learned on the training set. The optimal transport plan γ is computed on the test set, restricted to the job sector of logistics for scalability reasons, including 110,000 job seekers, 14,200 job ads and 450 matches in Ile-de-France in November 2018.

Results Figure 6.1 displays the different methods in the 2D recall@10, congestion@10 plane, illustrating the trade-off between both indicators, and shows the Pareto front of the non-dominated approaches. Table 6.3 summarizes the results of selected methods (full results are provided in Appendix E). Figure 6.2 also displays so-called Lorenz curves plotting the cumulative percentage of occurrences in recommendations among job ads sorted by number of occurrences (akin a Gini index) for selected methods.

Figure 6.1: Pareto front Congestion (-Congestion@10) - Recommendation accuracy (Recall@10) tradeoff, *France Travail* Dataset



Firstly, NN is dominated by XGB in terms of all three performance indicators: recall, coverage and congestion. Compared to XGB, the lesser recall of NN (4% loss in recall@100) comes with a much lower coverage (7% loss in coverage@1). This counter-performance is blamed on the architecture of the neural net (as reported in Chapter 3, it has since been improved). Congestion is found to be a potentially substantial issue. Despite job seekers outnumbering job ads by a factor of roughly 8, and the restriction of job seekers and job ads to a common occupation, only 12.94% of job ads appear in any of XGB's top-ones, and only a quarter of them in any of its top-tens.

Table 6.3: Comparative Results on *France Travail*: Recall, Coverage and Congestion.

	Algorithm	Recall (%)			Coverage (%)		Congestion	
		@1	@10	@100	@1	@10	@1	@10
	s Random	0	0.21	0.65	99.95	100	-0.99	-0.99
CAROT-XGB	s XGB	9.62	31.40	61.59	12.94	25.16	-0.62	-0.64
	γ^{XGB},$g = Id+$,$\varepsilon = 1.0$	4.81	21.99	57.87	21.61	31.76	-0.74	-0.75
	γ^{XGB},$g = Id+$,$\varepsilon = 0.1$	2.18	15.31	56.01	27.54	41.24	-0.78	-0.81
	γ^{XGB},$g = Id+$,$\varepsilon = 0.01$	4.37	20.45	43.21	46.75	57.61	-0.85	-0.79
	γ^{XGB},$g = ndcg$,$\varepsilon = 1.0$	9.62	31.61	62.36	12.96	26.14	-0.62	-0.67
	γ^{XGB},$g = ndcg$,$\varepsilon = 0.1$	8.97	25.38	46.06	14.69	30.84	-0.67	-0.74
	γ^{XGB},$g = ndcg$,$\varepsilon = 0.01$	5.03	14.00	18.81	36.81	57.52	-0.82	-0.81
CAROT-NN	s NN	5.68	28.66	57.98	6.02	17.78	-0.46	-0.49
	γ^{NN},$g = Id+$,$\varepsilon = 1.0$	6.78	26.14	60.39	11.99	26.30	-0.62	-0.65
	γ^{NN},$g = Id+$,$\varepsilon = 0.1$	2.40	19.03	50.43	28.23	40.16	-0.80	-0.79
	γ^{NN},$g = Id+$,$\varepsilon = 0.01$	3.93	16.30	27.89	53.38	62.35	-0.83	-0.70
	γ^{NN},$g = ndcg$,$\varepsilon = 1.0$	5.68	27.46	59.08	6.02	19.75	-0.46	-0.55
	γ^{NN},$g = ndcg$,$\varepsilon = 0.1$	5.25	23.3	49.01	8.85	26.40	-0.53	-0.65
	γ^{NN},$g = ndcg$,$\varepsilon = 0.01$	1.53	12.36	24.28	35.41	51.56	-0.81	-0.81

Secondly, coverage (@1 and @10) increases, and recall (@1, @10, @100) decreases as ε decreases from 1 to .01, leaving little hope that one can combine a good coverage with a decent recall. More encouraging is the fact that the congestion@1 can be significantly improved (from -.62 to .78 and -0.85) at the expense of a moderate recall loss (recall@100 goes from 62% to 56% and 43%) for $g = Id+, \varepsilon = .1$ and $\varepsilon = .01$.

The option $g = ndcg$ has little (slightly positive for both recall and congestion) effects for $\varepsilon = 1$ and strongly detrimental effects (for recall) for $\varepsilon = .1$ or .01.

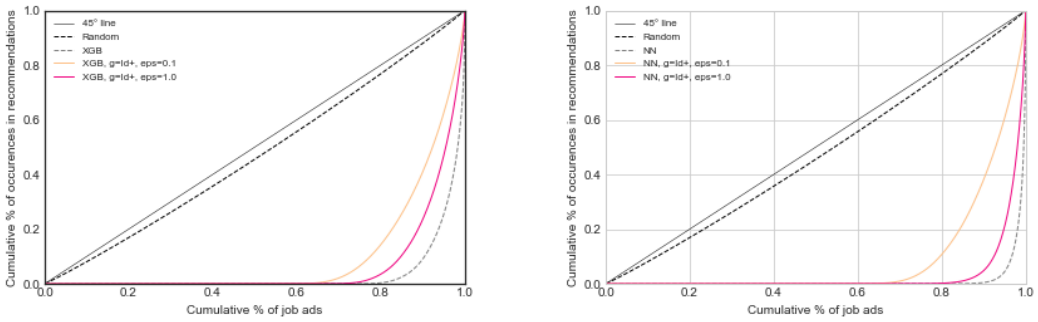
Somewhat surprisingly, decreasing ε yields a better (lower) congestion at the expense of a worse recall. This was not obvious: the higher ε, the more uniform the transport plan γ (everything else being equal). Yet, performance indicators depend on the order induced by γ, as opposed to the actual $\gamma_{i,j}$ values. Complementary experiments reported in Appendix E illustrate the interaction of ε and actual market shares.

6.4 Partial conclusion

This chapter aimed at documenting, and investigating algorithmic ways to prevent, undesirable side effects of recommender systems in terms of congestion.

On the *France Travail* dataset, congestion was found to be a potentially important issue - at least based on offline measures. Despite job seekers outnumbering job ads by a factor of eight in the experiments, three ads in four never appear in any job seekers' top-10 recommendation list (in the case of XGB, the least congested

Figure 6.2: Lorenz curves computed on Top10 recommendations



(a) XGB on *France Travail* dataset

(b) NN on *France Travail* dataset

standard recommender system studied).

To address congestion issues, we proposed CAROT, an algorithmic approach taking inspiration from computational optimal transport, with the idea of globally "transporting" the job seekers population onto the job ads population, enforcing a decent recall with low congestion. The key question becomes the definition of the transport cost, which in this chapter is based on a mainstream recommender score or ranks. The surprising lessons learned from the application of the approach on the *France Travail* dataset is that the transport cost and the entropic regularization (used to enforce OT's scalability [Cut13]) interact in subtle ways. Chiefly, a strong regularization significantly degrades the recall while it does not improve the congestion to the desired extent.

This work opens a series of perspectives, both from the point of view of computer science and economics.

Various perspectives exist to improve on CAROT. As a first step, learning the cost-defining function g to balance recall and congestion directly, or adopting an in-processing approach as in [Mas+23; Mas+24], should improve the balance between recall and congestion. Second, even given a satisfactory transport cost, the OT formalization is not completely adequate to the problem of interest despite its algorithmic convenience. The results of an OT plan γ can either be used in deterministic mode (sorting γ_{ij}'s) or probabilistic mode (drawing indices without replacement proportionally to $\{\gamma_{ij}\}_{j=1,\dots m}$). The probabilistic mode comes with guarantees in expectation if k=1, but degraded recall excessively in complementary experiments[3]. The deterministic mode is brittle, achieves imperfect congestion-avoidance (in the CAROT experiments, coverage@10 remains far 100% on the *France Travail* dataset despite the sample imbalance), and lacks mathematical guarantees. The choice of ε in deterministic mode only helps navigate the congestion-recall trade-off to an extent, without formal guarantees. Moreover, the OT formulation (forcing γ to respect pre-specified marginals μ and ν) implicitly assumes that the number of times each job ad should be shown can be pre-defined in advance. A more convincing ap-

[3]Unless ε is taken extremely low, which creates other numerical and convergence speed issues with Sinkhorn's algorithm.

proach would simply put a bound B on the number of times a job ad could at most be recommended, and leverage B to trade off congestion and recall. Given a cost matrix, the problem of job recommendation under such capacity constraints naturally takes the form of an integer linear programming problem (ILP). Tackling this ILP using a mix of linear programming relaxations and randomized rounding may yield formal and empirical improvements (see [Man+13] for a relevant large-scale application of such methods).

Congestion, and the effects of congestion-avoidance policies, should also be better understood from an economic perspective. First, to what extent does congestion already exist in the training data? What kinds of ads experience the most competition, and why? Conversely, why are some ads unpopular (shortage of applicants with specific skills, poor working conditions or wages, poor writing of the job ad)? Second, our congestion measures and algorithmic approach implicitly assumed that recommendations were followed upon in terms of applications by job seekers (at least in a uniform fashion). This is unlikely to be the case in practice (given, for instance, the reply and click rates documented in Chapter 5), and should be accounted for to confirm congestion is a primary problem for job recommender systems in practice. The effects of recommendations on congestion, and overall welfare effects, could be better understood using specific randomized control trials designs with levels of exposure varying by micro-markets [Cré+13] instead of relying on offline measures. Moreover, the reactions of job seekers to recommendations (especially ones far from their top ones) should be better accounted in algorithmic approaches to congestion - for instance by defining the cost in terms of economic fundamentals (rather than an ad hoc cost function).

Redirecting job search from a “central planner" point of view as in Carot also poses ethical issues. Understanding which job seekers bear the cost of congestion-avoidance is necessary before any application of such methods in the field (in Carot, this population may strongly vary on the definition of the cost function - especially, on whether it depends on scores or on ranks). This inquiry should be conducted on both sides on the market, as emphasized by [LTL23].

Instead of deciding to redirect their search efforts in their stead, giving job seekers information about (expected or actual) competition [BFH21; Gee19] in order to let them decide whether to apply (potentially showing a variety of more-or-less expected-to-be-competitive job ads), or more light-weight approaches [BZK17] might be more effective and respectful of job seekers' well being and decision ability in practice (although this could require fine-grained access to application counts, ads display and recommendations in real time). As noted by [Alt+22], another possible policy is simply targeting populations most likely to benefit from recommender system advice, while keeping overall program scale limited (although this also poses a range of ethical issues).

Chapter 7

Fairness in job recommendations: estimating, explaining, and reducing gender gaps

While job recommender systems have the potential to help reduce frictional unemployment, they are also a textbook case of fairness issues in machine learning [BHN23]. Algorithms trained on real-world data, which involve human biases and discriminatory practices, may reproduce past undesirable behavior such as gender stereotypes, and widen labor market inequalities. Ensuring this does not happen is a major concern for the scientific community, Public Employment Services as well as for all citizens.

This chapter investigates the issue of gender fairness [1] within the context of the audit of the MUSE recommender system (more precisely, of MUSE.1, learned solely from hires)[2].

While we document gender differences in recall, our discussion focuses on gender disparities in recommendations viewed in terms of job characteristics (*e.g.*, occupation, distance, wage, full or part-time status). Gender gaps may arise from different application behavior between men and women, which may reflect different preferences or distinct *perceived* hiring probabilities between job seekers. Some fairness definitions justify algorithms' replication of job seekers' preferences since it should maximise users' welfare – see the related individual, envy-freeness, and preference-based notions of fairness [Dwo+12; Var74; Zaf+17]. However, these gaps can also arise from differential valuations of inherent job seeker's characteristics by recruiters based on gender, which may be deemed discriminatory. Other fairness definitions such as (conditional) recommendation independence require (conditional) independence of recommendations from gender, regardless of the gaps' origins.

Our contributions are threefold. First, we discuss the origins of gender disparities and their links with common fairness definitions. Second, we propose measures of gender gaps conditional on job seekers' qualifications and preferences, using dou-

[1] Despite the limitations of representing gender as a binary construct, our analysis treats gender as binary, as it is provided in *France Travail*'s data sources in a binary fashion.

[2] Recall that MUSE does *not* use gender as a feature directly; it may nevertheless be indirectly learnt from other features during the learning process.

bly robust estimators [RRZ94; Che+17]. We document gender disparities in job ad recommendations both unconditionally and conditionally on qualifications and preferences and find that these standalone do not fully account for the observed gender gaps. We also show that the system does not exacerbate existing gaps in observed hiring or application decisions. This discussion brings forth a tension between a PES's missions and values: providing optimal personalized recommendations regarding access to employment while ensuring fair treatment between women and men. Our third contribution is to propose a scalable post-processing approach to mitigate gender gaps and to investigate the trade-off it entails. The proposed strategy is based on a Lagrangian relaxation of an integer linear programming problem, maximizing recommendation quality under unconditional or conditional gender gaps (in terms of recommendation characteristics). Empirically, we find that this approach reduces gender gaps in recommendations, at the cost of a loss of performance (as measured by recall).

The rest of this chapter is structured as follows. Section 7.1 reviews related work. Section 7.2 discusses how differential treatment may arise in recommendations, discusses its links to fairness definitions, and proposes to leverage doubly robust estimators to make inference on the effect of gender on the recommendations, while controlling for the channel of qualifications and preferences. Section 7.3 presents the experimental setting. Section 7.4 audits the algorithm in terms of recommendation performance, provides evidence of differentiated treatment, and compares these differences to those found in hiring and application behavior. Section 7.5 introduces a post-processing approach aiming at reducing (conditional) gender gaps and documents its impact on performance metrics as well as on gender gaps. Section 7.6 concludes and provides perspectives for further work.

7.1 Related work

Overview Fairness in the context of recommender systems draws an increasing amount of work, surveyed by [Eks+22; Wan+22; Li+22], and, specifically in recruitment applications, by [Kum+23]. Depending on the application domain, fairness issues may arise w.r.t. items (sharing users' attention among items in an equitable way), w.r.t. users (presenting a fair selection of items to the users), or both [SJ18; SJ19; GAK19; Do+21]. In the present work, we focus on user fairness.

Equity of utility Some approaches to user fairness question whether recommendations are equally useful to different subgroups of users - so-called *equity of utility* according to [EP22]. In practice, this may entail questioning whether standard metrics measuring the quality of recommendations are similar between different groups of users. In a collaborative filtering context, [YH17] proposes metrics to measure discrepancies between a recommender system's prediction behavior between groups. [Meh+17] audits search engines for differential satisfaction between demographics. [Eks+18] extends this investigation to several public recommendation datasets, discussing whether different groups of users (in terms of age or gender) retrieve the same utility from recommendations based on standard metrics. Such differences

may be due to class imbalance, which may lead a recommender system to better capture the interaction patterns of a majority group in a collaborative filtering setting [Mel+21]. Our audit takes into account these concerns by interrogating gender differences in terms of recall, as well as by measuring the fit of job recommendations to job seekers' search criteria.

(Conditional) recommendation independence Other works emphasize the trade-off between recommendation performance and other fairness measures. Among them, [Kam+12] approach the problem of collaborative filtering under the lenses of a notion of neutrality akin to demographic parity: recommendations should be independent from a user-specified viewpoint such as gender. In a similar fashion, [Isl+21], concerned with occupation recommendation, consider a differential fairness metric akin to demographic parity in the classification setting - in other words, the recommended occupation should not depend on gender. In occupation and job recommendation contexts respectively, [Rus+22] and [Li+23] consider the aggregate difference in wages between recommendations to men and women (the unconditional gender wage gap) as a fairness metric. As noted by [Eks+18], these objectives of recommendation independence are typically framed in terms of representational harms - the goal being to avoid conveying undesirable gender stereotypes in recommendations - as well as the real-world impact of reproducing gender inequalities. Our audit is also primarily concerned with recommendation dependence of job ad characteristics such as wage on gender, and documents such average gaps in recommendations. Yet, as noted in [Rus+22]'s discussion, "the disparate behavior of typical recommendation systems (...) may partly reflect legitimately differing real-world preferences in career choices by women and men". From a methodological point of view, drawing from the economics literature on gender gaps [SW21], we propose to also document, and focus our discussion, on gender gaps conditional on characteristics (qualifications and/or preferences), proposing to measuring these gaps using doubly robust estimators [RRZ94; Che+17; GQ10]. From the empirical point of view, we provide evidence of gender gaps in job recommendations through the scrutiny of a large-scale, real-world dataset of relevance for other PES settings. This focus on gaps in terms of conditional recommendation independence is also related with audit studies in economics. In particular, [Zha21] audits recommendations on Chinese job boards in an audit study setting (creating worker profiles differing only by gender), demonstrating the existence of gendered differential treatment.

Algorithmic interventions Interventions aimed at ensuring algorithmic fairness definitions are satisfied can be divided into the categories of pre-processing (adjust data before training), in-processing (adapt model training to include fairness-related aspects) and post-processing (modify predictions or rankings, taking a trained model as granted). In particular, adversarial in-processing methods [ES15; WVP18; Beu+17; Let+23], which attempt to decorrelate latent representations with gender, have been proposed for neural recommender systems in labor market settings [Rus+22; Isl+21; Li+21] with different motivations and notions of fairness in mind. On the other hand, post-processing methods have the merit of being model-agnostic.

As in the present work, [Li+23] cast the problem of recommendation selection under gender gap constraints as an integer linear programming (ILP) problem. While the authors also consider quantity constraints on the number of recommendations per job ads (which would be non-trivial to accommodate in our proposed framework), their use of off-the-shelf ILP solvers scales poorly to large-scale problems[3]. One of our contributions consists in addressing this limitation by noticing that the structure of the linear program of interest lends itself to Lagrangian relaxation [Fis81]. Furthermore, the proposed framework (presented in section 7.5) is extended to conditional gap reduction by inverse propensity weighting.

7.2 Measuring gender gaps in recommendations

7.2.1 Outcomes of interest

We consider three sets of outcomes defined at the job seeker level (generically denoted Y in the following).

Recommendation performance First, we seek to measure how the algorithm's top-k recommendation performance varies between men and women, which will be measured by the recall@k, evaluated on hires. Thus, for a job seeker in the test set which was hired at a given point in time, we will define his or her contribution to the recall as $Y_i = 1$ if the person's future job was correctly ranked in the algorithm's top-k recommendations, and 0 otherwise.

Characteristics of recommended jobs We also wish to describe how the characteristics of the top k job ads[4] depend on gender, unconditionally and conditionally on qualifications and preferences. We will thus consider the following average characteristics of the top k recommended job ads: i) the logarithm of the ad's wage; ii) the distance in kilometers of the job's workplace to the job seeker's zip code; iii) whether the job ad corresponds to an executive position in the company; iv) whether the contract is defined for an indefinite duration or not; v) whether the contract is full-time; vi) the experience required for the job (in months). k will be set to 10 in the experiments.

Fit to job seeker's search criteria We also consider an aggregate indicator of the fit between the job seeker's search criteria and the recommended jobs, defined as an average of five binary indicators describing the fit w.r.t. to the job seeker's i) accepted geographic mobility; ii) desired type of occupation; iii) desired wage; iv) desired type of contract; v) desired working hours.

[3]Their experiments are conducted on 3,000 users and 5,105 companies - several orders of magnitude below our real-world setting of interest.

[4]Note that this measure fixes the number k of recommended jobs by job seekers, and gives each of these k jobs equal importance. Other measures weighting recommended jobs based on their ranks (*e.g.* with DCG-type weights), either among the top-k ads or throughout the full ranking of all job ads, could also be relevant, and would be compatible with our methodology for the analysis of gender gaps.

7.2.2 Measuring unconditional and conditional gaps

Average gaps In order to first assess whether the algorithm recommends different job ads to men and women, we consider the naive average characteristics Y of the recommended offers:

$$\delta = \mathbb{E}[Y|G=1] - \mathbb{E}[Y|G=0],$$

$G=1$ and $G=0$ denoting respectively women and men hereafter. This parameter can simply be estimated by taking a difference in means[5].

Conditional gaps However, part of the gaps δ might be shaped by characteristics of job seekers that could be considered legitimate to take into account when generating recommendations. Conditioning on such characteristics Z may provide more insights on the composition of the raw gender gaps δ. If the measured gaps are to be interpreted in a normative fashion (*i.e.* as fairness measures), defining what variables Z should be conditioned upon is a crucial choice.

In the following, we consider two specifications for Z. In the first specification, Z represents the qualifications of job seekers, including education, experience, and job type. In the second specification, control variables include both qualifications and job seekers' stated job search criteria - for instance, their search criteria in terms of reservation wage, commuting time, type of contract, and working hours; this augmented set of controls will be denoted Z_p. Table 2.6 in Chapter 2 lists the features used to define Z and Z_p in more detail.

Denote $m_0(z) = \mathbb{E}[Y|Z=z, G=0]$ the expected characteristics of recommended jobs for men with features z, and $m_1(z) = \mathbb{E}[Y|Z=z, G=1]$ the expected characteristics of recommended jobs for women with features z. We are then interested in estimating the average conditional gender gap:

$$\tau = \mathbb{E}[m_1(Z) - m_0(Z)]$$

Note that the expectation is taken over the distribution of Z in the entire population of job seekers, pooling together men and women.[6]

Estimation of conditional gaps Estimating such average conditional gender gaps is closely linked to the problem of estimating average treatment effects given differences in population characteristics in the causal estimation literature [IR15], granted that in our setting, gender is not literally a "treatment". It is also linked to the so-called Kitagawa-Oaxaca-Blinder gender gaps decomposition in the economics literature [Kit55; Oax73; Bli73; SW21].

[5] This focus on unconditional, or later conditional, gender gaps may hide heterogeneity (a particular form of which may take the form of intersectionality). Other features of the distribution of outcomes than its average may also be of interest.

[6] τ is thus related to δ by:

$$\delta = [(\mathbb{E}[m_1(Z)|G=1] - \mathbb{E}[m_1(Z)]) - (\mathbb{E}[m_0(Z)|G=0] - \mathbb{E}[m_0(Z)])] + \tau$$

Two approaches can be used to estimate τ. The *direct method* forms estimates $\hat{m}_1(z)$ and $\hat{m}_0(z)$ of conditional expectations $m_1(z)$ and $m_0(z)$ using standard supervised learning techniques to form an estimate:

$$\hat{\tau}^{DM} = \frac{1}{n}\sum_i (\hat{m}_1(z_i) - \hat{m}_0(z_i)).$$

where n is the population size. The *inverse propensity score* approach instead proceeds by re-weighting each sample by an estimate of the inverse of the propensity score $e_g(z_i) = \mathbb{P}(G_i = g|Z = z_i)$ [7]:

$$\hat{\tau}^{IPW} = \frac{1}{n}\sum_i \left(\frac{y_i 1\{g_i = 1\}}{\hat{p}_1(z_i)} - \frac{y_i 1\{g_i = 0\}}{\hat{p}_0(z_i)}\right).$$

Both methods are consistent under standard assumptions on the direct method models or the propensity score, but present drawbacks. The direct method approach is model-dependent, whereas the inverse propensity weighting method depends on the propensity score model and has poor small sample properties when propensity scores are close to zero or one. Doubly robust, or "augmented inverse propensity weighting" estimators [RRZ94; Che+17; GQ10] combine both approaches, forming estimates of the form:

$$\hat{\tau}^{AIPW} = \frac{1}{n}\sum_i \left(\hat{m}_1(z_i) + \frac{y_i - \hat{m}_1(z_i)}{\hat{e}_1(z_i)} 1\{g_i = 1\}\right) - \left(\hat{m}_0(z_i) + \frac{y_i - \hat{m}_0(z_i)}{\hat{e}_0(z_i)} 1\{g_i = 0\}\right)$$

This approach has the merit of being consistent either of the estimators $\hat{m}$ or $\hat{p}$ consistently estimate m and p, and is robust to associated estimation errors.

7.2.3 Discussion

Differential treatment between individuals by an algorithm may arise from preexisting biases in the training dataset. This section discusses how the proposed gender gap measures may relate to actual economic behavior and to existing normative stances on fairness.

A simple model of application and hiring Let us first analyze the application and hiring behavior in a simple economic model (a simplified version of the model in Chapter 4). Let x denote the vector representing job seeker characteristics (*e.g.* qualifications, preferences, and other relevant attributes) and, with a slight overload of notations, y represent the characteristics of job advertisements (*e.g.* proposed wage, contract type, and job location). Job seekers are risk neutral, and decide whether or not to apply to jobs in order to maximize their expected utility. We denote $U(x, y) + \nu$ the amount of utility the job seeker x would perceive if hired for the job offer y, where ν is an unobserved random part. If a job seeker applies, she or he has a probability $p(x, y)$ of actually being hired for the job. However, as in Chapter 4, job seekers do not necessarily know the true success chances of their

[7] One may re-normalize the weights so they sum to one, as we do in practice.

applications $p(x, y)$. Instead, they form their own estimates $\pi(x, y)$ of $p(x, y)$, which may differ from the actual value. When a job seeker applies to a job ad, he or she incurs a fixed cost c. Let us also assume that that if the job seeker does not apply to any offer, they receive a utility amount of zero. In this model, the job seeker x decides to apply to job ad y if and only if his expected utility is positive:

$$\text{(Decision applying)} \qquad \underbrace{\pi(x, y)(U(x, y) + \nu) - c}_{\text{Expected utility when applying}} \geq \underbrace{0}_{\text{Utility without applying}}$$

The probability of observing an application of x on a job ad of type y is

$$\text{(Probability of observing an application)} \qquad A(x, y) = F_{-\nu}\left(U(x, y) - \frac{c}{\pi(x, y)}\right),$$

where $F_{-\nu}$ denotes the CDF of $-\varepsilon$. The probability of observing a hiring is simply the product of the probability of application times the objective probability of a positive output after the interview: $H(x, y) = A(x, y)p(x, y)$.

When the cost of application is zero, *i.e.* $c = 0$, only utility matters in job seekers' decisions. Otherwise, their expected chances $\pi(x, y)$ that their application succeeds weigh their utility gains and could censor their decision of applying. This simply underlines that realized applications are then not a pure expression of preferences, but also mix with possibly wrong expectations.

Possible explanations of gender gaps We may observe different labor market outcomes for men and women, leading to different recommendations learnt in the algorithm's training process, if they differ in terms of utility $U(x, y)$, bias in estimating their success chances $\pi(x, y)$, or in actual application success chance $p(x, y)$.

First, preferences U might be gender-specific. For example, possibly due to social norms and other constraints, women may appreciate the relative values of commuting time and wages differently from men [LRR21], or prefer or be forced to work for less hours due to childcare [GPO22]. The data collected on job seekers' declared preferences and presented in Panel B of Table F.1 in the Appendix within our sample supports this hypothesis of gendered preferences. On average, women tend to seek job opportunities located, on average, 5 kilometers closer to their place of residence. Their average reservation wage (the minimum salary accepted to work) is €230 per month less than that of men. Additionally, women are less often searching for full-time contracts, with only 64% pursuing such positions compared to 83% of men.

Second, even if job seekers hold rational expectations, there might be gendered differences in the hiring chances π [GMR15], *e.g.* taste or statistical discrimination against a gender by recruiters. For instance, in the French context, a large-scale correspondence study [Arn+21] highlights the existence of heterogeneous gender biases in callback rates in different industries.

Third, hiring expectations π might also be gendered: aside from a possible anticipation of discrimination from recruiters, there might be differences in the perceptions and the representations of the chances to be hired, leading to differences in self-censorship or over-confidence [Cor+23]; as well risk aversion (or different values for the cost of an application c in the model).

Gender gaps and fairness While this work is motivated by common definitions of fairness applicable to the labor market, and while we believe the proposed gender gap measures to be valuable in the audit of algorithms with respect to these notions, we make no normative claim on what a fair recommender should be, and as to whether the measured gaps are to be directly interpreted as measures of (un)fairness.

As explained above, the proposed definitions of gender gaps are closely linked to demographic parity and recommendation independence in the algorithmic fairness literature, as well as to issues of distributional and representational harm. Our exposition nevertheless also puts emphasis on conditional measures of gaps, which could be mapped to conditional fairness measures - as some elements of a job seekers' profiles, such as qualifications and / or stated preferences, could possibly be deemed legitimate for use by a recommender system, despite being distributed differently across genders for (possibly unfair) historical reasons.

Measured gender gaps also imperfectly map to the structural economic parameters linked above, as it is unclear whether the contributions of gender differences in p, π, U or c, or a mixture thereof in gaps can be disentangled. This is an issue with respect to fairness notions emphasizing the origins of gender gaps: one may for instance enable recommendations to allow for differences in preferences, but forbid differences due to discrimination by recruiters. Such a discussion would require learning a structural model of the labor market that is out of the scope of this work. Nevertheless, let us first note that conditioning on a set Z containing stated preferences might partially account for gender differences in utility U. Secondly, as tentative evidence to describe the role that gender differences in acceptance probability p may play in generating gaps in historical data, we documents gender gaps not only in hires (the algorithm's training data) but also in job seekers' applications (where only π, U and c are at play according to the model).

7.3 Experimental setting

Datasets We study the *France Travail* dataset relative to the Auvergne-Rhône-Alpes region from 2019 to mid-2022, as described in Chapter 2. The train and test set cover 1.2 million job seekers and 2.2 million job ads. The 285,992 observed hires are split between train and test on a weekly basis: 85% of weeks are assigned to the train set (representing 241,715 hires), and the rest to the test set (44,277 hires, 46.66% of which include men).

To study gender gaps conditionally on features Z, the analysis must be restricted to men and women that cannot be perfectly distinguished on the basis of Z, following the *overlap / common support* assumption [IR15]. More precisely, if individuals' gender could be perfectly predicted on the basis of Z, one could hardly disentangle the impact of Z and that of gender on the recommendations.

The population with common support is selected as follows. The prediction of gender is achieved using a logistic regression considering selected features including education, desired wage, experience, geographic location, desired contract type, occupation, level of qualification, search for part-time job, accepted mobility. The learned classifier, referred to as *propensity score*, with accuracy circa 85%, is used to

select job seekers in the common support, retaining individuals with the propensity score in $[0.01, 0.99]$.

To study recommendations issued to all job seekers at a given point in time, we consider all job seekers registered during a randomly chosen week of the test set (the fourteenth week of 2022). In order to measure recommendation performance and to contrast differentiated treatment by the algorithm with differences observed in hiring behavior, we also consider recommendations to all job seekers which are hired during the test weeks. To study application behavior, we consider the average characteristics (all weeks pooled together) of the applications of job seekers with applications in the test weeks.

The sizes, compositions in terms of gender, and size after restriction to job seekers in the common support, of the datasets of interest are reported in Table F.2 in the appendix. The distribution of propensity scores among our population is given in appendix F.3.

7.4 Results

7.4.1 Recommendation performance is higher for women

Table 7.1 reports the recall@k for all hires in the test set, as well as for men and women separately. The recall@10 is 24% for men, and 26.5% for women, with a statistically significant difference. More generally, we find the recall@k to be higher for women than for men at all values of k considered. While the magnitude of the difference is limited, it is statistically significant. [8]

Top k	Recall@k	Men	Women	p-value
1	0.0573	0.0546	0.0597	0.0281
5	0.1744	0.1641	0.1833	0.0000
10	0.2532	0.2401	0.2647	0.0000
20	0.3468	0.3279	0.3632	0.0000
50	0.4843	0.4670	0.4995	0.0000
100	0.5834	0.5680	0.5968	0.0000

Notes: Recall@k is the recall for the population for the first top recommendations k. The columns “Men" and “Women" present the same recall@k separately for men and women. The last column performs a test of equality between columns 2 and 3.

Table 7.1: Difference in recall between genders.

[8]Hypothesized reasons for this disparity in recall include sample imbalance (women slightly outnumber men) and the role of the distance criterion (women may assign greater value to proximity when searching for a job, see Table 7.4 on hires and applications, which could make their job choices easier to predict). In preliminary experiments, re-training the model on a balanced sample, and controlling for distance and/or characteristics Z, reduces but does not completely eliminate the gap.

7.4.2 Characteristics of job ads recommended to men and women are different

Table 7.2 provides conditional and unconditional estimates of gender gaps for a subset of characteristics of the recommended job ads. The first column provides the mean average difference between women and men (for the entire job-seeking population of the fourteenth week of 2022). On average, women are recommended job ads that have different characteristics than those recommended to men. Their recommended job ads are paid 2.1% less than men; 350 meters closer to home, less often in full-time contract (19.7 percentage points); less often of indefinite duration (4 percentage points less often) and executive status (0.5 percentage points); require less experience (1.7 year on average). Jobs recommended to women also have a lower degree of fit with their own search criteria (a loss of 0.031 points in the aggregate fit measure between 0 and 1). All of these differences are statistically significant. These results also hold for the population of common support (which has a propensity score that ranges between 0.01 and 0.99) presented in the third column of the table (Uncond. δ (overlap)).

We now turn to the analysis of conditional gaps, using the estimators described in Section 7.2, and conditioning on qualifications Z. Our analysis focuses on the population that fulfills the common support assumption, which comprises job seekers with propensity scores ranging between 0.01 and 0.99. The fifth column (Cond. τ (IPW)) and the seventh column (Cond. τ (DRL)) respectively present the inverse propensity weighting estimator and doubly robust estimators. Conditioning for job seeker's characteristics Z using the IPW estimator leads to a reduced gender gap for all job ads characteristics considered. However, the recommended jobs for women still fit less with their search parameters (by 0.014 points) and remain significantly different in all the dimensions discussed except for executive positions. For example, 33% of the gender wage gap is left unexplained by the characteristics and qualifications of the job seekers. The same conclusions hold when the doubly robust estimator is used to compute the gender gaps. The minimal disparities between the two estimators suggest the robustness of our results.

In conclusion, even after controlling for a set of observable factors defining job seeker key qualifications, women and men continue to receive job recommendations that differ in their attributes.

Including job seekers' preferences into the control vector Z As discussed in section 7.2, incorporating stated preferences into Z (we will denote this augmented vector Z_p) in addition to objective qualifications could partially explain gender differences due to differences in utility. Table 7.3 presents an analysis similar to that of Table 7.2, but conditioning on Z_p instead of Z. Including preferences in the control vector helps to better explain gender gaps in the recommendations generated. The portion of the gender gap left unexplained diminishes across most job offer characteristics, although it remains significant except for executive positions and experience. For instance, the unexplained part of the gender wage gap now stands at only 23% (compared to 33% when conditioning on Z), yet it remains statistically significant. Overall, incorporating preferences into the control vector

	Uncond. δ (full pop.)	pval	Uncond δ (overlap)	pval	Cond. τ IPW	pval	Cond. τ DRL	pval
Wage (log)	-0.021	0.0	-0.018	0.0	-0.006	0.000	-0.006	0.0
Distance	-0.350	0.0	-0.219	0.0	0.639	0.000	0.674	0.0
Executive Position	-0.005	0.0	-0.008	0.0	-0.001	0.117	-0.002	0.0
Indefinite duration	-0.040	0.0	-0.044	0.0	-0.016	0.000	-0.020	0.0
Full time	-0.197	0.0	-0.165	0.0	-0.045	0.000	-0.044	0.0
Experience (months)	-1.696	0.0	-1.199	0.0	-0.166	0.000	-0.170	0.0
Fit to job search parameters	-0.031	0.0	-0.027	0.0	-0.014	0.000	-0.016	0.0

Notes : The first column reports the average gender gaps δ recommendations in the total population. The third column reports the gender gap in recommendations for job seekers with a propensity score between 0.01 and 0.99, i.e., belonging to the common support. The fifth column represents the conditional gaps measured by the inverse propensity weighting estimator. The seventh column reports the estimates using doubly robust estimators. The results are given using random forests as estimators for the function m and logistic regression for p. All "pval" columns present the p-value indicating the significance of the measure reported in the adjacent left column. For conditional estimators, the p-value is computed by bootstrapping.

Table 7.2: Unconditional and conditional gender differences in the characteristics of the recommended offers.

results in minimal changes in estimated gender gaps compared to controlling for qualifications alone.

	Uncond. δ (full pop.)	pval	Uncond δ (overlap)	pval	Cond. τ IPW	pval	Cond. τ DRL	pval
Wage (log)	-0.021	0.0	-0.017	0.0	-0.004	0.000	-0.004	0.000
Distance	-0.350	0.0	-0.206	0.0	0.681	0.000	0.742	0.000
Executive Position	-0.005	0.0	-0.008	0.0	0.001	0.292	-0.000	0.121
Indefinite duration	-0.040	0.0	-0.044	0.0	-0.009	0.000	-0.017	0.000
Full time	-0.197	0.0	-0.160	0.0	-0.030	0.000	-0.030	0.000
Experience (months)	-1.696	0.0	-1.170	0.0	-0.077	0.129	-0.030	0.683
Fit to job search parameters	-0.031	0.0	-0.025	0.0	-0.008	0.000	-0.011	0.000

Notes : The first column reports the gender gap δ on average in the total population. The third column reports the gender gap in recommendations for job seekers with a propensity score between 0.01 and 0.99, i.e., belonging to the common support. The fifth column represents the conditional gaps measured by the inverse propensity weighting estimator. The seventh column reports the estimates using doubly robust estimators. The results are given using random forests as estimators for the function m and logistic regression for p. All "pval" columns present the p-value indicating the significance of the measure reported in the adjacent left column. For conditional estimators, the p-value is computed by bootstrapping.

Table 7.3: Unconditional and conditional gender differences in the characteristics of the recommended offers (when conditioning on qualifications and preferences Z_p).

Heterogeneity Average gaps can hide substantial heterogeneity. To provide suggestive evidence about possible heterogeneity, we identify, for each type of job ad characteristic, the top decile of the population in terms for which the doubly robust scores (Eq. 7.2.2) differ the most in disfavor of women - in other words, the job seekers with qualifications Z for which the largest gender gaps disfavoring women are predicted by the learned models. Table F.5 provides descriptive statistics on the demographic characteristics of these top deciles - with column 1 providing population means for reference.

Based on column 2 (*Wage*), we observe that compared to the population mean, the subgroup experiencing the most pronounced wage loss associated to being a woman (approximately 2.5% average loss) exhibits different characteristics. These

individuals possess a higher level of experience (78 months compared to 53 months in the overall population), are more frequently in executive positions (16.5% compared to 9.4%), are more often married (43% compared to 37%), and have higher reservation wages (2440 euros compared to 1923 euros). Moreover, they are more often aged between 30 to 50 years (58% compared to 54%). In terms of sectors, the most affected group is more often looking for jobs in Industry (27% compared to 8% in the general population).

7.4.3 Comparison of gender gaps in recommendations, hirings and applications

We now turn to the comparison of gender gaps in recommendations to those found in the training data, namely hires; and to those found in job seekers' own application behavior.

Comparison with hires To compare gender gaps in recommendations to those found in hiring behavior, we focus on all hires that occurred during the test weeks[9]. When job seeker i is hired on ad j^* during week t, we compare the characteristics of j^* with those of the top-10 recommendations i would have received at time t. Table 7.4, panel A, displays the results of this comparison.

The first column of Table 7.4, Panel A (τ_{Hire}), corresponds to IPW estimates of gender gaps conditional on qualifications *in hiring data* (*i.e.* in terms of the actual hires j^*)[10], on the population with common support. Conditional on qualifications, women are hired on jobs that are paid less (1.2 percentage points)[11], less often in indefinite duration contracts (4 percentage points) and less often in full time contract (7.4 percentage points). Moreover, women obtain jobs with a lower aggregate fit to their search parameters compared to men (by 0.02 points). These gender gaps are statistically significant.

The third column ($\tau(MUSE)$) of Table 7.4, Panel A, presents the corresponding estimates (gender gaps conditional on qualifications on the common support population) on the recommendations hired job seekers would have received at the time t of their hire. Similarly, women are found to receive recommendations with lower wages (0.7 percentage points), a lower share of indefinite duration contracts (1

[9]As noted above, we pool hires from all weeks, rather than restraining the analysis to the fourteenth week of 2022, for the sake of statistical power. We thus study a total population of 41, 787 individuals (22, 291 of which are women).

[10]Table F.4 in the Appendix provides estimates of all quantities of Table 7.4 using doubly robust estimators, rather than IPW ones, as a robustness check.

[11]An estimate of 1% for the gender wage gap on the job offers, conditional on qualifications, might be surprising considering the larger magnitudes generally discussed in the economics literature. It should be noted that we condition on a large set of variables and that the analysis focuses on registered job seekers (rather than on the working population as a whole), with jobs closer to the national minimum wage than those in the national population. Moreover, reported wages are those posted in job openings (more precisely, an average between an lower and upper bound for the proposed wage, when the upper bound is available). Gaps in wages on the job may be larger than the reported ones due to gendered behavior in negotiations between the job seeker and the employer, as emphasized by [Rou24].

	Differences between women and men				Difference of Differences	
A. In hirings	$\tau_{\text{Hire}}(Observed)$	p-value	τ (MUSE)	p-value	τ_{DifH} (MUSE)	p-value
Wage (log)	-0.012	0.000	-0.007	0.001	0.005	0.162
Distance	-2.517	0.130	0.812	0.000	3.329	0.032
Executive Position	-0.001	0.534	-0.001	0.689	0.000	0.682
Indefinite duration	-0.040	0.001	-0.010	0.032	0.029	0.048
Full time	-0.074	0.000	-0.044	0.000	0.030	0.000
Experience (months)	0.504	0.359	0.175	0.123	-0.329	0.791
Fit to job search parameters	-0.020	0.000	-0.016	0.000	0.003	0.211
B. In applications	τ_{App} (Observed)	p-value	τ (MUSE)	p-value	τ_{DifA} (MUSE)	p-value
Wage (log)	-0.011	0.000	-0.005	0.004	0.006	0.002
Distance	-6.409	0.000	0.648	0.000	7.056	0.000
Executive Position	-0.003	0.220	0.001	0.240	0.004	0.095
Indefinite duration	-0.025	0.001	-0.009	0.011	0.016	0.062
Full time	-0.076	0.000	-0.051	0.000	0.025	0.000
Experience (months)	-0.132	0.312	0.115	0.230	0.247	0.113
Fit to job search parameters	-0.020	0.000	-0.015	0.000	0.004	0.013

Notes: In Panel A, the results are presented for the subsample of hired job seekers with hires in the testing weeks. Panel B presents results on the subsample of job seekers for which we observe applications in the testing weeks. The first column presents the conditional estimates for the gender gaps on observed hirings (resp. observed applications) between women and men for the population with common support. The third one presents the same difference on the characteristics of the algorithm's recommendations. Differences with the conditional effects presented in the fifth column of Table 7.2 are due to the restriction on the subsample of job seekers with hires / applications. The fifth column reports the difference of two latter differences, i.e., the conditional estimates for the differences between a hire's characteristics (resp application's) and the algorithm's recommendation.

Table 7.4: Conditional gender gaps in hires and applications and in the algorithm's recommendations on the subsample of hired job seekers

percentage points) and full time jobs (4.4 percentage points), and lower fit to their search parameters (0.016 points). These gender gaps are statistically significant.

Yet, when comparing the magnitude of gender gaps in hires and in recommendations, we note that those in recommendations are not wider than those observed in hiring. Column τ_{DifH} of Table 7.4, Panel A, provides the results of a formal comparison of the gender gaps in recommendations and in hirings, achieved by running IPW estimates on Y taken as *the difference between the characteristics of MUSE recommendations and those hires.* An estimate of zero would indicate that the conditional gaps remain unchanged in recommendations, whereas positive coefficients indicate conditional difference between men and women are reduced in recommendations compared to hires. The algorithm's recommendations are found to significantly reduce gaps in terms of indefinite duration and full time recommendations. Altogether, the algorithm either maintains the gender gap unchanged compared to its training data, or, sometimes, narrows it.

Comparison with applications Panel B of Table 7.4 replicates the exercise of panel A, comparing the algorithm's recommendations to job seekers' application behavior (which underly hiring behavior). The analysis is conducted on the average characteristics of job seekers' applications over all test weeks (compared to the average characteristics of the corresponding recommendations over all test weeks).

When conditioning on qualifications, women tend to apply to positions that offer lower pay (0.01 percentage points), shorter commute distances (6km closer), fewer long-term contracts (0.025 percentage points) and fewer full-time positions (7.6 percentage points). Additionally, they apply to positions that align less with their search parameters by 0.02. All of these gaps are statistically significant. The algorithm does not exacerbate the gender gaps observed in application behavior (Column 5, τ_{Diff}). For most job characteristics, we observe a positive coefficient, with several of them being statistically significant (wage, distance, and full time nature of the job). This indicates that the algorithm tends to narrow the gender gap observed in application behavior for most characteristics.

Eventually, if the algorithm recommends different types of job offers to men and women, there is no evidence suggesting that it exacerbates the existing gender gaps observed in the labor market, whether in hiring behavior or in application behavior, when we control for job seekers' qualifications.

7.5 A post-processing approach to reducing gender gaps in recommendations

The goal of this section is twofold. First, we present a scalable post-processing approach to reduce unconditional or conditional average gender gaps in terms of recommendation characteristics, casting the selection of recommendations as a Lagrangian relaxation of an integer linear programming problem. Second, we describe the trade-offs it implies between standard performance measures and achieving reduced gender gaps in recommendations.

7.5.1 Methodology

We cast the problem of recommendation selection under gender gap constraints as an integer linear programming (ILP) problem. We circumvent the issue of scalability by noticing that the structure of the linear program of interest lends itself to Lagrangian relaxation [Fis81].[12]

Unconstrained top-k recommendation as an integer linear program Consider n users and m job ads registered at a given point in time. Assume that for each job seeker - job ad pair $i - j$, we have access to the recommender system's

[12] An earlier workshop paper this chapter is based on [Bie+23a] rather used an adversarial in-processing approach at the pair level to decorrelate latent representations of job seeker - job ad pairs from gender during training, as *e.g.* in [Rus+22]. Both approaches have merits. In general, in-processing approaches have the potential to be more efficient than post-processing ones. Adversarial training would also target distributions (of pairwise scores) rather than averages. On the other, the ILP approach used here optimizes overall rankings whereas the adversarial one operates at the pair level (translation from scores to rankings comes without guarantees). Moreover, the ILP approach is straightforward to extend to conditional gaps; is more explicit in terms of its gaps targets and enforcing trade-offs than the adversarial approach; is model-agnostic and does not require retraining.

score s_{ij} [13]. Formally, selecting the set of the top k job ads in terms of s_{ij}'s for each job seeker i may be written as solving the integer linear program:

$$\max_{\{\gamma_{ij}\}_{i\leq n, j\leq m}} \sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij}\gamma_{ij} = \max_{\gamma\in\mathbb{R}^{nm}} \mathbf{s}^T\gamma \tag{7.1}$$

subject to:

$$\gamma_{ij} \in \{0,1\}, \quad \forall i,j, \qquad \sum_{j=1}^{m}\gamma_{ij} = k, \quad \forall i \tag{7.2}$$

Here, $\gamma_{ij} = 1$ signifies that job ad j would be chosen among job seeker i's top k ads; and $\gamma_{ij} = 0$ that job ad j is not shown to job seeker i. Contraints 7.2 simply encode that exactly k job ads must be selected to feature in each job seeker's recommendations. Equation 7.1 indicates that recommendations should have the highest scores according to the baseline recommender system. [14]

Top-k recommendation with constraints on unconditional average gender gaps Let $g_i = 1$ if i is a women, 0 otherwise, and let n_{g_i} be the number of individuals with the same gender as i. Let w_{ij} be a characteristic of job ad j, possibly depending on i, in terms of which gender gaps should be restricted in recommendations (for instance, w_{ij} can set to be the wage of job ad j, or the geographic distance between j and i). This can be achieved by adding to the integer linear program described above the additional constraint:

$$-\epsilon \leq \frac{1}{k}\sum_{i:g_i=1}\sum_{j}\frac{w_{ij}}{n_{g_i}}\gamma_{ij} - \frac{1}{k}\sum_{i:g_i=0}\frac{w_{ij}}{n_{g_i}}\gamma_{ij} \leq \epsilon \tag{7.3}$$

where $\epsilon > 0$ should ideally be chosen by consensus among stake-holders (public decision-makers, job seekers and recruiters). This equation encodes that the gender gap in terms of $w'_{ij}s$ of recommendations must be lower than ϵ in absolute value. This gender gap constraint can be rewritten as $\mathbf{A}\gamma \leq \mathbf{b}$, $\mathbf{A} \in \mathbb{R}^{2\times nm}$, $\mathbf{b} = (\epsilon,\epsilon)^T$ (details of the transformation are provided in Appendix F.4). The formulation easily extends to constraints on gaps in terms of several features w_{ij}'s (*e.g.* constraints on both the wage and distance gaps).

Top-k recommendation with constraints on conditional gender gaps If, rather than unconditional gaps, gender gaps conditional on covariates Z are to be targeted, one may substitute to constraint (7.3) the alternative constraint:

$$-\epsilon \leq \frac{1}{k(\sum_{i:g_i=1}\frac{1}{\hat{e}_1(z_i)})}\sum_{i|g_i=1}\sum_{j}\frac{w_{ij}}{\hat{e}_1(z_i)}\gamma_{ij} - \frac{1}{k(\sum_{i:g_i=0}\frac{1}{\hat{e}_0(z_i)})}\sum_{i:g_i=0}\sum_{j}\frac{w_{ij}}{\hat{e}_0(z_i)}\gamma_{ij} \leq \epsilon \tag{7.4}$$

[13] In the following, s_{ij} could be replaced by any transformation of scores and initial ranks (for instance, an NDCG-like formula based on initial rank) that would be deemed to express in a meaningful way how relevant job ad i is for job seeker j.

[14] This integer linear programming problem selects which k job ads should be shown, but does not specify their order. Optimizing rankings (to account for the so-called position bias) is important in practice, but is left for further work.

where $\hat{e}_g(z_i)$ corresponds to an estimate of the propensity score $e_g(z_i) = P(G = g|Z = z_i)$. In other words, features w_{ij} are weighted by the inverse of the propensity score in order to take into account the differences in covariates Z between men and women.

A scalable Lagrangian relaxation approach While the size of the integer linear programming problem (in nm variables) is forbidding, its structure lends itself to the use of Lagrangian relaxation. Indeed, this hard problem can be viewed as an easy problem - solving (7.1) under constraint (7.2) - complicated by side constraints (7.3) or (7.4), which can be dualized. We thus consider the Lagrangian relaxation

$$L(\lambda) = \max_{\gamma \in \mathbf{R}^{nm}} \mathbf{s}^T\gamma - \lambda(\mathbf{b} - \mathbf{A}\gamma) \qquad \text{s.t} \qquad \gamma_{ij} \in \{0,1\}, \quad \forall i,j, \qquad \sum_j \gamma_{ij} = k, \quad \forall i$$

For a given value of λ, solving the problem defining $L(\lambda)$ simply amounts, for each job seeker, to the unconstrained problems of finding the highest k values in a size m array (as detailed in Appendix F.4). This task can be achieved in $O(m + k\log k)$ (expected) complexity for a given job seeker, and thus for a total complexity of $O(n(m + k\log k))$ for all job seekers. The overall Lagrangian relaxation problem we seek to solve for is then:

$$\min_{\lambda \geq 0} L(\lambda)$$

which we optimize by subgradient descent, noting that a subgradient of $L(\lambda)$ with respect to λ is $-(\mathbf{b} - \mathbf{A}\gamma)$. In practice, we reduce the problem to the top-200 recommendations of each job seeker i, which can be interpreted in the problem above as adding the constraint $\gamma_{ij} = 0$ must hold for recommendations ranked above 200. This restriction stems from a pragmatic perspective, and further eases the computational burden.

Discussion Recommendation independence requires $Y \perp G$, which is a stronger requirement than the restriction on $|\mathbb{E}[Y|G = 1] - \mathbb{E}[Y|G = 0]|$ studied here in the non-conditional setting. The two approaches are equivalent for binary variables. For continuous variables (*e.g.* wage), requiring both genders to have the same distributions can be approximated in the proposed approach by discretizing the distributions, and simultaneously imposing constraints on the share of recommendations in each discretized bucket (since the framework can accommodate several constraints).

These remarks can be extended to the conditional setting, replacing $Y \perp G$ by $Y \perp G|Z$, and $|\mathbb{E}[Y|G = 1] = \mathbb{E}[Y|G = 0]|$ by $|\mathbb{E}[Y|G = 1|Z] = \mathbb{E}[Y|G = 0|Z]|$.

Ensuring $Y \perp Z|Z = z$ for all z is a question of primary interest but is left for further work (one way to work towards satisfying this constraint in our setting would be to create clusters among Z and solve integer linear programming programs in each cluster).

7.5.2 Results

Experimental setting This section investigates the consequences on recommendation characteristics and recall of reducing gender gaps using the methodology

proposed in Section 7.5. A first subsection studies the reduction of average (unconditional) gaps, while the second subsection focuses on analyzing the reduction of conditional gaps.

In both cases, three sets of constraints ϵ, of increasing stringency, are implemented for the sake of illustration. These constraints are simultaneously imposed on several objective characteristics of recommended job ads: log-wage, distance, executive qualification, indefinite duration contract, full time status and experience requirements. Values of the constraints are summarized in Table F.6 in the appendix. The constraints are imposed on the top $k = 10$ recommendations.

Constrained recommendations are computed for all weeks in the test set, on a week by week basis. Consistently with the methodology employed in the first part of this chapter, all results except those for recall are presented on the fourteenth week of 2022, and the recall is computed on the full set of test weeks to ensure sufficient statistical power.

Aside from gender gaps and recall, we also document the share of job seekers for which post-processing causes at least one recommendation to change compared to unconstrained recommendations, and the share of pairs of job seekers and job ads in top-10 recommendations that are modified after post-processing. Since one might fear that the modification of the γ_{ij}'s comes at the price of increased congestion of recommendations, our analysis also takes into account congestion measures. Congestion is measured by the Gini Index, taking values between 0 and 1. A lower Gini Index indicates lower congestion (*i.e.* a more equal spread of recommendations across all available job ads).

Constraining average (unconditional) gaps Table 7.5 presents the consequences of imposing constraints on unconditional gender gaps, for the three considered levels of ϵ. For each level of constraint stringency (Low, Intermediate and High), column ϵ displays the constraints that are simultaneously imposed on the different types of gender gaps, whereas column "value" displays the actual gender gaps achieved in recommendations after post-processing. The average gender gaps in the attributes of the recommended job ads align with the specified constraints, as required. Surprisingly, the gender gap in terms of fit of recommendations to job search parameters (which does *not* feature in the required constraints) also decreases.

The second panel of Table 7.5 displays the recall@10 for all hirings in the test set, segmented by gender. The recall is 0.253 for unconstrained recommendations, and decreases slightly when constraints are imposed - reaching 0.2484 for "high" constraint levels. Both genders experience a decline in recall.

The third panel of Table 7.5 presents other relevant descriptive statistics. Correcting for average gaps appears to slightly increase congestion, with the Gini Index rising from an initial value of 0.7877 to 0.7900 in the case of high constraints on average gaps. Depending on constraint stringency, from 68% to 72% of job seekers experience at least one change in the recommendations they receive due to post-processing. However, only 11% to 12% of the total pairs present in the unconstrained rankings are modified after post-processing, suggesting that the modifications of displayed rankings at the individual level remain limited on average.

	Initial		Low			Intermediate			High		
	value	p-value	ϵ	value	p-value	ϵ	value	p-value	ϵ	value	p-value
Unconditional gaps											
Wage (log)	-0.0213	0.0	0.0050	-0.0050	0.0000	0.0025	-0.0025	0.0000	0.0010	-0.0010	0.0020
Distance	-0.4293	0.0	0.5000	-0.3541	0.0000	0.2500	-0.2500	0.0000	0.1250	-0.1250	0.0000
Executive Position	-0.0048	0.0	0.0025	0.0022	0.0000	0.0010	0.0011	0.0035	0.0005	0.0005	0.1597
Indefinite duration	-0.0404	0.0	0.0100	-0.0100	0.0000	0.0050	-0.0050	0.0000	0.0025	-0.0025	0.0069
Full time	-0.1965	0.0	0.0100	-0.0101	0.0000	0.0050	-0.0051	0.0000	0.0025	-0.0026	0.0007
Experience (months)	-1.8079	0.0	1.0000	-1.0000	0.0000	0.5000	-0.5000	0.0000	0.2500	-0.2500	0.0000
Fit to job search parameters	-0.0308	0.0		-0.0001	0.8031		0.0017	0.0008		0.0027	0.0000
Performance indicators											
R@10	0.2530			0.2498			0.2487			0.2484	
R@10 (Women)	0.2645			0.2620			0.2608			0.2605	
R@10 (Men)	0.2398			0.2358			0.2349			0.2345	
Other descriptive statistics											
Gini Index	0.7877			0.7896			0.7898			0.7900	
% Offers modified	0.0000			0.1110			0.1172			0.1210	
% Ranking modified	0.0000			0.6777			0.7045			0.7208	

Notes: The results are presented using recommendations generated for job seekers (and job ads) existing at week 2022-14, the overall population of job seekers being taken into account. We present results for increasingly stringent unconditional constraints ϵ ("weak", "intermediate", and "high"). The set of ϵ values is given for each intensity level, respectively, in the third, sixth, and ninth columns. The first two columns restate the results of the standard algorithm in terms of average gender gaps for comparison purposes. p-values are computed via bootstrap estimation (100 randomly selected samples drawn from the initial population).

Table 7.5: Unconditional gender gaps in job ads characteristics after imposing unconditional constraints

Distributional changes As discussed above, for continuous variables such as the wage, constraining the mean alone is a weaker constraint than requiring equal distributions for both genders. In particular, one may be worried that adjustment only occurs through the manipulation of a few outlying values. Figure 7.1 displays the impact of the selected constraints on the distribution of wages for men and women, to further assess the distributional changes that occur when imposing average constraints. Applying the mean constraints impacts the wage distribution as a whole, and not a few outliers. However, the distributions do not fully overlap after post-processing.

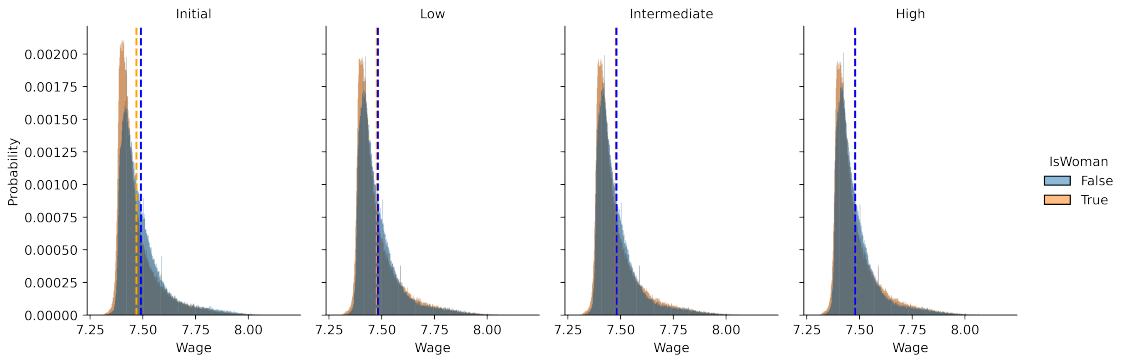


Figure 7.1: Distribution of Wages (log) according to stringency of applied constraints

	Initial		Low			Intermediate			High		
	value	p-value	ϵ	value	p-value	ϵ	value	p-value	ϵ	value	p-value
Conditional gaps											
Wage (log)	-0.0065	0.0000	0.0050	-0.0052	0.0000	0.0025	-0.0027	0.0006	0.0010	-0.0012	0.1594
Distance (km)	0.7175	0.0000	0.5000	0.4986	0.0000	0.2500	0.2473	0.0035	0.1250	0.1215	0.2934
Executive position	-0.0007	0.1965	0.0025	-0.0002	0.8132	0.0010	0.0005	0.3090	0.0005	0.0003	0.4852
Indefinite duration	-0.0153	0.0000	0.0100	-0.0097	0.0000	0.0050	-0.0047	0.0204	0.0025	-0.0022	0.2878
Full time	-0.0450	0.0000	0.0100	-0.0102	0.0000	0.0050	-0.0052	0.0217	0.0025	-0.0027	0.2526
Experience (months)	-0.2070	0.0000	1.0000	-0.1429	0.0025	0.5000	-0.0898	0.0601	0.2500	-0.0591	0.2189
Fit to job search parameters	-0.0141	0.0000		-0.0084	0.0000		-0.0070	0.0000		-0.0061	0.0000
Performance indicators											
R@10	0.2388			0.2386			0.2388			0.2387	
R@10 (Women)	0.2512			0.2512			0.2515			0.2515	
R@10 (Men)	0.2236			0.2232			0.2233			0.2231	
Other descriptive statistics											
Gini Index	0.7969			0.7969			0.7969			0.7969	
% Offers modified	0.0000			0.0048			0.0066			0.0074	
% Ranking modified	0.0000			0.0451			0.0614			0.0687	

Notes: The results are presented using recommendations generated for job seekers (and job ads) for week 2022-14, the overall population of job seekers being taken into account. We present results for increasingly stringent conditional constraints ϵ ("weak", "intermediate", and "high"). The set of ϵ values is given for each intensity level, respectively, in the third, sixth, and ninth columns. The first two columns restate the results of the standard algorithm in terms of average gender gaps for comparison purposes. p-values are computed via bootstrap estimation (100 randomly selected samples drawn from the initial population).

Table 7.6: Conditional gender gaps in job ads characteristics after imposing conditional constraints

Constraining conditional gaps We now turn to the analysis of the impact of imposing conditional constraints. We focus on the analysis of gaps conditional on the specification of Z containing qualifications only. Complementary results for the specification including both qualifications and preferences are provided in the Appendix (Table F.7).

Table 7.6 describes the impact of constraining average gaps conditional on qualifications. The measures for gaps are *conditional* ones (conditioning by qualifications), for the sake of coherence with the required constraints. The post-processed recommendations result in a reduction in the conditional gender gaps, aligned with the enforced constraint ϵ. Surprisingly, gender gaps in terms of fit to job search parameters also decrease as a by-product of the imposition of the other constraints.

The imposition of conditional constraints has negligible impact on the recall@10, both on average and for either gender.

This is attributed to the fact that fulfilling the require constraints is achieved with little modifications to overall recommendations. For instance, on the population of the fourteenth week of 2022, the most stringent constraints ϵ on conditional gaps are achieved by modifying less than 1% of the initial job seeker-job ad pairs recommended, and the post-processed rankings remain unchanged for more than 93% of job seekers. The congestion in recommendations remains unchanged by post-processing. We emphasize that the number of recommended jobs k, set to 10 for these experiments, might affect the performance-fairness trade-off.

7.6 Partial conclusion

In this chapter, we build on the literature in labor economics to propose and measure gender gaps in job recommendations, emphasizing the relevance of conditional gap measures. These gap measures are furthermore applied in the context of the audit of Muse.

Our main findings are the following. First, we find recall to be slightly higher for women than for men. Second, we provide evidence of differentiated treatment of men and women by the algorithm in terms of recommended job characteristics (in particular, wages, contract lengths, full time job status), even conditionally on job seekers' qualifications and search criteria. However, the algorithm's recommendations does not increase gendered gaps observed in hirings, and even decreases them in some cases. A comparison of recommended job ads to application behavior leads to similar conclusions. Third, we propose a scalable post-processing approach, under the form of the Lagrangian relaxation of an integer linear program, to reduce unconditional and conditional gender gaps. We empirically demonstrate the method's ability to reduce conditional gender gaps, and investigate the trade-offs between recall and gender gaps it entails.

Our analysis has focused on average gaps, which could hide substantial heterogeneity - in particular, intersectional biases. Future work could consider other features of the distribution of gaps, both in terms of measure and post-processing targets.

One of the drawbacks of our analysis is that we relied on the declared preferences of job seekers, summarized by a few variables (*e.g.* desired job and contract type, maximum geographic mobility). Such declarative variables might be too minimalist to truthfully reflect the complexity of job seekers' trade-offs between the different components of a job ad. A way to improve our analysis would be to collect more information on these preferences, and find a way to elicit them empirically, *e.g.* through experiments exposing job seekers to job ads with randomized components.

Ultimately, the merits of de-biased algorithms attempting to reduce gender gaps in recommendations hinge on the acceptability of the proposed job ads in terms of job seekers' (possibly gendered) preferences. An algorithm straying off too far from job seekers' search behavior might lead to a dead-weight loss: a loss in recommendation quality without any effect on labor market inequalities if recommendations are simply discarded as irrelevant. Moreover, formal guarantees for gender gaps in recommendations do not imply bounds on the actual impact of an algorithm's implementation on gender gaps in applications and hirings. Measuring the algorithm's capacity to actually modify labor market decisions toward fairer outcomes requires experiments in practice, for instance under the form of A/B tests comparing post-processed recommendations to a suitably chosen benchmark.

Major limitations of the inquiry we conducted must be acknowledged. First, the analysis has focused on a single protected attribute (gender, moreover treated as binary due to data limitations), leaving aside other groups for which fairness concerns exist (*e.g.* age, ethnicity), and possible issues of intersectionality. Second, our discussion of fairness should be extended to the value alignment question and tied to the investigation conducted in Chapter 4. Indeed, the choice of an algorithm's

objective function is itself a key problem in discussing fairness issues [KA21; Kas24]. Moreover, our analysis has focused on the narrow, technical viewpoint of AI fairness, despite its inherent limitations [BD24]. A complete analysis of fairness issues with job recommender systems should take into account the full socio-technical scope of their deployment, how these tools are used and understood by users, and how they contribute to shaping the broader relations between job seekers, employers and government services [CP22].

Conclusion and perspectives

Conclusion

This work aimed at exploring problems related to the design of job recommender systems in the context of a collaboration with the French Public Employment Service, *France Travail*.

Our first contribution is MUSE, a two-tiered neural architecture aimed at addressing the challenges of learning from sparse interactions and heterogeneous data sources while ensuring scalability. MUSE's first tier is a "two-tower" recommender system, with separate embeddings accounting for the specific roles of geography and of skill proximity in labor market matching. The second tier re-ranks the first tier's selection, leveraging more elaborate features and a more complex architecture with multiplicative interactions. MUSE is comparatively assessed to the state of the art on public data from the Xing 2017 RecSys Challenge as well as *France Travail* data. It is shown to have strong performances in recall while maintaining scalability (several orders of magnitude compared to boosted tree ensembles). MUSE was also validated in the context of two large-scale randomized field experiments, which gathered feedback from job seekers on recommendations. Variants of the two-tier MUSE architecture learned from hires and applications were shown to significantly outperform boosted tree ensembles as well as the institution's current expert systems on explicit and implicit satisfaction metrics.

Our second contribution is a discussion of the objective that job recommender systems should pursue in order to be aligned with job seekers' goals. Based on a formal economic model, we argued that such a well-aligned algorithm would rank job ads by their expected utility for job seekers - a quantity that involves the utility job seekers would derive from a job posting, and their probability of being hired conditional on applying. Using proxies for job seekers' utility and hiring probabilities, we documented the fact that the rankings they induced significantly differed, underlining that this algorithm design problem is important in practice. Efficiently combining utility and hiring probabilities in recommendations could lead to sizeable gains for job seekers, although identifying these quantities is non-trivial.

Our third contribution questions the issue of congestion in job recommender systems. As job ads are rival goods, recommending the same subset of job ads to all job seekers might be sub-optimal, however good the recommendations may individually seem. We found congestion to be a likely issue in practice: in a given occupation, despite job seekers outnumbering job ads by a factor of roughly eight, only a quarter of all job ads appeared at all in any job seekers' top-ten recommen-

dations. Leveraging tools from the computational optimal transport literature, we proposed CAROT, a post-processing approach to congestion-reducing recommendation. Comparative validation on public and *France Travail* data investigated CAROT's ability to balance recall and congestion.

Our fourth contribution is linked to the measure of gender gaps in recommendations, in the context of concerns about the algorithm's fairness with respect to gender. We discussed possible origins for these gaps, which may come from job seekers (different utility functions, risk aversion, over or under-confidence) as well as from discrimination from recruiters. Leveraging ideas from the gender gaps literature in economics, we proposed tools to measure gender gaps in recommendations, possibly conditioning on job seekers' qualifications and preferences. An audit of MUSE led to the conclusion that gender gaps existed in recommendations, but were no greater than those observed in the training data (hires), nor than those observed in job seekers' applications. We also proposed a scalable post-processing approach to reducing (possibly conditional) average gender gaps, leveraging the Lagrangian relaxation of an integer linear program. We investigated the trade-offs between gender gap reduction and recall the approach entailed.

Perspectives

Noting that contribution-specific research perspectives were outlined in partial conclusions at the end of relevant chapters of this thesis, let us now focus on a few selected avenues for further work.

Improving MUSE's scope and architecture To improve the scope and quality of MUSE recommendations, the most obvious perspectives are to scale the approach nationwide, and to remove key limitations on the data MUSE leverages (as described in Chapter 2), which in turn requires improvement in MUSE's architecture. MUSE could be modified to leverage more hires of job seekers than only those identified in our work (less than a tenth of the total), perhaps using multi-instance learning. More elaborate natural language processing tools, perhaps including Large Language Models, could be leveraged to better represent and leverage textual data.

Towards an economic understanding of job recommendation A key theme underlying this work is that job recommendation is *not* an usual learning to rank problem.

Past query-level or pair-level data is not *i.i.d.* in the sense that a "good" recommendation or match depends on context - the state of the labor market, and local supply and demand of labor. On the algorithmic side, seeking to account for this may take the form of posing the problem as a *reciprocal recommendation* one, and learning from *labor market data viewed as a graph.*

A better understanding of *how data is generated and what constitutes optimal interventions* is also required from the point of view of economic theory - separating the role of utility, perceived and actual hiring probabilities, and competition in the data generating process, and modeling the welfare impact of the introduction of

recommendations on the labor market. This improved understanding could then *be leveraged into the recommendation process* by mixing structural assumptions with the non-parametric flexibility of machine learning to identify key components in job seekers' decisions (hiring probabilities, utility), although this comes with statistical and econometric challenges.

Communication and behavioral aspects Another key perspective is understanding how best to convey recommendations to job seekers.

For instance, what key pieces of information on job ads should be presented to job seekers? Especially when learning from hires, some elements underlying recommendations (such as how they were computed, and/or the probability of being hired conditional on application), might be valuable for job seekers, potentially representing a key complement to provided rankings. Yet, how can this information be conveyed given uncertainty on the probability's estimation quality, and without any risk of psychological harm?

Moreover, especially if recommendations are identified as generated by an algorithm, the documented phenomenon of *aversion* to algorithms could reduce job seekers' interest in the proposed job ads. To what extent does this phenomenon exist in the job recommendation context? Can communication variants highlighting selected features of algorithms reduce aversion?

These questions are also crucially linked to those of explainability and interpretability. Seeking to propose simple and relevant explanations of recommendations is an important and open problem, due to the complexity of leveraged architectures and features in multi-tiered recommender systems, and because faithfully explaining *rankings* is arguably harder than simply plugging in local post-hoc explainability tools (*e.g.* LIME or SHAP). Ensuring interpretability by design may be a way forward.

Moreover, we have so far treated job seekers as passive consumers of lists of recommendations. Could algorithmic tools giving job seekers agency in *exploring* currently posted job ads in an intuitive and interactive fashion also be relevant?

Large-scale impact evaluation A main perspective is a randomized evaluation of the causal impact of Muse' deployment on the labor market, including job seekers' search behavior, return to, and quality of employment. A proper experiment design may enable the measure of externalities (such as congestion and displacement effects), for instance by randomizing the level of exposure of local labor markets to recommendations. Such a study would thus be relevant to the RecSys community, to labor economics, and in terms of policy implications. An improved understanding of how job seekers react to recommendations would greatly further our understanding of many themes discussed throughout the thesis, including congestion and fairness.

Exploration Whether algorithms learn from clicks, hires or applications, the data they learn from is noisy and biased (in a statistical sense): job seekers may suboptimally look for jobs, probably missing relevant opportunities; and their past labor market opportunities depend on past states of the labor market. Principled

exploration, *e.g.* under the form of contextual bandits, may improve recommendations and provide positive side effects in terms of diversity and fairness [LRB20; Cow18]. Exploration may also be an attractive alternative to a structural econometric approach to value alignment, by identifying promising recommendation opportunities not tried in past data under weaker assumptions. It may also provide a path towards lifelong learning (adaptation to cohort changes). This comes with many algorithmic challenges, which include seeking to build strong priors from past data; measuring uncertainty in large-scale, complex neural recommender systems; and exploiting feedback (*e.g.* hiring data) which may be sparse and only available on the long run, without excessively degrading user experience.

Beyond job recommendation Further opportunities exist for machine learning to play a useful role on the labor market, especially in a PES setting.

First, while the economic literature studying interventions targeting recruiters remains limited [ACG20; BHK24; Lu18], recommending job seekers to recruiters has a large potential for impact, yet raises even more acute (and different) fairness issues.

Second, caseworkers play an essential role in the missions of PESs to bridge the gap between caseworkers and firms, yet they are under overwhelming demand. Given caseworkers' current portfolio of job seekers and current job ads on the markets, machine learning methods (with a special concern for interpretability, and perhaps a human-in-the-loop approach) could help caseworks identify job seeker, or job seeker - job ad pairs, to focus on.

Finally, the recommendation of training opportunities, requiring counter-factual causal reasoning, could help tackle structural rather than frictional unemployment.

Publications

[Bie+21] Guillaume Bied, Elia Pérennès, Victor Alfonso Naya, Philippe Caillou, Bruno Crépon, Christophe Gaillac, and Michele Sebag. "Congestion-Avoiding Job Recommendation with Optimal Transport". In: *FEAST workshop ECML-PKDD 2021*. 2021.

[Bie+23a] Guillaume Bied, Philippe Caillou, Bruno Crépon, Christophe Gaillac, Elia Pérennès, and Michèle Sebag. "Designing labor market recommender systems: the importance of job seeker preferences and competition". Working paper. 2023.

[Bie+23b] Guillaume Bied, Christophe Gaillac, Morgane Hoffmann, Philippe Caillou, Bruno Crépon, Solal Nathan, and Michele Sebag. "Fairness in job recommendations: estimating, explaining, and reducing gender gaps". In: *AEQUITAS 2023-First AEQUITAS Workshop on Fairness and Bias in AI, co-located with ECAI 2023*. Vol. 3523. CEUR-WS.org. 2023.

[Bie+23c] Guillaume Bied, Solal Nathan, Elia Pérennès, Morgane Hoffmann, Philippe Caillou, Bruno Crépon, Christophe Gaillac, and Michèle Sebag. "Toward Job Recommendation for All". In: *Thirty-Second International Joint Conference on Artificial Intelligence {IJCAI-23}*. Macau, China: International Joint Conferences on Artificial Intelligence Organization, Aug. 2023, pp. 5906–5914. DOI: 10.24963/ijcai.2023/655. URL: https://hal.science/hal-04245528.

References

[Abe+16] Fabian Abel, András Benczúr, Daniel Kohlsdorf, Martha Larson, and Róbert Pálovics. "Recsys challenge 2016: Job Recommendations". In: *Proceedings of the 10th ACM Conference on Recommender Systems*. 2016, pp. 425–426.

[Abe+17] Fabian Abel, Yashar Deldjoo, Mehdi Elahi, and Daniel Kohlsdorf. "Recsys challenge 2017: Offline and Online Evaluation". In: *Proceedings of the 11th ACM Conference on Recommender Systems*. 2017, pp. 372–373.

[ACG20] Yann Algan, Bruno Crépon, and Dylan Glover. "Are Active Labor Market Policies Directed at Firms Effective? Evidence from a Randomized Evaluation with Local Employment Agencies". J-PAL Working paper. 2020.

[Agg16] Charu Aggarwal. *Recommender Systems*. Vol. 1. Springer, 2016.

[Alt+18] Steffen Altmann, Armin Falk, Simon Jäger, and Florian Zimmermann. "Learning about Job Search: A Field Experiment with Job Seekers in Germany". In: *Journal of Public Economics* 164 (2018), pp. 33–49.

[Alt+22] Steffen Altmann, Anita Glenny, Robert Mahlstedt, and Alexander Sebald. "The Direct and Indirect Effects of Online Job Search Advice". In: *SSRN Electronic Journal* (2022).

[Arn+21] Emilie Arnoult, Marie-Odile Ruault, Emmanuel Valat, Pierre Villedieu, Thomas Breda, Nicolas Jacquemet, Morgane Laouenan, Roland Rathelot, Mirna Safi, Clara Schaeper, Joyce Sultan Parraud, Amélie Allegre, Anna Bagramova, Frédérique Bouvier, Fabrice Foroni, Sara Ftoh Fennane, Isabelle Huet, Bianka Kozma, Amine Medaghri Alaoui, and Elshaday Tekle. *Gender discrimination in hiring: Lessons from a large-scale correspondence test*. Ed. by Institut des politiques publiques. IPP Policy brief, n° 67. 2021. URL: https://shs.hal.science/halshs-03524771.

[ASL23] Bissan Audeh, Maia Sutter, and Christine Largeron. "Comparative Study of Unsupervised Keyword Extraction Methods for Job Recommendation in an Industrial Environment". In: *International Conference on Research Challenges in Information Science*. Springer. 2023, pp. 551–558.

[Aut01] David Autor. "Wiring the Labor Market". In: *Journal of Economic Perspectives* 15.1 (2001), pp. 25–40.

[Bab+12] Linda Babcock, William J Congdon, Lawrence F Katz, and Sendhil Mullainathan. "Notes on Behavioral Economics and Labor Market Policy". In: *IZA Journal of Labor Policy* 1 (2012), pp. 1–14.

[Ban+22] Abhijit Banerjee, Bruno Crépon, Elia Pérennès, and Cécile Walter-Médée. "Using stated individual preferences for job ads' attributes to design a better matching algorithm between job ads and job-seekers." In: *AEA RCT Registery* (2022). DOI: 10.1257/rct.8719.

[BC22] Abhijit Banerjee and Gaurav Chiplunkar. "How important are matching frictions in the labour market? Experimental & non-experimental evidence from a large indian firm". In: *Money* (2022).

[BCC17] Alejandro Bellogín, Pablo Castells, and Iván Cantador. "Statistical biases in Information Retrieval Metrics for Recommender Systems". In: *Information Retrieval Journal* 20 (2017), pp. 606–634.

[BD24] Maarten Buyl and Tijl De Bie. "Inherent Limitations of AI Fairness". In: *Communications of the ACM* 67.2 (2024), pp. 48–55.

[Beh+22] Luc Behaghel, Sofia Dromundo Mokrani, Marc Gurgand, Yagan Hazard, and Thomas Zuber. *Encouraging and Directing Job Search: Direct and Spillover Effects in a Large Scale Experiment*. Tech. rep. Banque de France, 2022.

[Beu+17] Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. "Data Decisions and Theoretical Implications when Adversarially Learning Fair Representations". In: *arXiv preprint arXiv:1707.00075* (2017).

[Beu+18] Alex Beutel, Paul Covington, Sagar Jain, Can Xu, Jia Li, Vince Gatto, and Ed H Chi. "Latent Cross: Making Use of Context in Recurrent Recommender Systems". In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. 2018, pp. 46–54.

[BFH21] Monica Bhole, Andrey Fradkin, and John Horton. *Information About Vacancy Competition Redirects Job Search*. Tech. rep. Center for Open Science, 2021.

[BHK24] Ines Black, Sharique Hasan, and Rembrand Koning. "Hunting for talent: Firm-driven labor market search in the United States". In: *Strategic Management Journal* 45.3 (2024), pp. 429–462.

[BHN23] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and machine learning: Limitations and opportunities*. MIT press, 2023.

[BHS22] Aurélien Bellet, Amaury Habrard, and Marc Sebban. *Metric Learning*. Springer Nature, 2022.

[BK07] Robert M Bell and Yehuda Koren. "Lessons from the Netflix Prize Challenge". In: *ACM SIGKDD Explorations Newsletter* 9.2 (2007), pp. 75–79.

[BKM19] Michele Belot, Philipp Kircher, and Paul Muller. "Providing advice to jobseekers at low cost: An experimental study on online advice". In: *The Review of Economic Studies* 86.4 (2019), pp. 1411–1447.

[BKM22] Michele Belot, Philipp Kircher, and Paul Muller. "Do the Long-Term Unemployed Benefit from Automated Occupational Advice During Online Job Search?" In: *SSRN Electronic Journal* (Jan. 2022). DOI: 10.2139/ssrn.4178928.

[Bli73] Alan S Blinder. "Wage Discrimination: Reduced Form and Structural Estimates". In: *The Journal of Human Resources* (1973), pp. 436–455.

[Bro23] Stijn Broecke. "Artificial intelligence and labour market matching". In: *OECD Social, Employment and Migration Working Papers* 284 (2023).

[Bur+05] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. "Learning to rank using gradient descent". In: *Proceedings of the 22nd International Conference on Machine learning*. 2005, pp. 89–96.

[Bur02] Robin Burke. "Hybrid Recommender Systems: Survey and Experiments". In: *User Modeling and User-Adapted Interaction* 12 (2002), pp. 331–370.

[Bur10] Christopher JC Burges. "From RankNet to LambdaRank to LambdaMART: An Overview". In: *Learning* 11.23-581 (2010), p. 81.

[BZK17] Fedor Borisyuk, Liang Zhang, and Krishnaram Kenthapadi. "LiJAR: A System for Job Application Redistribution towards Efficient Career Marketplace". In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2017, pp. 1397–1406.

[Cao+07] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. "Learning to Rank: From Pairwise Approach to Listwise Approach". In: *Proceedings of the 24th International Conference on Machine learning*. 2007, pp. 129–136.

[CG16] Tianqi Chen and Carlos Guestrin. "Xgboost: A scalable tree boosting system". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016, pp. 785–794.

[Che+16] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. "Wide & Deep learning for Recommender Systems". In: *Proceedings of the 1st workshop on deep learning for recommender systems*. 2016, pp. 7–10.

[Che+17] Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, and Whitney Newey. "Double/Debiased/Neyman Machine Learning of Treatment Effects". In: *American Economic Review* 107.5 (2017), pp. 261–265.

[Che+18] Victor Chernozhukov, Mert Demirer, Esther Duflo, and Ivan Fernandez-Val. *Generic Machine Learning Inference on Heterogenous Treatment Effects in Randomized Experiments*. Tech. rep. National Bureau of Economic Research, 2018.

[CHL19] Kuan-Ming Chen, Yu-Wei Hsieh, and Ming-Jen Lin. "Prediction and Congestion in Two-sided Markets: Economist versus Machine Matchmakers". In: *SSRN Electronic Journal* (2019).

[CHL23] Kuan-Ming Chen, Yu-Wei Hsieh, and Ming-Jen Lin. "Reducing Recommendation Inequality via Two-Sided Matching: A Field Experiment of Online Dating". In: *International Economic Review* (2023).

[CK20] Michael Cooper and Peter Kuhn. "Behavioral Job Search". In: *Handbook of Labor, Human Resources and Population Economics* (2020), pp. 1–22.

[Cor+23] Patricia Cortés, Jessica Pan, Laura Pilossoph, Ernesto Reuben, and Basit Zafar. "Gender Differences in Job Search and the Earnings Gap: Evidence from the Field and Lab". In: *The Quarterly Journal of Economics* 138.4 (2023), pp. 2069–2126.

[Cow18] Bo Cowgill. *Bias and Productivity in Humans and Algorithms: Theory and Evidence from Resume Screening*. Tech. rep. Columbia Business School, Columbia University, 2018.

[CP22] Hadrien Clouet and Jean-Marie Pillon. "Chapitre 7: Forcer le destin. Le travail d'appariement dans les services publics de l'emploi français et allemand". In: *Comment ça matche? Une sociologie de l'appariement* (2022), pp. 251–293.

[Cré+13] Bruno Crépon, Esther Duflo, Marc Gurgand, Roland Rathelot, and Philippe Zamora. "Do Labor Market Policies have Displacement Effects? Evidence from a Clustered Randomized Experiment". In: *The Quarterly Journal of Economics* 128.2 (2013), pp. 531–580.

[CS06] Eugene Choo and Aloysius Siow. "Who Marries Whom and Why". In: *Journal of Political Economy* 114.1 (2006), pp. 175–201.

[CS16] Pierre-André Chiappori and Bernard Salanié. "The Econometrics of Matching Models". In: *Journal of Economic Literature* 54.3 (2016), pp. 832–861.

[Cut13] Marco Cuturi. "Sinkhorn Distances: Lightspeed Computation of Optimal Transport". In: *Advances in Neural Information Processing Systems* (2013), pp. 2292–2300.

[DB21] Corné De Ruijt and Sandjai Bhulai. "Job Recommender Systems: A Review". In: *arXiv preprint arXiv:2111.13576* (2021).

[Dee+90] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. "Indexing by latent semantic analysis". In: *Journal of the American Society for Information Science* 41.6 (1990), pp. 391–407.

[DG14] Arnaud Dupuy and Alfred Galichon. "Personality Traits and the Marriage Market". In: *Journal of Political Economy* 122.6 (2014), pp. 1271–1319.

[Dia82] Peter A Diamond. "Wage Determination and Efficiency in Search Equilibrium". In: *The Review of Economic Studies* 49.2 (1982), pp. 217–227.

[Do+21] Virginie Do, Sam Corbett-Davies, Jamal Atif, and Nicolas Usunier. "Two-sided fairness in rankings via Lorenz dominance". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 8596–8608.

[Dwo+12] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. "Fairness Through Awareness". In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. 2012, pp. 214–226.

[Eks+18] Michael D Ekstrand, Mucun Tian, Ion M Azpiazu, Jennifer D Ekstrand, Oghenemaro Anuyah, David McNeill, and Maria S Pera. "All the Cool Kids, How Do They Fit In?: Popularity and Demographic Biases in Recommender Evaluation and Effectiveness". In: *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*. PMLR. 2018, pp. 172–186.

[Eks+22] Michael D Ekstrand, Anubrata Das, Robin Burke, and Fernando Diaz. "Fairness in Information Access Systems". In: *Foundations and Trends® in Information Retrieval* 16.1-2 (2022), pp. 1–177.

[EP22] Michael D Ekstrand and Maria S Pera. "Matching Consumer Fairness Objectives & Strategies for RecSys". In: *arXiv preprint arXiv:2209.02662* (2022).

[ES15] Harrison Edwards and Amos Storkey. "Censoring representations with an adversary". In: *arXiv preprint arXiv:1511.05897* (2015).

[FC21] Mauricio N Freire and Leandro N de Castro. "e-Recruitment recommender systems: a systematic review". In: *Knowledge and Information Systems* 63 (2021), pp. 1–20.

[Fie+23] Erica Field, Robert Garlick, Nivedhitha Subramanian, and Kate Vyborny. "Why Don't Jobseekers Search More? Barriers and Returns to Search on a Job Matching Platform". Working paper. 2023.

[Fis81] Marshall L Fisher. "The Lagrangian relaxation method for solving integer programming problems". In: *Management science* 27.1 (1981), pp. 1–18.

[FNO22] Brian Feld, AbdelRahman Nagy, and Adam Osman. "What do jobseekers want? Comparing methods to estimate reservation wages and the value of job attributes". In: *Journal of Development Economics* 159 (2022), p. 102978.

[GAK19] Sahin C Geyik, Stuart Ambler, and Krishnaram Kenthapadi. "Fairness-Aware Ranking in Search & Recommendation Systems with Application to LinkedIn Talent Search". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2019. DOI: 10.1145/3292500.3330691.

[Gal18] Alfred Galichon. *Optimal Transport Methods in Economics*. Princeton University Press, 2018.

[Gee19] Laura K Gee. "The More You Know: Information Effects on Job Application Rates in a Large Field Experiment". In: *Management Science* 65.5 (2019), pp. 2077–2094.

[GMR15] Laurent Gobillon, Dominique Meurs, and Sébastien Roux. "Estimating Gender Differences in Access to Jobs". In: *Journal of Labor Economics* 33.2 (2015), pp. 317–363.

[GMZ13] Stanislao Gualdi, Matúš Medo, and Yi-Cheng Zhang. "Crowd Avoidance and Diversity in Socio-Economic Systems and Recommendations". In: *Europhysics Letters* 101.2 (2013), p. 20008.

[Gol+92] David Goldberg, David Nichols, Brian M Oki, and Douglas Terry. "Using collaborative filtering to weave an information tapestry". In: *Communications of the ACM* 35.12 (1992), pp. 61–70.

[GPO22] Claudia Goldin, Sari Pekkala Kerr, and Claudia Olivetti. *When the Kids Grow Up: Women's Employment and Earnings across the Family Cycle*. Working Paper 30323. National Bureau of Economic Research, 2022. DOI: 10.3386/w30323.

[GQ10] Adam N Glynn and Kevin M Quinn. "An Introduction to the Augmented Inverse Propensity Weighted Estimator". In: *Political analysis* 18.1 (2010), pp. 36–56.

[GS21] Alfred Galichon and Bernard Salanié. "Structural Estimation of Matching Markets with Transferable Utility". In: *arXiv preprint arXiv:2109.07932* (2021).

[GS22] Alfred Galichon and Bernard Salanié. "Cupid's invisible hand: Social surplus and identification in matching models". In: *The Review of Economic Studies* 89.5 (2022), pp. 2600–2629.

[Guo+17] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. "DeepFM: a Factorization-Machine based Neural Network for CTR Prediction". In: *arXiv preprint arXiv:1703.04247* (2017).

[Gut+19] Francisco Gutiérrez, Sven Charleer, Robin De Croon, Nyi Nyi Htun, Gerd Goetschalckx, and Katrien Verbert. "Explaining and exploring job recommendations: a user-driven approach for interacting with knowledge-based job recommender systems". In: *Proceedings of the 13th ACM Conference on Recommender Systems*. 2019, pp. 60–68.

[He+17] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. "Neural Collaborative Filtering". In: *Proceedings of the 26th International Conference on World Wide Web*. 2017, pp. 173–182.

[Her+04] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. "Evaluating Collaborative Filtering Recommender Systems". In: *ACM Transactions on Information Systems (TOIS)* 22.1 (2004), pp. 5–53.

[HHA10] Gunter J Hitsch, Ali Hortaçsu, and Dan Ariely. "Matching and Sorting in Online Dating". In: *American Economic Review* 100.1 (2010), pp. 130–63.

[HKV08] Yifan Hu, Yehuda Koren, and Chris Volinsky. "Collaborative Filtering for Implicit Feedback Datasets". In: *2008 Eighth IEEE International Conference on Data Mining*. IEEE. 2008, pp. 263–272.

[Hsi+17] Cheng-Kang Hsieh, Longqi Yang, Yin Cui, Tsung-Yi Lin, Serge Belongie, and Deborah Estrin. "Collaborative Metric Learning". In: *Proceedings of the 26th International Conference on World Wide Web*. 2017, pp. 193–201.

[IR15] Guido W Imbens and Donald B Rubin. *Causal Inference in Statistics, Social, and Biomedical Sciences*. Cambridge University Press, 2015.

[Isl+21] Rashidul Islam, Kamrun N Keya, Ziqian Zeng, Shimei Pan, and James Foulds. "Debiasing Career Recommendations with Neural Fair Collaborative Filtering". In: *Proceedings of the Web Conference 2021*. 2021, pp. 3779–3790.

[Jay+20] Siddhant M Jayakumar, Wojciech M Czarnecki, Jacob Menick, Jonathan Schwarz, Jack W Rae, Simon Osindero, Yee Whye Teh, Tim Harley, and Razvan Pascanu. "Multiplicative Interactions and Where to Find Them". In: *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. 2020.

[KA21] Maximilian Kasy and Rediet Abebe. "Fairness, Equality, and Power in Algorithmic Decision-Making". In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. 2021, pp. 576–586.

[Kam+12] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. "Enhancement of the Neutrality in Recommendation". In: *Decisions@ RecSys*. 2012, pp. 8–14.

[Kas24] Maximilian Kasy. "The Political Economy of AI: Towards Democratic Control of the Means of Prediction". In: *SSRN Electronic Journal* (2024).

[KB14] Diederik P Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[KB19] Mahmut Kaya and Hasan Şakir Bilge. "Deep Metric Learning: A Survey". In: *Symmetry* 11.9 (2019), p. 1066.

[Kir22] Philipp Kircher. "Job Search in the 21St Century". In: *Journal of the European Economic Association* 20.6 (Oct. 2022), pp. 2317–2352. ISSN: 1542-4766. DOI: 10.1093/jeea/jvac057.

[Kit55] Evelyn M. Kitagawa. "Components of a Difference Between Two Rates". In: *Journal of the American Statistical Association* 50.272 (1955), pp. 1168–1194. DOI: 10.2307/2281213.

[KLD18] Bo Kang, Jefrey Lijffijt, and Tijl De Bie. "Conditional Network Embeddings". In: *arXiv preprint arXiv:1805.07544* (2018).

[KM14] Peter Kuhn and Hani Mansour. "Is Internet Job Search Still Ineffective?" In: *The Economic Journal* 124.581 (2014), pp. 1213–1233.

[KMR23] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. "The Challenge of Understanding What Users Want: Inconsistent Preferences and Engagement Optimization". In: *Management Science* (2023).

[Kon+97] Joseph A Konstan, Bradley N Miller, David Maltz, Jonathan L Herlocker, Lee R Gordon, and John Riedl. "Grouplens: Applying collaborative filtering to Usenet news". In: *Communications of the ACM* 40.3 (1997), pp. 77–87.

[KP17] Matevž Kunaver and Tomaž Požrl. "Diversity in recommender systems–A survey". In: *Knowledge-Based Systems* 123 (2017), pp. 154–162.

[Kul13] Brian Kulis. "Metric Learning: A Survey". In: *Foundations and Trends® in Machine Learning* 5.4 (2013), pp. 287–364.

[Kum+23] Deepak Kumar, Tessa Grosz, Navid Rekabsaz, Elisabeth Greif, and Markus Schedl. "Fairness of recommender systems in the recruitment domain: an analysis from technical and legal perspectives". In: *Frontiers in Big Data* 6 (2023).

[Lav21] Dor Lavi. "Learning to Match Job Candidates using Multilingual Bi-encoder BERT". In: *Proceedings of the 15th ACM Conference on Recommender Systems*. 2021, pp. 565–566.

[LBZ19] Ruishan Liu, Akshay Balsubramani, and James Zou. "Learning transport cost from subset correspondence". In: *arXiv preprint arXiv:1909.13203* (2019).

[Let+23] Thibaud Leteno, Antoine Gourru, Charlotte Laclau, Rémi Emonet, and Christophe Gravier. "Fair Text Classification with Wasserstein Independence". In: *arXiv preprint arXiv:2311.12689* (2023).

[LHR23] Thomas Le Barbanchon, Lena Hensvik, and Roland Rathelot. "How can AI improve search and matching? Evidence from 59 million personalized job recommendations". Working paper. 2023.

[Li+19] Ruilin Li, Xiaojing Ye, Haomin Zhou, and Hongyuan Zha. "Learning to Match via Inverse Optimal Transport." In: *Journal of Machine Learning Research* 20.80 (2019), pp. 1–37.

[Li+21] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. "Towards Personalized Fairness based on Causal Notion". In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2021. DOI: 10.1145/3404835.3462966.

[Li+22] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, Juntao Tan, Shuchang Liu, and Yongfeng Zhang. "Fairness in Recommendation: A Survey". In: *arXiv preprint arXiv:2205.13619* (2022).

[Li+23] Yunqi Li, Michiharu Yamashita, Hanxiong Chen, Dongwon Lee, and Yongfeng Zhang. "Fairness in Job Recommendation under Quantity Constraints". In: *AAAI-23 Workshop on AI for Web Advertising*. 2023.

[Lia+14] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. "GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation". In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2014, pp. 831–840.

[Liu09] Tie-Yan Liu. "Learning to Rank for Information Retrieval". In: *Foundations and Trends® in Information Retrieval* 3.3 (2009), pp. 225–331.

[LRB20] Danielle Li, Lindsey R Raymond, and Peter Bergman. *Hiring as exploration*. Tech. rep. National Bureau of Economic Research, 2020.

[LRR21] Thomas Le Barbanchon, Roland Rathelot, and Alexandra Roulet. "Gender Differences in Job Search: Trading off Commute against Wage". In: *The Quarterly Journal of Economics* 136.1 (2021), pp. 381–426.

[LTL23] Tobias Lehmann, Camille Terrier, and Rafael Lalive. "Costs and benefits of congestion in two-sided markets: Evidence from the dating market". Working Paper. 2023.

[Lu18] Sibo Lu. "Search and Signaling on an Online Labor Market". PhD thesis. UC Berkeley, 2018.

[Ma+22] Shichuan Ma, Haiyan Luo, Jianjie Ma, Ziying Liu, Yu Sun, Xingang Huang, Fengdan Wan, Veeresh Beeram, Henry Oh, Santosh R Kumar, and Sreenivasa R Ambati. "Jobs Filter to Improve the Job Seeker Experience at Indeed. com". In: *The 15th ACM International Conference on Web Search and Data Mining (WSDM 2022). First International Workshop on Computational Jobs Marketplace*. 2022.

[Man+13] Faraz M Manshadi, Baruch Awerbuch, Rainer Gemulla, Rohit Khandekar, Julián Mestre, and Mauro Sozio. "A Distributed Algorithm for Large-Scale Generalized Matching". In: *Proceedings of the VLDB Endowment* 6.9 (2013), pp. 613–624. DOI: 10.14778/2536360.2536362.

[Mas+22] Yoosof Mashayekhi, Nan Li, Bo Kang, Jefrey Lijffijt, and Tijl De Bie. "A challenge-based survey of e-recruitment recommendation systems". In: *arXiv preprint arXiv:2209.05112* (2022).

[Mas+23] Yoosof Mashayekhi, Bo Kang, Jefrey Lijffijt, and Tijl De Bie. "ReCon: Reducing Congestion in Job Recommendation using Optimal Transport". In: *Proceedings of the 17th ACM Conference on Recommender Systems*. 2023, pp. 696–701.

[Mas+24] Yoosof Mashayekhi, Bo Kang, Jefrey Lijffijt, and Tijl De Bie. "Scalable Job Recommendation with Lower Congestion using Optimal Transport". In: *IEEE Access* (2024).

[Mat+18] Yusuke Matsui, Yusuke Uchida, Hervé Jégou, and Shin'ichi Satoh. "A Survey of Product Quantization". In: *ITE Transactions on Media Technology and Applications* 6.1 (2018), pp. 2–10.

[Meh+17] Rishabh Mehrotra, Ashton Anderson, Fernando Diaz, Amit Sharma, Hanna Wallach, and Emine Yilmaz. "Auditing Search Engines for Differential Satisfaction Across Demographics". In: *Proceedings of the 26th International Conference on World Wide Web Companion - WWW '17 Companion*. ACM Press, 2017. DOI: 10.1145/3041021.3054197.

[Mel+21] Alessandro B Melchiorre, Navid Rekabsaz, Emilia Parada-Cabaleiro, Stefan Brandl, Oleg Lesota, and Markus Schedl. "Investigating gender fairness of recommendation algorithms in the music domain". In: *Information Processing & Management* 58.5 (2021), p. 102666.

[ML10] Brian McFee and Gert Lanckriet. *Metric Learning to Rank*. Tech. rep. UC San Diego, 2010.

[Mor82] Dale T Mortensen. "The Matching Process as a Noncooperative Bargaining Game". In: *The economics of information and uncertainty*. University of Chicago Press, 1982, pp. 233–258.

[MP17] Alexandre Mas and Amanda Pallais. "Valuing Alternative Work Arrangements". In: *American Economic Review* 107.12 (2017), pp. 3722–59.

[MS23] Andreas I Mueller and Johannes Spinnewijn. "Expectations data, labor market, and job search". In: *Handbook of Economic Expectations* (2023), pp. 677–713.

[MST21] Andreas I Mueller, Johannes Spinnewijn, and Giorgio Topa. "Job Seekers' Perceptions and Employment Prospects: Heterogeneity, Duration Dependence, and Bias". In: *American Economic Review* 111.1 (2021), pp. 324–363.

[Oax73] Ronald Oaxaca. "Male-Female Wage Differentials in Urban Labor Markets". In: *International Economic Review* (1973), pp. 693–709.

[Ozc+19] Cagri Ozcaglar, Sahin Geyik, Brian Schmitz, Prakhar Sharma, Alex Shelkovnykov, Yiming Ma, and Erik Buchanan. "Entity Personalized Talent Search Models with Tree Interaction Features". In: *The World Wide Web Conference*. 2019, pp. 3116–3122.

[Pal+21] Iván Palomares, Carlos Porcel, Luiz Pizzato, Ido Guy, and Enrique Herrera-Viedma. "Reciprocal Recommender Systems: Analysis of state-of-art literature, challenges and opportunities towards social recommendation". In: *Information Fusion* 69 (2021), pp. 103–127.

[PC19] Gabriel Peyré and Marco Cuturi. "Computational Optimal Transport". In: *Foundations and Trends® in Machine Learning* 11.5-6 (2019), pp. 355–607.

[Pis85] Christopher A Pissarides. "Short-Run Equilibrium Dynamics of Unemployment, Vacancies, and Real wages". In: *The American Economic Review* 75.4 (1985), pp. 676–690.

[Ren+12] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. "BPR: Bayesian Personalized Ranking from Implicit Feedback". In: *arXiv preprint arXiv:1205.2618* (2012).

[Ren+20] Steffen Rendle, Walid Krichene, Li Zhang, and John Anderson. "Neural Collaborative Filtering vs. Matrix Factorization Revisited". In: *Proceedings of the 14th ACM Conference on Recommender Systems*. 2020, pp. 240–248.

[Rob77] Stephen E Robertson. "The Probability Ranking Principle in IR". In: *Journal of documentation* 33.4 (1977), pp. 294–304.

[Rou24] Nina Roussille. "The Role of the Ask Gap in Gender Pay Inequality". In: *The Quarterly Journal of Economics* 139.3 (Feb. 2024), pp. 1557–1610.

[RRZ94] James M Robins, Andrea Rotnitzky, and Lue Ping Zhao. "Estimation of regression coefficients when some regressors are not always observed". In: *Journal of the American statistical Association* 89.427 (1994), pp. 846–866.

[Rus+22] Clara Rus, Jeffrey Luppes, Harrie Oosterhuis, and Gido H Schoenmacker. "Closing the Gender Wage Gap: Adversarial Fairness in Job Recommendation". In: *arXiv preprint arXiv:2209.09592* (2022).

[Sar+00] Badrul Sarwar, George Karypis, Joseph Konstan, and John T Riedl. *Application of Dimensionality Reduction in Recommender System-A Case Study*. Tech. rep. University of Minnesota, 2000.

[Sch+17] Thomas Schmitt, François Gonard, Philippe Caillou, and Michele Sebag. "Language Modelling for Collaborative Filtering: Application to Job Applicant Matching". In: *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE. 2017, pp. 1226–1233.

[Sch18] Thomas Schmitt. "Appariements collaboratifs des offres et demandes d'emploi". PhD thesis. Université Paris-Saclay (ComUE), 2018.

[Sch19] Bernhard Schmitzer. "Stabilized Sparse Scaling Algorithms for Entropy Regularized Transport Problems". In: *arXiv preprint arXiv:1610.06519* (2019).

[Shi+22] Jun Shi, Chengming Jiang, Aman Gupta, Mingzhou Zhou, Yunbo Ouyang, Qiang Charles Xiao, Qingquan Song, Yi Wu, Haichao Wei, and Huiji Gao. "Generalized Deep Mixed Models". In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2022, pp. 3869–3877.

[SJ18] Ashudeep Singh and Thorsten Joachims. "Fairness of Exposure in Rankings". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '18. London, United Kingdom: Association for Computing Machinery, 2018, pp. 2219–2228. ISBN: 9781450355520. DOI: 10.1145/3219819.3220088.

[SJ19] Ashudeep Singh and Thorsten Joachims. "Policy Learning for Fairness in Ranking". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc., 2019.

[SW21] Anthony Strittmatter and Conny Wunsch. "The Gender Pay Gap Revisited with Big Data: Do Methodological Choices Matter?" In: *arXiv preprint arXiv:2102.09207* (2021).

[Tom+23] Yoji Tomita, Riku Togashi, Yuriko Hashizume, and Naoto Ohsaka. "Fast and Examination-Agnostic Reciprocal Recommendation in Matching Markets". In: *Proceedings of the 17th ACM Conference on Recommender Systems*. RecSys '23. Singapore, Singapore: Association for Computing Machinery, 2023, pp. 12–23. DOI: 10.1145/3604915.3608774.

[TS16] Gerhard Tutz and Matthias Schmid. *Modeling Discrete Time-to-Event Data*. Springer, 2016.

[Var74] Hal Varian. "Efficiency, Equity and Envy". In: *Journal of Economic Theory* 9.1 (1974), pp. 63–91.

[VYP17a] Maksims Volkovs, Guang Wei Yu, and Tomi Poutanen. "Content-based Neighbor Models for Cold Start in Recommender Systems". In: *Proceedings of the Recommender Systems Challenge 2017 - RecSys Challenge 17*. ACM Press, 2017. DOI: 10.1145/3124791.3124792.

[VYP17b] Maksims Volkovs, Guangwei Yu, and Tomi Poutanen. "DropoutNet: Addressing Cold Start in Recommender Systems". In: *Advances in Neural Information Processing Systems* 30 (2017).

[Wan+22] Yifan Wang, Weizhi Ma, Min Zhang, Yiqun Liu, and Shaoping Ma. "A Survey on the Fairness of Recommender Systems". In: *Journal of the ACM (JACM)* (2022).

[WS09] Kilian Q Weinberger and Lawrence K Saul. "Distance Metric Learning for Large Margin Nearest Neighbor Classification." In: *Journal of Machine Learning Research* 10.2 (2009).

[WVP18] Christina Wadsworth, Francesca Vera, and Chris Piech. "Achieving Fairness through Adversarial Learning: an Application to Recidivism Prediction". In: *arXiv preprint arXiv:1807.00199* (2018).

[Xia+16] Wenming Xiao, Xiao Xu, Kang Liang, Junkang Mao, and Jun Wang. "Job recommendation with Hawkes process: an effective solution for RecSys Challenge 2016". In: *Proceedings of the Recommender Systems Challenge*. RecSys Challenge '16. Boston, Massachusetts, USA: Association for Computing Machinery, 2016. ISBN: 9781450348010. DOI: 10.1145/2987538.2987543.

[Xia+19] Bin Xia, Junjie Yin, Jian Xu, and Yun Li. "WE-Rec: A fairness-aware reciprocal recommendation based on Walrasian equilibrium". In: *Knowledge-Based Systems* 182 (2019), p. 104857.

[YAÖ21] Ezgi Yıldırım, Payam Azad, and Şule G Öğüdücü. "biDeepFM: A multi-objective deep factorization machine for reciprocal recommendation". In: *Engineering Science and Technology, an International Journal* 24.6 (2021), pp. 1467–1477.

[YH17] Sirui Yao and Bert Huang. "Beyond Parity: Fairness Objectives for Collaborative Filtering". In: *Advances in Neural Information Processing Systems* 30 (2017).

[Zaf+17] Muhammad Bilal Zafar, Isabel Valera, Manuel Rodriguez, Krishna Gummadi, and Adrian Weller. "From Parity to Preference-based Notions of Fairness in Classification". In: *Advances in Neural Information Processing Systems* 30 (2017).

[ZH20] Simon Zhuang and Dylan Hadfield-Menell. "Consequences of Misaligned AI". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 15763–15773.

[Zha+16] Xianxing Zhang, Yitong Zhou, Yiming Ma, Bee-Chung Chen, Liang Zhang, and Deepak Agarwal. "GLMix: Generalized Linear Mixed Models for Large-Scale Response Prediction". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016, pp. 363–372.

[Zha+19] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. "Deep Learning based Recommender System: A Survey and New Perspectives". In: *ACM computing surveys (CSUR)* 52.1 (2019), pp. 1–38.

[Zha+21] Jing Zhao, Jingya Wang, Madhav Sigdel, Bopeng Zhang, Phuong Hoang, Mengshu Liu, and Mohammed Korayem. "Embedding-based Recommender System for Job to Candidate Matching on Scale". In: *arXiv preprint arXiv:2107.00221* (2021).

[Zha21] Shuo Zhang. "Understanding Algorithmic Bias in Job Recommender Systems: An Audit Study Approach". Working Paper. 2021.

Appendices

Appendix A

MUSE: hyper-parameters and configuration

This Appendix describes the hyper-parameters used in the experiments presented in Chapter 3.

MUSE First, let us detail the MUSE.0, MUSE.1 and MUSE.2 architecture dimensions and hyper-parameters. The dimensions of the inputs provided to the different MUSE modules are described in Table A.1. The neural architecture and hyper-parameters of MUSE are displayed in Table A.2.

Table A.1: France Travail dataset input dimensions

	Muse.0			Muse.1 / Muse.2
Input dim.	General	Geo	Skills	
Job seekers	483	573	12.3k	483($\mathbf{x}.Gal$)+771(ϕ_0)
Job ads	469	571	12.3k	469($\mathbf{y}.Gal$)+771(ψ_0)+ 13($Var(\mathbf{x}, \mathbf{y})$)+2($r_0$,$M_0$)

DropoutNet DropoutNet's hyper-parameters on the Xing dataset are left as reported in the authors' code.

XGB On the *France Travail* dataset, XGB is used with the hyper-parameters reported in Table A.3. Other hyper-parameters are set to their default value. Negative examples are sampled uniformly at random.

Table A.2: Muse hyper-parameters: Neural architecture of Muse.0, Muse.1 and Muse.2 on the RecSys and France Travail datasets

Recsys Dataset	Muse.0			Muse.1/Muse.2
Input dim	Job ad:2738 / Job seeker: 831			
Hidden Layer Size	800-800 →400			ϕ_1, ψ_1: 200 MLP: 200
Muse.2 Layer				2→100→100
Batch size	128			128
Learning rate	10^{-4}			10^{-3}
Negative sampling	Uniform			Top-1000
France Travail dataset	**Muse.0**			**Muse.1 / Muse.2**
	Gal	Geo	Skills	
Hidden Layer Size	500 →100	573 → 571	200 →100	ϕ_1, ψ_1: 200 MLP: 200
Muse.2 Layer				2→100→100
Batch size	256	32	32	128
Learning rate	10^{-3}	10^{-3}	10^{-3}	10^{-4}
Negative sampling	Uniform	d(x,neg)> d(x,pos)	Uniform	Top-1000

col_sample_bytree	0.6
eta	0.075
gamma	0.85
max_depth	12
min_child_weight	1
subsample	0.9
num_boost_round	400
Loss	Logistic
Negative sampling ratio	5

Table A.3: Xgb Hyper-parameters on France Travail dataset

Appendix B

MUSE: recall heterogeneity analysis

This Appendix provides a decomposition of MUSE.2's recall@10 (trained in the experimental settings described in Chapter 3), on the test set (44 277 matches), by characteristics of the job seekers and job seeker - job ad pairs. The R@10 overall is 30.4.

Based on Figure B.1, MUSE.2's recall@10 tends to increase with job seeker's age category. Based on Figure B.2, recall is highest for job seekers looking for blue-collar positions, and lower for job seekers looking for executive jobs (or with missing qualification level). Figure B.3 presents recall according to job seekers' level of education: recall is lowest for those with five or more years spent in higher education. When looking at the recall based on the characteristics of the job seeker - job ad pair (Figures B.4 and B.5), MUSE.2 achieves higher recall when job seekers find jobs close to their geographic location, and in the exact occupation they are looking for.

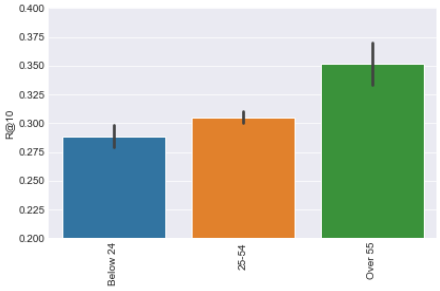


Figure B.1: Recall@10 by job seeker age

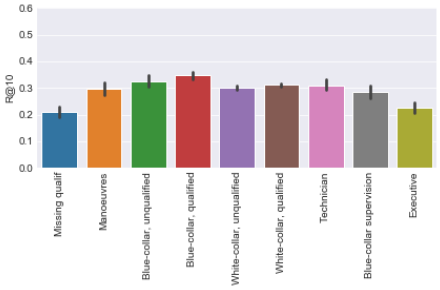


Figure B.2: Recall@10 by job seeker qualification

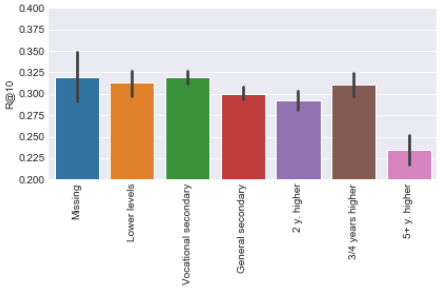


Figure B.3: Recall@10 by job seeker education

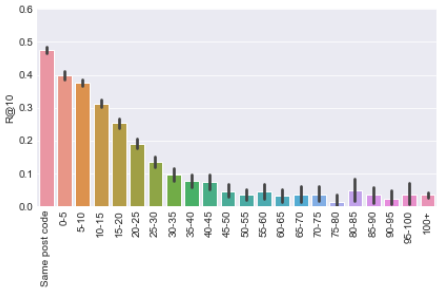


Figure B.4: Recall@10 by distance to workplace

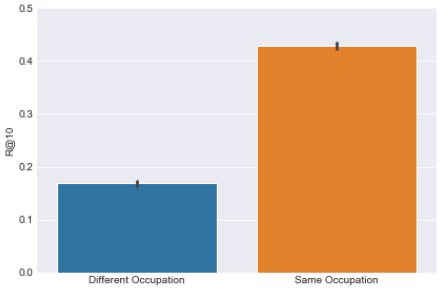


Figure B.5: Recall@10 according whether job seeker and job ad are in the same occupation

Appendix C

Complements on value alignment

C.1 Algorithm details

Table C.1: Information on ads and job seekers respectively used by $\mathcal{U}$ and $\mathcal{P}$

Preference-based		Machine learning (Muse.0)	
Job seekers	Offers	Job seekers	Offers
Skills	Skills	Skills (SVD, embedding)	Skill (SVD, embedding)
Diploma	Diploma	Diploma	Diploma
Languages	Languages		
Driver's licence	Driver's licence	Driver's licence	Driver's licence
Experience	Experience	Experience	Experience
Occupation (lv. 3)	Occupation (lv. 3)	Occupation (lv. 1, 2, 3)	Occupation (lv. 1, 2, 3)
Working hours	Working hours	Working hours	Working hours
Wage	Wage	Wage (several measures)	Wage (upper, lower bounds)
Location	Location	Location	Location
Geo. mobility		Geo. mobility	
Contrat type	Contract type	Contract type	Contract type
		Qualification	Qualification
		Soft skills	Soft skills
			Job description (text)
			Firm description (text)
			Contract type
			Contract duration
			Establishment size
			Establishment status
			Num. applications (ad)
			Num. applications (establishment)
			Num. days since posted
			Geo. soc.-dem. features
		Former occupation	
		Gender	
		Num. children	
		Search obligations	
		Job search type	
		Min. allowance status	
		Days unemployed	
		Age	
		Num. applications	
		Geo. soc.-dem. features	

C.2 Calibration of the ML score into a hiring probability

One might want to characterize the true probability of i being matched with j conditional on the available information X_i, Y_j, namely $\mathbb{P}\left(M^*_{i,j} = 1 | X_i, Y_j\right)$. There are two main difficulties arising in the estimation of this true conditional probability. First, as we want to consider all the potentially relevant covariates at our disposal, this is an high dimensional problem. Second, there is a selection issue as we only observe matches conditional on an past interviews $A_{i,j} = 1$, thus the variable $M_{i,j} = M^*_{i,j} A_{i,j}$. In this section, we provide a framework which allows to calibrate the ML score $S_{i,j} := S(X_i, Y_j)$ produced by the MUSE.0 and learned on past matches to learn about the true conditional hiring probability.

We want to give an interpretation to the ML score in terms of hiring probability and to characterize the information contained in $S_{i,j}$ about the probability of a match. Thus, our aim is to identify the probability of a match conditional on the score for a randomly chosen pair (i, j) that would be put artificially in contact with a given score, namely

$$p(i,j) = \mathbb{P}\left(M^*_{i,j} = 1 | S_{i,j} = s_{i,j}\right).$$

We leverage all inter-mediation acts (see Chapter 2) operated by *France Travail* between job seekers and job ads, at the initiative of the job seeker, the recruiter, or the caseworker, and (with the limitations noted in Chapter 2) observe the dates of the contact as well as whether it resulted in a hire. We want to identify $P(M^*_{i,j} = 1 | S_{i,j})$, but show in the next paragraph that our data only allows identifying $P(M_{i,j} = 1 | S_{i,j}, A_{i,j} = 1)$. To handle this selection issue and relate $p(i,j)$ to the true hiring probability conditional on $X_{i,j}$, we make two assumptions:

- **Assumption 1**. Selection on observables:

$$M^*_{i,j} \perp\!\!\!\perp A_{i,j} \mid X_i, Y_j.$$

- **Assumption 2**. $S_{i,j}$, which is only a function of X_i and Y_j, is a sufficient statistic, *i.e.*,

$$\mathbb{P}(M^*_{ij,} = 1 | S_{i,j}, X_i, Y_j) = \mathbb{P}(M^*_{ij,} = 1 | S_{i,j})$$

Under these two assumptions, $\mathbb{P}(M_{i,j} = 1 | S_{i,j}, A_{i,j} = 1) = \mathbb{P}(M^*_{ij,} = 1 | S_{i,j})$. Thus, we now describe the procedure we follow to identify $\mathbb{P}(M_{i,j} = 1 | S_{i,j}, A_{i,j} = 1)$.

Identification and estimation of the conditional hiring probability. We consider the model described in Section 4.1, where observations follow (4.1). Note that, in this model the probability that a job seeker matches on the n-th offer of the sequence S is given by

$$\begin{aligned}
&\mathbb{P}(M_{i,1(i)} = 0, \ldots, M_{i,n-1(i)} = 0, M_{i,n(i)} = 1 | \mathcal{J}(i), S) \\
&= \Lambda(\alpha_n + \beta S_{i,n(i)}) \prod_{j=1}^{n(i)-1} (1 - \Lambda(\alpha_{r(i,j(i))} + \beta S_{i,j(i)})),
\end{aligned}$$

where $n(i)$ is the number of observed application for job seeker i.

Taking into account completed and censored spells (see, *e.g.* p.53 in [TS16]), the log-likelihood function, conditional on the scores produced by the recommender system, is given by

$$\begin{aligned}\mathcal{L}(\alpha,\beta|M,S) = &\sum_{i=1}^{N} M_{i,n(i)} \ln(\Lambda(\alpha_{r(i,n(i))} + \beta S_{i,n(i)})) \\ &+ \sum_{i=1}^{N} \sum_{j\in\mathcal{J}(i)\setminus\{n(i)\}} (1-M_{i,j}) \ln(1-\Lambda(\alpha_{r(i,j} + \beta S_{i,j})),\end{aligned}$$

where N the number of observed job seekers.

Note that, if we omit the rank of vacancy j, this expression is symmetric in i and j. Thus, we could see as well this expression as the result of a process in which firm posting vacancy j sequentially considers candidates applying to the vacancy. This simple remark shows that there is a simple generalisation of the former expression to account for $r(i,j)$ the rank of offer j in the application set of job seeker i, but also $q(i,j)$ the rank of i in the applicant pool for vacancy j. The expression of the likelihood in this case writes as

$$\begin{aligned}\mathcal{L}(\alpha,\beta|M,S) = &\sum_{(i,j):\ A_{i,j}=1} M_{i,j} \ln(\Lambda(\alpha^{v}_{q(i,j)} + \alpha^{js}_{r(i,j)} + \beta S_{i,j}) \\ &+ \sum_{(i,j):\ A_{i,j}=1} (1-M_{i,j}) \ln(1-\Lambda(\alpha^{v}_{q(i,j)} + \alpha^{js}_{r(i,j)} + \beta S_{i,j})),\end{aligned}$$

where α^{v} and α^{js} are the sequences of "weariness" effects for vacancies and job seekers.

The calibration of the ML score is performed on 34,255 randomly selected job seekers in the test set representing 84,538 applications.

C.3 Complements on the model

We denote by Λ the cumulative logistic distribution function. To be able to derive hiring probabilities conditional on the observed types, we make the following standard assumption. They are sufficient to obtain the results mentioned in Section 4.3.

Assumption 1 *For all $i \in I$, $j \in J$,*

1. *The errors $\delta_{i,y}$ are i.i.d distributed according to a logistic distribution, and independent of $\varepsilon_{i,y}$ and $\eta_{x,j}$;*

2. *The errors $\delta_{i,y}$ for $y \in \mathcal{Y}$, do not enter the subjective hiring probability conditional on applying on an offer of type y: $\pi(M^*_{i,j} = 1|\mathcal{I}_1) = \pi(M^*_{i,j} = 1|x, y, A_{i,j} = 1) = \pi(M^*_{i,j} = 1|x, y, y \in \mathcal{C}_i)$.*

We assume that the market is large, which implies that firms can always find any type of worker. In general, the consideration set for the firms, denoted by $\mathcal{X}(y, F_{\delta,\pi})$, depends only on y and the distributions of $\{\delta_{i,y}\}_{y\in\mathcal{Y}}$ and $\{\pi(M^*_{i,j} = 1|\mathcal{I}_1)\}_{y\in\mathcal{Y}}$ among the population of workers. As the distribution of $(c + r)/\pi(M^*_{i,j} = 1|\mathcal{I}_1) - \delta_{i,y}$ has full support for all types, then with nonzero probability all firms receive application from all types of workers $\mathcal{X}(y, F_{\delta,\pi}) = \mathcal{X}$.

Assumption 1.1 means that $\delta_{i,y}$ does not contain useful and additional to x, y information for predicting the utility before the interview.[1] This can be thought as unobservable quantities such as mood, weather, etc, that would impact the subjective beliefs but not the final utility the job seekers would get from a match. Assumption 1.2 first restricts the use of the private information $\{\delta_{i,y_j}\}$, which is not used by job seekers, to predict their chances of success.

Assumption 2 *For all $i \in \mathcal{I}$, $j \in \mathcal{J}$, $\varepsilon_{i,\cdot}$ and $\eta_{\cdot,j}$ are i.i.d of standard type I extreme value (Gumbel).*

Under this assumption it is possible to derive the hiring probability resulting from the job seeker's application and the selection on the employer side. Lemma C.3.1 in Appendix C.3 gives the exact form of this hiring probability. This result is more complex but close to the one of [CS06] and [GS21]. In their model, there is no stage 1: all job seekers and firms in a market meet each other $A_{i,j} = 1\ \forall i, j$, hence $M^*_{i,j} = M_{i,j}$. They derive the following expression for the hiring probability:

$$P\left(M^*_{i,j} = 1\middle|x, y, A_{i,j} = 1\right) = \underbrace{p_{f,0}(y)e^{V(x,y)}}_{\text{Probability } y \text{ selects } x} \underbrace{p_{js,0}(x)e^{U(x,y)}}_{\text{Probability } x \text{ selects } y}, \tag{C.1}$$

[1] When relaxing this Assumption 1.1, assuming that $\delta_{i,y}$ is correlated with $\varepsilon_{i,y}$, then the job seekers would have relevant private information about their second stage utilities. This can be handled by conditioning on this information to derive expressions of the hiring probabilities, but this complicates the discussion and the final formulae, without providing further insights for the role of recommender systems in this context.

where $p_{f,0}(y) = 1/(1 + \sum_{x' \in \mathcal{X}} e^{V(x',y)})$ and $p_{js,0}(x) = 1/(1 + \sum_{y' \in \mathcal{Y}} e^{U(x,y')})$ are the probabilities that a firm of type y and a job seeker of type x prefer to remain unmatched. A notable difference, due to the selection step, between the expression of the equation (4.8) and the one we derive in the Lemma C.3.1 concerns the probability $p_{j,0}(x)$. In our model, job seekers' expectations in the preliminary selection stage distorts $p_{j,0}(x)$ as it integrates over the possible consideration sets.

In the following lemma, we obtain a similar result, the only change being related to the probability that x selects y, which takes into account all the possible consideration sets.

Lemma C.3.1 (Observed hiring probabilities with a selection stage) *Under assumptions 1 and 2 and at equilibrium, the probability that we observe a hire for a worker i and firm j of types $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ is given by*

$$P\left(M_{i,j} = 1 | x, y\right) = P\left(M^*_{i,j} = 1 \middle| x, y, A_{i,j} = 1\right) P(A_{i,j} = 1 | x, y),$$

where the observed probability of applying for a job is

$$P(A_{i,j} = 1 | x, y) = P(y \in \mathcal{C}_i | x, y) = \Lambda(\Psi(x, y)),$$

where

$$\Psi(x, y) = U(x, y) - U_{x,0} + r - \frac{c + r}{\pi(M^*_{i,j} = 1 | x, y, A_{i,j} = 1)},$$

and the probability that we observe a hire conditionally on the application is

$$P\left(M^*_{i,j} = 1 \middle| x, y, A_{i,j} = 1\right) = p_{f,0}(y) p_{js,0}(x | y \in \mathcal{C}_i) e^{U(x,y) + V(x,y)}, \tag{C.2}$$

where

$$p_{js,0}(x | y \in \mathcal{C}_i) = \sum_{\substack{S \subseteq \mathcal{Y} \\ y \in S}} p_{js,0}(x | S) P(\mathcal{C}_i = S | x, y, y \in \mathcal{C}_i), \tag{C.3}$$

$$p_{js,0}(x | S) = 1/(1 + \sum_{y' \in S} e^{U(x,y') + w_{x,y'}}),$$

$$P(\mathcal{C}_i = S | x, y, y \in \mathcal{C}_i) = \prod_{y' \in S, y' \neq y} \Lambda\left(\Psi(x, y')\right) \prod_{y' \notin S} \left(1 - \Lambda\left(\Psi(x, y')\right)\right). \tag{C.4}$$

The main difference between equations (4.8) and (C.2) is the expression of the probability that a job seeker of type x prefers staying unmatched. In equation (4.8) it is $p_{j,0}(x|\mathcal{Y}) = p_{j,0}(x)$, as he is facing all possible alternatives in $\mathcal{Y}$ rather than a selected set. In Lemma C.3.1, we express the observed hiring probabilities from the point of view of a researcher who does not observe to the consideration set $\mathcal{C}_i$. Thus, in the formula (C.3) of $p_{j,0}(x|y \in \mathcal{C}_i)$, the probability that a job seeker of type x prefers staying unmatched after the interview conditionally on having applied to a type y offer, we integrate over all the possible values $S \subseteq \mathcal{Y}$ for the consideration set. In this context, the probability that job seeker i considers the set of offer S, given in (C.4), is the product of the probabilities that he selects all the offers in S and that he does not apply for the rest, which are all results of unrelated binary choices (4.6).

Proof of Lemma C.3.1. Let $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, then based on (4.6) and with Assumption 2, we have (C.3). We also have

$$P(M_{i,j} = 1|x, y) = P(M^*_{i,j} = 1|x, y, y \in \mathcal{C}_i)P(A_{i,j} = 1|x, y),$$

and $P(M^*_{i,j} = 1|x, y, y \in \mathcal{C}_i) = P(i \text{ chosen by } j|x, y, y \in \mathcal{C}_i)P(j \text{ chosen by } i|x, y, y \in \mathcal{C}_i)$. First, because $\mathcal{C}_i$ is unobserved, we integrate over it:

$$P(i \text{ chosen by } j|x, y, y \in \mathcal{C}_i) = \sum_{\substack{S \subseteq \mathcal{Y} \\ y \in S}} P(\mathcal{C}_i = S|x, y, y \in \mathcal{C}_i) \frac{e^{U(x,y)}}{1 + \sum_{y' \in S} e^{U(x,y')+w_{x,y'}}}.$$

Second, using that $\mathcal{X}(y, F_{\delta,\pi}) = \mathcal{X}$, we obtain

$$P(j \text{ chosen by } i|x, y, y \in \mathcal{C}_i) = p_{f,0}(y)e^{V(x,y)-w_{x,y}}.$$

This yields the result. □

C.4 Robustness check - modelling applications without fixed effects

Table C.2: Estimates of the model of application on job offers without fixed effects

	Estimate	Std. error	Estimate	Std. error	Estimate	Std. error	Estimate	Std. error
Utility score $\mathcal{U}(i,j)$ (α)	1.375**	0.180						
Unif. Utility score $\overline{U}(i,j)$ (α)			0.914**	0.204				
Sector occupation					-0.277	0.184		
Occupation					1.088**	0.221	0.801**	0.064
Skills					0.198*	0.095	0.198*	0.087
Reservation wage					0.432**	0.085	0.437**	0.055
Languages					0.028	0.194	0.027	0.177
Experience in occ.					-1.188**	0.340	-1.180**	0.340
Diploma					0.392	0.108**	0.402**	0.106
Driving license					0.023	0.100	0.025	0.101
Geographic mobility					0.414†	0.232	0.417†	0.232
Duration					0.141	0.114	0.139	0.114
Type of contract					0.078	0.053	0.076	0.053
Inverse of $\mathcal{P}(i,j)$ (β)	-0.019**	0.007	-0.022**	0.007	-0.016*	0.007	-0.016*	0.007
Intercept	-3.375**	0.162	-3.314**	0.169	-3.287**	0.169	-3.287**	0.168
Nb. observations	70,557		70,557		70,557		70,557	
AIC	18,230		18,300		18,100		18,100	

Estimation of equation (4.7) modeling applications as a logit model without fixed effects

Notes: Our sample is the set of all applications for job seekers in the transportation and logistic sector during week 44 of 2019, leading to a hiring or not. We have 70,557 observations for 8,105 job seekers. Standard errors are clustered at the individual level. Results are robust to the different negative sampling strategies we considered. Significance levels: $< 1\% : {}^{**}$, $< 5\% : {}^{*}$, $< 10\% : {}^{\dagger}$.

Appendix D

Complements on field experiments

D.1 March 2022 field experiment

This appendix provides details on algorithms used for the March 2022 field experiment, as well as supplementary material for its analysis.

D.1.1 Algorithms

The Pbs algorithm The Pbs algorithm takes the form:

$$s_{ij} = f_{ij} \times \left(\sum_{k \in \text{Criteria}} w_k s_{ijk} \right)$$

where:

- $f_{ij} \in \{0, 1\}$ is a filter, taking value 1 if the score assigned to geography is different from zero
- The criteria s_{ijk} and their weights w_k are:
    1. Skills: weight 1000
    2. Job type: weight 1500 (500 at the ROME level and 1000 at a finer level of granularity)
    3. Experience: weight 100
    4. Geographic mobility[1], weight 100
    5. Contract type, weight 10
    6. Weekly working hours, weight 100
    7. Education, weight 100
    8. Languages, weight 100

[1]This score is not necessarily based on distance between the ad and the job seekers' place of residence. It takes into account the kind of mobility declared acceptable by the job seeker. Job seekers can declare a zip code and a commuting radius (in which case the score is a decreasing function of distance, taking value 0 after a threshold), but also, less often, a country, region or *département* (in which case the score takes binary values).

9. Driver's licence, weight 100
10. Wage, weight 200

Scores take value between 0 (complete mismatch) and 1 (perfect fit), with intermediate values determined by expert-provided matrices and discontinuities. The final ranking is a lexicographic sort by decreasing s_{ij}, increasing geographic distance (to one's zip code of residence), and decreasing job ad creation date.

The Mix algorithm We first attribute "stars" to job ads with respect to both Muse.0 and Pbs to construct a consideration set of job ads that have a high ranking for one of these algorithms, or good rankings for both. Stars with respect to an algorithm are determined in the following fashion: 4 if the ad's rank is below 10; 3 if the ad's rank is below 25; 2 if the ad's rank is below 50; 1 if the ad's rank is below 100; 0 otherwise.

Mix only takes into consideration job ads for which the sum of Muse.0 and Pbs "stars" are greater or equal to 3. This consideration set takes a size between 25 (the top-25s of Muse.0 and Muse.1 are the same) and 100 (disjoint top-25s, and the job ads ranked 25-50 by an algorithm are among the other's top 50-100).

From this consideration set, Mix aims to generate 15 recommendations per job seeker[2]. Mix-p ($p \in \{1/4, 1/2, 3/4\}$) takes the $\max(15, p\times$ size(consideration set)) first ones according to Muse.0, and reorders them by the Pbs score.

[2]In order to present 10 job ads to each job seeker in the experiment. A larger amount of ads nevertheless has to be ranked in order to anticipate a mismatch between job ads available at recommendation time and those actually online at the time of sending the survey (in order to make sure recommended job ads are actually online).

D.1.2 Survey design

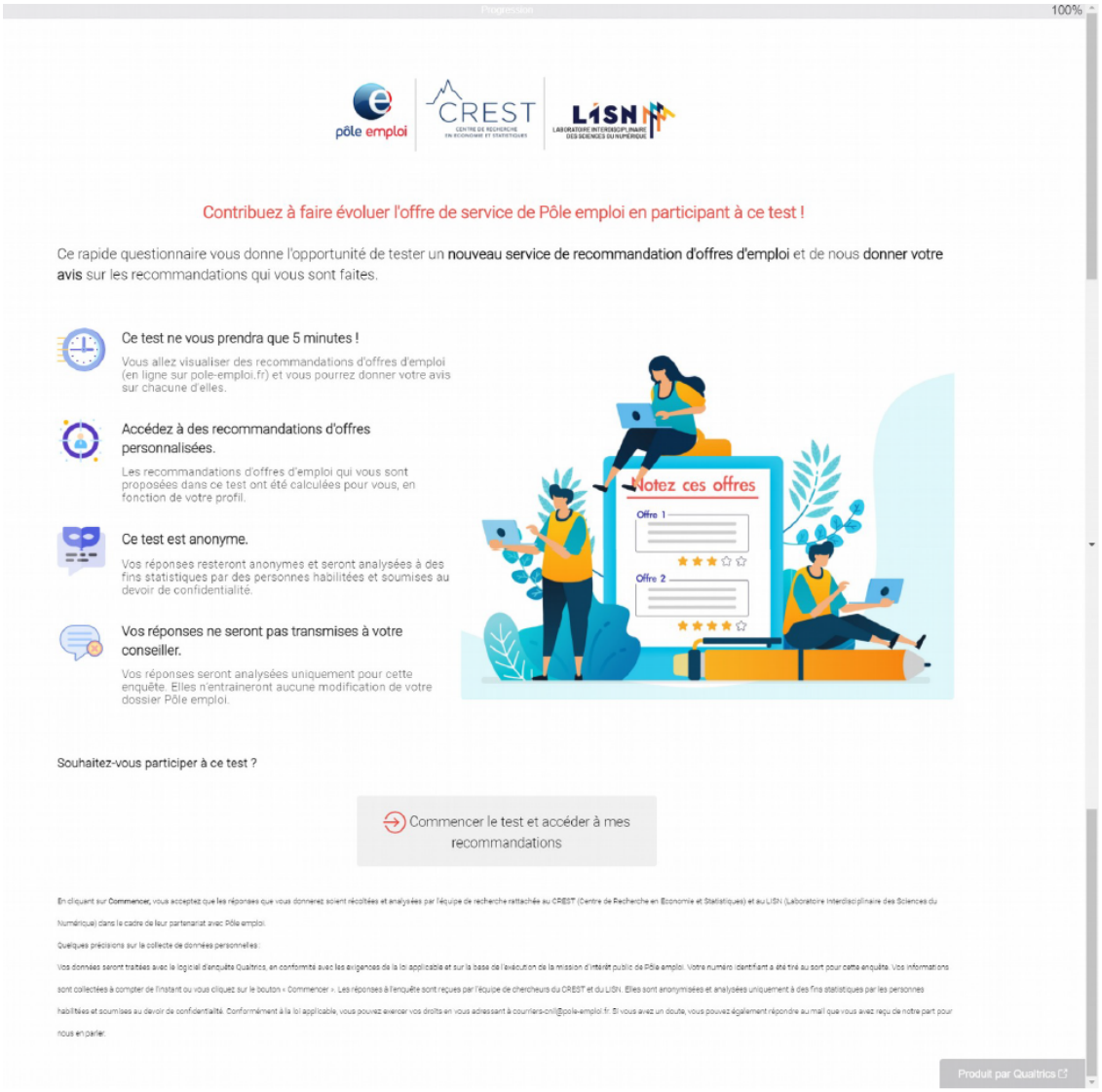


Figure D.1: March 2022 Field Experiment - Landing Page

Figure D.2: March 2022 Field Experiment - First page - Job Ad Description and Rating Scale

Figure D.3: March 2022 Field Experiment - Second Page
Clicking on "Voir le détail de l'offre sur Pôle-emploi.fr" leads to a more thorough description of job ads on *France Travail*'s website, on which job seekers may also apply.

D.1.3 Attrition differential

Table D.1 displays the results of the regression:

$$Y_i = \alpha + \sum_k \beta_k \{T_i = k\} + \epsilon_i$$

among job seekers who received an email, where T_i is job seeker i's received treatment, and Y_i corresponds to a binary indicator of having completed the survey (rated the top two ads and accessed the final page). The PBS treatment serves as the reference category. A F-test of the joint nullity of coefficients associated to MUSE.0, MIX-1/4, MIX-1/2 and MIX-3/4 yields a F-stat 1.885 (p=0.11). Accordingly, we do not attempt to model attrition differential.

	Coefficient	Std. err.	t	P> |t|	[0.025	0.975]
α	0.1718	0.004	45.288	0.000	0.164	0.179
Muse.0	0.0042	0.005	0.784	0.433	-0.006	0.015
Mix-1/4	0.0012	0.005	0.216	0.829	-0.009	0.012
Mix-1/2	0.0133	0.005	2.470	0.014	0.003	0.024
Mix-3/4	0.0055	0.005	1.022	0.307	-0.005	0.016

Table D.1: Survey completion

D.2 June 2023 field experiment

D.2.1 Survey

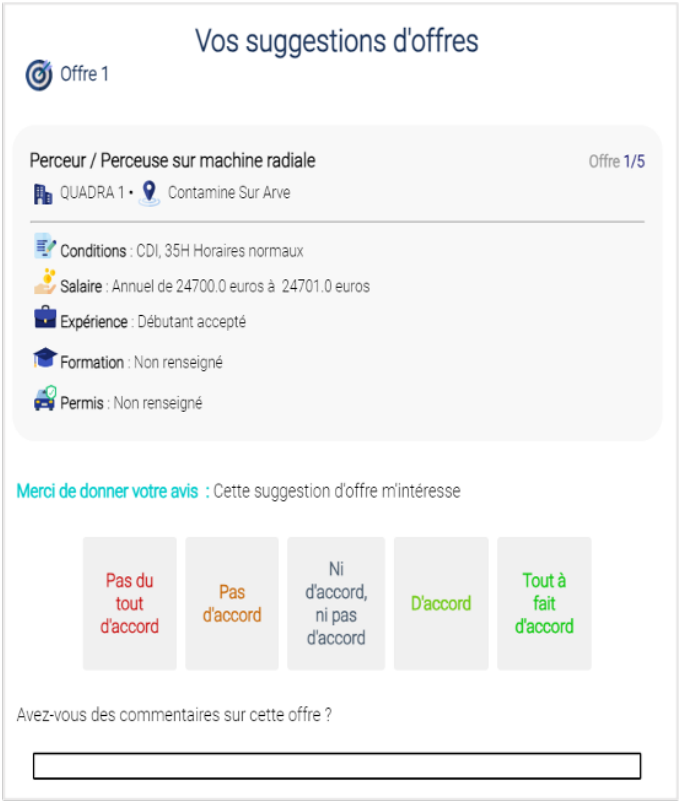


Figure D.4: June 2023 Field Experiment - First Page

Figure D.5: June 2023 Field Experiment - Second Page

Appendix E

Complements on congestion-avoiding job recommendation

E.1 Additional tables

Table E.1 details the full results obtained on the MAR dataset, including cluster-level evaluations.

Table E.2 provides the full results obtained on the *France Travail* dataset.

Table E.3 displays computational costs. It shows a limited training time of respectively XGB (circa 2 hours) and MUSE.0 (circa 30 mn). The cost of optimal transport increases as ε decreases, up to circa 10mn for $\varepsilon = .01$ (see also [Sch19]). The highest part of the cost comes from computing the recommendations with XGB and γ^{XGB}, due to the fact that it requires to compute joint adequacy features for all (user, item) pairs.

Table E.1: Results - MAR Matrimonial dataset

Algorithm	Recall		Coverage		Congestion		Ind-Cluster		Cluster-Cluster	
	@1	@10	@1	@10	@1	@10	RMSE	MAE	RMSE	MAE
ϕ Random	0.16	2.27	63.32	100	-0.90	-0.98	12.68	6.186	nc	nc
ϕ XGBoost	7.93	27.88	48.55	98.69	-0.84	-0.94	12.60	5.619	nc	nc
ϕ NN	3.82	15.5	46.27	98	-0.83	-0.93	12.99	5.905	nc	nc
Carot - XGBoost										
γ^{XGB},$g = exp+$,$\varepsilon = 100.0$	8.01	28.16	48.51	99.14	-0.84	-0.95	12.64	5.629	9.044	5.944
γ^{XGB},$g = exp+$,$\varepsilon = 10.0$	7.97	28.16	48.59	99.14	-0.84	-0.95	12.64	5.629	9.016	5.928
γ^{XGB},$g = exp+$,$\varepsilon = 1.0$	8.09	28.08	49.57	99.22	-0.85	-0.95	12.57	5.616	8.856	5.756
γ^{XGB},$g = exp+$,$\varepsilon = 0.1$	8.14	28.37	73.82	100	-0.93	-0.98	12.06	5.427	16.41	6.376
γ^{XGB},$g = exp+$,$\varepsilon = 0.01$	6.63	26.98	95.44	100	-0.98	-0.95	11.87	5.341	24.30	7.221
γ^{XGB},$g = Id+$,$\varepsilon = 100.0$	8.1	28.41	49.2	99.06	-0.84	-0.95	12.56	5.603	9.044	5.944
γ^{XGB},$g = Id+$,$\varepsilon = 10.0$	8.1	28.41	49.2	99.1	-0.84	-0.95	12.56	5.603	9.022	5.931
γ^{XGB},$g = Id+$,$\varepsilon = 1.0$	8.05	28.41	49.77	99.18	-0.85	-0.95	12.55	5.596	8.887	5.786
γ^{XGB},$g = Id+$,$\varepsilon = 0.1$	8.01	27.02	72.73	100	-0.93	-0.95	12.13	5.440	19.51	6.704
γ^{XGB},$g = Id+$,$\varepsilon = 0.01$	6.47	23.77	96.05	100	-0.98	-0.84	11.99	5.391	24.49	7.257
γ^{XGB},$g = ndcg$,$\varepsilon = 100.0$	7.93	28.2	48.55	98.98	-0.84	-0.95	12.60	5.619	nc	nc
γ^{XGB},$g = ndcg$,$\varepsilon = 10.0$	7.93	28.24	48.55	99.02	-0.84	-0.95	12.60	5.619	nc	nc
γ^{XGB},$g = ndcg$,$\varepsilon = 1.0$	7.93	28.2	48.55	99.02	-0.84	-0.95	12.60	5.619	nc	nc
γ^{XGB},$g = ndcg$,$\varepsilon = 0.1$	8.1	25.72	59.42	100	-0.89	-0.93	12.34	5.512	nc	nc
γ^{XGB},$g = ndcg$,$\varepsilon = 0.01$	6.06	19.49	94.26	100	-0.98	-0.73	12.02	5.433	nc	nc
γ^{XGB},$g = rank-based$,$\varepsilon = 100.0$	7.93	27.63	48.55	98.98	-0.84	-0.95	12.60	5.619	nc	nc
γ^{XGB},$g = rank-based$,$\varepsilon = 10.0$	7.93	27.63	48.55	98.98	-0.84	-0.95	12.60	5.619	nc	nc
γ^{XGB},$g = rank-based$,$\varepsilon = 1.0$	7.93	27.63	48.55	99.02	-0.84	-0.95	12.60	5.619	nc	nc
γ^{XGB},$g = rank-based$,$\varepsilon = 0.1$	7.53	25.76	62.43	99.95	-0.90	-0.93	12.25	5.473	nc	nc
γ^{XGB},$g = rank-based$,$\varepsilon = 0.01$	6.02	21.41	84.08	100	-0.96	-0.79	11.87	5.352	nc	nc
Carot NN										
γ^{NN},$g = exp+$,$\varepsilon = 100.0$	1.5	9.32	13.75	51.4	-0.56	-0.65	16.10	6.824	9.045	5.945
γ^{NN},$g = exp+$,$\varepsilon = 10.0$	1.54	9.48	14.56	53.07	-0.58	-0.66	15.96	6.795	9.030	5.935
γ^{NN},$g = exp+$,$\varepsilon = 1.0$	1.95	11.35	20.38	65.76	-0.65	-0.75	15.17	6.655	8.976	5.839
γ^{NN},$g = exp+$,$\varepsilon = 0.1$	3.74	15.5	54.94	99.06	-0.87	-0.97	12.77	5.896	12.32	5.940
γ^{NN},$g = exp+$,$\varepsilon = 0.01$	3.78	15.67	88.15	100	-0.97	-0.97	12.03	5.543	23.14	7.164
γ^{NN},$g = Id+$,$\varepsilon = 100.0$	2.76	14	35	88.88	-0.77	-0.87	13.68	6.196	9.045	5.944
γ^{NN},$g = Id+$,$\varepsilon = 10.0$	2.72	14	35.4	89.58	-0.78	-0.88	13.64	6.179	9.024	5.931
γ^{NN},$g = Id+$,$\varepsilon = 1.0$	2.84	14.32	38.86	92.47	-0.80	-0.90	13.40	6.085	8.980	5.798
γ^{NN},$g = Id+$,$\varepsilon = 0.1$	3.94	15.46	70.12	100	-0.92	-0.98	12.36	5.668	17.08	6.512
γ^{NN},$g = Id+$,$\varepsilon = 0.01$	3.78	15.46	93.48	100	-0.98	-0.95	12.02	5.576	24.37	7.264
γ^{NN},$g = ndcg$,$\varepsilon = 100.0$	3.82	15.67	46.27	98.53	-0.83	-0.94	12.99	5.905	nc	nc
γ^{NN},$g = ndcg$,$\varepsilon = 10.0$	3.82	15.67	46.27	98.53	-0.83	-0.94	12.99	5.905	nc	nc
γ^{NN},$g = ndcg$,$\varepsilon = 1.0$	3.82	15.63	46.27	98.73	-0.83	-0.94	12.99	5.905	nc	nc
γ^{NN},$g = ndcg$,$\varepsilon = 0.1$	4.23	13.87	57.99	99.91	-0.88	-0.93	12.51	5.716	nc	nc
γ^{NN},$g = ndcg$,$\varepsilon = 0.01$	2.89	11.6	93.44	100	-0.98	-0.72	11.94	5.504	nc	nc
γ^{NN},$g = rank-based$,$\varepsilon = 100.0$	3.82	15.14	46.27	99.26	-0.83	-0.94	12.99	5.905	nc	nc
γ^{NN},$g = rank-based$,$\varepsilon = 10.0$	3.82	15.14	46.27	99.26	-0.83	-0.94	12.99	5.905	nc	nc
γ^{NN},$g = rank-based$,$\varepsilon = 1.0$	3.82	15.18	46.27	99.3	-0.83	-0.94	12.99	5.905	nc	nc
γ^{NN},$g = rank-based$,$\varepsilon = 0.1$	3.7	14.28	65.81	100	-0.91	-0.94	12.29	5.674	nc	nc
γ^{NN},$g = rank-based$,$\varepsilon = 0.01$	2.76	12.29	83.84	100	-0.96	-0.84	12.16	5.612	nc	nc

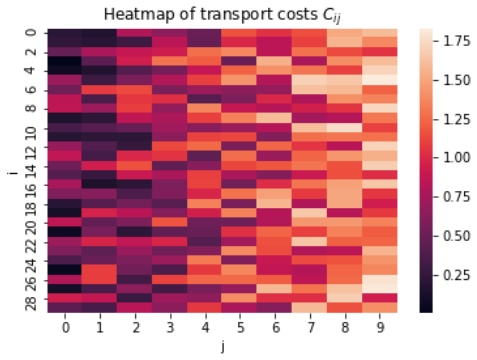
Table E.2: Results - *France Travail* dataset

Algorithm	Recall @1 (1)	Recall @10 (2)	Recall @100 (3)	Coverage @1 (4)	Coverage @10 (5)	Congestion @1 (6)	Congestion @10 (7)	OT Comp. Time sec. (8)
ϕ Random	0	0.21	0.65	99.95	100	-0.99	-0.99	
ϕ XGB	9.62	31.4	61.59	12.94	25.16	-0.62	-0.64	
ϕ NN	5.68	28.66	57.98	6.02	17.78	-0.46	-0.49	
CAROT - XGBoost								
$\gamma^{XGB}, g = exp+, \varepsilon = 1000.0$	3.93	16.3	52.18	20.98	34.25	-0.73	-0.75	35.93
$\gamma^{XGB}, g = exp+, \varepsilon = 100.0$	3.93	16.3	52.18	21	34.26	-0.73	-0.75	40.04
$\gamma^{XGB}, g = exp+, \varepsilon = 10.0$	3.93	15.86	52.18	21.03	34.33	-0.73	-0.75	49.91
$\gamma^{XGB}, g = exp+, \varepsilon = 1.0$	3.71	14.98	50.76	20.93	34.8	-0.73	-0.75	45.84
$\gamma^{XGB}, g = exp+, \varepsilon = 0.1$	1.53	11.59	49.67	27.23	44.7	-0.78	-0.80	55.20
$\gamma^{XGB}, g = exp+, \varepsilon = 0.01$	3.06	15.97	52.29	48.88	59.05	-0.86	-0.83	514.1
$\gamma^{XGB}, g = Id+, \varepsilon = 1000.0$	5.03	22.42	59.73	21.19	31.01	-0.74	-0.74	36.03
$\gamma^{XGB}, g = Id+, \varepsilon = 100.0$	5.03	22.42	59.4	21.18	31.01	-0.74	-0.74	40.03
$\gamma^{XGB}, g = Id+, \varepsilon = 10.0$	5.03	22.42	58.97	21.24	31.09	-0.74	-0.74	49.60
$\gamma^{XGB}, g = Id+, \varepsilon = 1.0$	4.81	21.99	57.87	21.61	31.76	-0.74	-0.75	48.27
$\gamma^{XGB}, g = Id+, \varepsilon = 0.1$	2.18	15.31	56.01	27.54	41.24	-0.78	-0.81	67.69
$\gamma^{XGB}, g = Id+, \varepsilon = 0.01$	4.37	20.45	43.21	46.75	57.61	-0.85	-0.79	448.8
$\gamma^{XGB}, g = ndcg, \varepsilon = 1000.0$	9.62	31.83	62.36	12.96	26.05	-0.62	-0.67	40.69
$\gamma^{XGB}, g = ndcg, \varepsilon = 100.0$	9.62	31.83	62.36	12.96	26.05	-0.62	-0.67	37.50
$\gamma^{XGB}, g = ndcg, \varepsilon = 10.0$	9.62	31.83	62.36	12.96	26.06	-0.62	-0.67	36.34
$\gamma^{XGB}, g = ndcg, \varepsilon = 1.0$	9.62	31.61	62.36	12.96	26.14	-0.62	-0.67	42.03
$\gamma^{XGB}, g = ndcg, \varepsilon = 0.1$	8.97	25.38	46.06	14.69	30.84	-0.67	-0.74	45.99
$\gamma^{XGB}, g = ndcg, \varepsilon = 0.01$	5.03	14	18.81	36.81	57.52	-0.82	-0.81	478.0
$\gamma^{XGB}, g = rank - based, \varepsilon = 1000.0$	9.4	27.13	60.5	15.82	37.2	-0.69	-0.73	36.36
$\gamma^{XGB}, g = rank - based, \varepsilon = 100.0$	9.4	27.13	60.5	15.82	37.2	-0.69	-0.73	36.28
$\gamma^{XGB}, g = rank - based, \varepsilon = 10.0$	9.4	27.13	60.28	15.85	37.2	-0.69	-0.73	39.69
$\gamma^{XGB}, g = rank - based, \varepsilon = 1.0$	9.4	26.91	59.19	16.09	37.28	-0.69	-0.73	45.54
$\gamma^{XGB}, g = rank - based, \varepsilon = 0.1$	7	22.53	44.42	24.06	38.74	-0.76	-0.79	49.69
$\gamma^{XGB}, g = rank - based, \varepsilon = 0.01$	2.18	11.59	21	56.69	68.13	-0.87	-0.85	312.7
CAROT - NN								
$\gamma^{NN}, g = exp+, \varepsilon = 1000.0$	5.25	20.35	51.2	19.7	32.96	-0.69	-0.71	36.46
$\gamma^{NN}, g = exp+, \varepsilon = 100.0$	5.25	20.35	51.2	19.73	32.96	-0.69	-0.71	39.29
$\gamma^{NN}, g = exp+, \varepsilon = 10.0$	5.25	20.13	50.98	19.83	33.1	-0.69	-0.71	49.46
$\gamma^{NN}, g = exp+, \varepsilon = 1.0$	4.15	20.24	50	21.37	34.41	-0.71	-0.72	49.30
$\gamma^{NN}, g = exp+, \varepsilon = 0.1$	0.65	6.89	42.23	35.04	50.43	-0.82	-0.83	58.90
$\gamma^{NN}, g = exp+, \varepsilon = 0.01$	2.62	17.39	37.85	58.32	65.97	-0.87	-0.80	490.8
$\gamma^{NN}, g = Id+, \varepsilon = 1000.0$	6.78	26.8	59.19	11.03	25.21	-0.60	-0.64	36.07
$\gamma^{NN}, g = Id+, \varepsilon = 100.0$	6.78	26.8	59.19	11.05	25.21	-0.60	-0.64	36.08
$\gamma^{NN}, g = Id+, \varepsilon = 10.0$	6.78	26.8	59.19	11.14	25.3	-0.60	-0.64	46.01
$\gamma^{NN}, g = Id+, \varepsilon = 1.0$	6.78	26.14	60.39	11.99	26.3	-0.62	-0.65	49.23
$\gamma^{NN}, g = Id+, \varepsilon = 0.1$	2.4	19.03	50.43	28.23	40.16	-0.80	-0.79	54.80
$\gamma^{NN}, g = Id+, \varepsilon = 0.01$	3.93	16.3	27.89	53.38	62.35	-0.83	-0.70	518.9
$\gamma^{NN}, g = ndcg, \varepsilon = 1000.0$	5.68	28.11	59.73	6.02	19.51	-0.46	-0.54	36.80
$\gamma^{NN}, g = ndcg, \varepsilon = 100.0$	5.68	28.11	59.73	6.02	19.51	-0.46	-0.54	37.60
$\gamma^{NN}, g = ndcg, \varepsilon = 10.0$	5.68	28.11	59.51	6.02	19.53	-0.46	-0.54	40.72
$\gamma^{NN}, g = ndcg, \varepsilon = 1.0$	5.68	27.46	59.08	6.02	19.75	-0.46	-0.55	45.86
$\gamma^{NN}, g = ndcg, \varepsilon = 0.1$	5.25	23.3	49.01	8.85	26.4	-0.53	-0.65	46.89
$\gamma^{NN}, g = ndcg, \varepsilon = 0.01$	1.53	12.36	24.28	35.41	51.56	-0.81	-0.81	517.8
$\gamma^{NN}, g = rank - based, \varepsilon = 1000.0$	5.68	25.27	52.95	10.74	38.47	-0.58	-0.67	37.20
$\gamma^{NN}, g = rank - based, \varepsilon = 100.0$	5.68	25.27	52.95	10.76	38.47	-0.58	-0.67	36.78
$\gamma^{NN}, g = rank - based, \varepsilon = 10.0$	5.68	25.27	52.95	10.79	38.5	-0.58	-0.67	40.34
$\gamma^{NN}, g = rank - based, \varepsilon = 1.0$	5.68	25.27	52.29	11.24	38.71	-0.59	-0.67	46.50
$\gamma^{NN}, g = rank - based, \varepsilon = 0.1$	3.06	12.58	40.7	26.55	42.65	-0.74	-0.77	49.85
$\gamma^{NN}, g = rank - based, \varepsilon = 0.01$	0.65	4.81	25.6	61.91	73.03	-0.89	-0.88	502.9

Table E.3: Computational runtime in seconds on *France Travail* (averaged over all g options). NN is trained on a server with 2 Intel Xeon Silver 4214 2,2GHz CPUs, 192Go RAM, and a Tesla T4 GPU. XGB is trained on a DELL PowerEdge R640 server with 2X Intel Xeon Gold 6130 2.10GHz CPUs (2 $\times$ 16 cores) and 384Go RAM. The optimal transport plan is computed on the DELL with same resources as for XGB.

Comp. Time	ϕ		γ^{XGB}			γ^{NN}		
	XGBoost	NN	$\epsilon = 0.01$	$\epsilon = 0.1$	$\epsilon = 1$	$\epsilon = 0.01$	$\epsilon = 0.1$	$\epsilon = 1$
Total	104,340	4,104	104,778	104,394	104,385	4,611	4,156	4,148
(inc. Learning/OT)	(7,454/−)	(2,039/−)	(7,454/438)	(7,454/54)	(7,454/45)	(2,039/507)	(2,039/52)	(2,039/44)

Figure E.1: Input cost matrix



E.2 Higher entropic regularization may not reduce congestion

After [PC19] (prop. 4.1), when the weight ε of the entropic regularization term goes to ∞, the solution γ of the regularized optimal transport problem tends to a uniform coupling. When $\epsilon \to 0$ instead the solution converges to the optimal transport plan with maximal entropy. Informally, increasing ϵ leads to solutions γ that are less sparse.

However, the exploitation of γ through the sorting recommendation process is such that a more uniform γ does not necessarily lead to less congestion.

This phenomenon is investigated in simulation. 1,000 cost matrices C of size $n = 30$, $m = 10$ are independently generated, with $C_{ij} \sim \mathcal{U}(\frac{j}{m}, \frac{j}{m} + 1)$ (items being ordered by increasing attractiveness). Transport plans γ with uniform marginals w.r.t. users and items are then computed using Sinkhorn algorithm with entropic regularization weight $\varepsilon = 100$ and $\varepsilon = 0.01$. The average and standard deviation over the 1,000 runs of the congestion obtained after sorting these plans indicate that the congestion is significantly higher for the higher value of ε:

ϵ	Mean congestion@1	Std.
100	-0.940521	0.029445
0.01	-0.996059	0.003586

Figures E.1, E.2, E.3 and E.4 illustrate this phenomenon on a single representative run. γ_{ij}'s are more uniform when $\epsilon = 100$ than when $\epsilon = 0.01$, yet sorting each line leads to a more unequal distribution of recommendations towards the different offers.

Altogether, higher entropic regularization has an indeterminate impact on congestion, and may increase it in practice. The choice of the ϵ should thus be chosen based on a validation set, as well as on numerical criteria for the convergence of Sinkhorn's algorithm. One may note that taking extremely small values of ϵ using a naive implementation of Sinkhorn's algorithm may have adverse consequences on numerical stability as well as convergence speed, although alternatives have been developed, for instance in [Sch19].

Figure E.2: $\epsilon = 100$

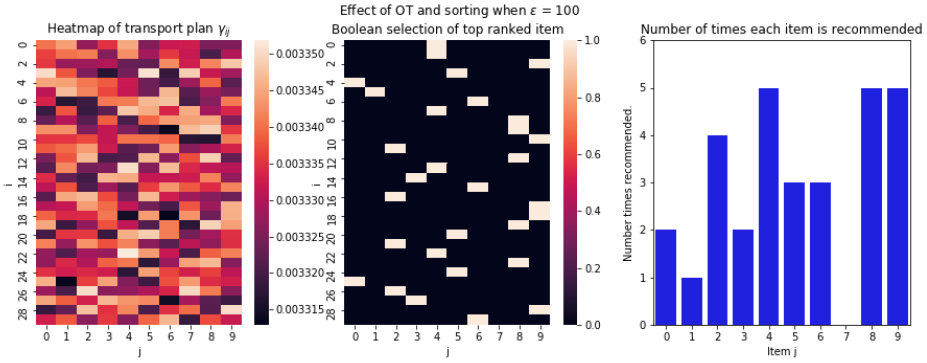


Figure E.3: $\epsilon = 0.1$

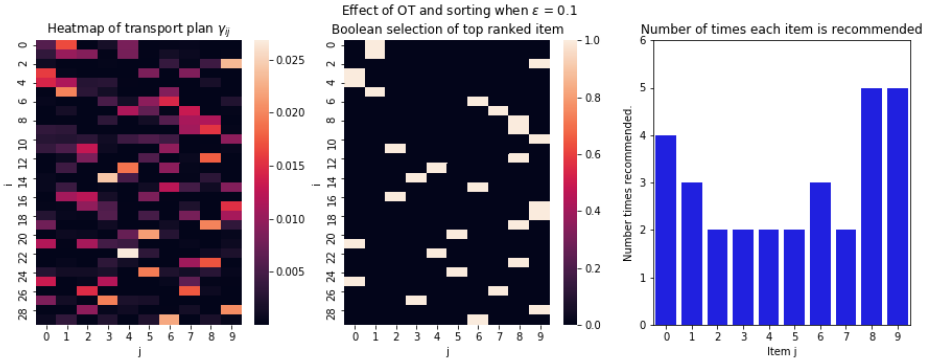
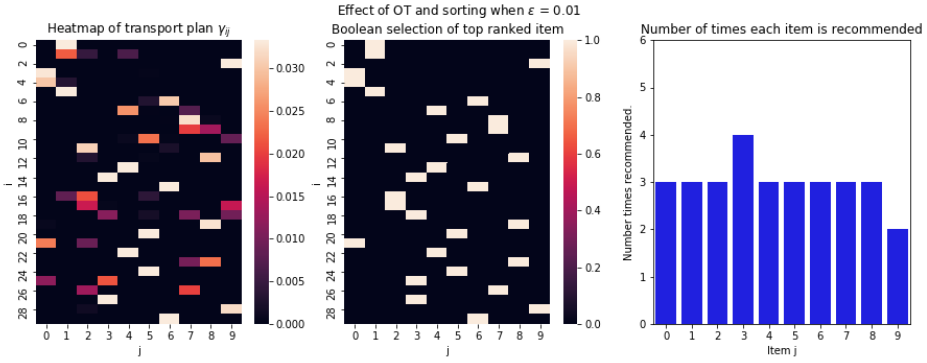


Figure E.4: $\epsilon = 0.01$



E.3 Hyperparameters

This appendix details the hyperparameters used to train XGB and NN on both benchmarks.

XGB

On MAR, XGB is used with its default parameters, except for the number of boosting rounds, set to 200. A logistic loss is used and the negative sampling ratio is set to 50 (Table E.4).

Table E.4: Xgb Hyperparameters on MAR

num_boost_round	200
Loss	Logistic
Negative sampling ratio	50

On *France Travail*, XGB is used with the hyper-parameters reported in Table E.5. Other hyper-parameters are set to their default value.

Table E.5: Xgb Hyperparameters on France Travail

col_sample_bytree	0.6
eta	0.075
gamma	0.85
max_depth	12
min_child_weight	1
subsample	0.9
num_boost_round	400
Loss	Logistic
Negative sampling ratio	50

NN

The margin parameter η in the triplet loss is set to 1 in all experiments.

On MAR, NN is used with the hyper-parameters reported in Table E.6. In each batch, 10 negative pairs are uniformly selected for each positive one.

On the *France Travail* dataset, the neural architecture is adapted to account for the domain knowledge, involving four modules:

A "geographic" 2-100-100-50 module takes as input the (standard-scaled) latitude and longitude, with 2 hidden layers of size 100 and outputs a representation of the user/item location in dimension 50. All activation functions are tanh. This module is trained for 100 epochs (batch size 32) with Adam optimizer and base learning rate 10^{-4}. Negative sampling selects items farther than the actual positive one.

Table E.6: NN Hyperparameters on MAR

Layer 1	tanh, size = 300
Embedding	tanh, size = 300
Optimizer	Adam
Learning rate	0.001
Epochs	300
Batch size	64
Negative sampling ratio (per epoch)	10

A "skill" 14,000-200-100 module takes as input the (standard-scaled) skills, with 1 hidden layer of size 200 (activation RELU) and outputs a representation of size 100 (activation function tanh). The module is trained for 100 epochs (batch size 32) with Adam optimizer and base learning rate 10^{-4}. The similarity matrix is diagonal.

An "other" d-500-200 module takes as input the other descriptive features, with $d = 448$ for users and $d = 582$ for items, with a hidden layer of size 500 (activation RELU) and outputs a representation of size 200 (activation function tanh). The module is trained for 100 epochs (batch size 32) with Adam optimizer and base learning rate 10^{-4}.

The overall architecture is warm-started using the preliminary training of the above three modules. The similarity matrix A is constrained to be block-wise diagonal. The module is trained for 35 epochs (batch size 256) with Adam optimizer and base learning rate 10^{-4}.

Except for the "geography" module, negative examples are sampled uniformly anew in each epoch, with a negative ratio of 50.

Other hyper-parameters are detailed in Table E.5.

Table E.7: Hyperparameters - NN (France Travail)

Geography module	
Layer 1	tanh, size = 100
Layer 2	tanh, size = 100
Embedding	tanh, size = 50
Optimizer	Adam
Learning rate	0.0001
Epochs	100
Batch size	32
Skills module	
Layer 1	ReLu, size = 200
Embedding	tanh, size = 100
Optimizer	Adam
Learning rate	0.0001
Epochs	100
Batch size	32
Other module	
Layer 1	ReLu, size = 500
Embedding	tanh, size = 200
Optimizer	Adam
Learning rate	0.0001
Epochs	100
Batch size	256
Training from warm start	
Block-diagonal Structure	True
Epochs	10 / 25
Optimizer	Adam
Learning rates	0.0001 / 0.00001
Batch size	256

Appendix F

Complements on gender gaps

F.1 Sample characteristics

Table F.1 displays the average characteristics of the job seekers in our sample of interest (present on the fourteenth week of 2022). Column 1 displays the population mean, while columns 2 and 3 respectively illustrate the average characteristics for men and women. Column 4 showcases the difference between women (column 3) and men (column 2), with the final column providing the associated p-value for the difference estimate.

On average, women possess higher levels of education compared to men. Specifically, 65% of women have attained post-secondary education, whereas 51% of men have. Women are more prevalent in sectors such as Services to individuals and communities, Business support, Sales and Retail, and Health, while men dominate in areas like Construction, Industrial Installation and Maintenance, and Transportation and Logistics. Additionally, women exhibit, on average, approximately one year less of experience than men (with men averaging 5 years of experience compared to 4 years for women).

Furthermore, preferences also exhibit gendered patterns. Women tend to seek job opportunities located, on average, 5 kilometers closer to their residences. They also demonstrate a lower reservation wage (the minimum salary accepted to work), which is €230 per month less than that of men. Additionally, women search less often for full-time contracts, with only 64% pursuing such positions compared to 83% of men.

	Pop. Mean	Men	Women	Women - Men	p-value
Qualifications					
Education					
Post Secondary Education	0.58	0.51	0.65	0.14	0.0
Vocational Certificate	0.28	0.34	0.23	-0.10	0.0
Other education	0.11	0.12	0.09	-0.02	0.0
Qualification category					
Unskilled workers	0.08	0.12	0.04	-0.08	0.0
Qualified workers	0.09	0.15	0.03	-0.13	0.0
Unskilled employees	0.22	0.18	0.26	0.08	0.0
Skilled employees	0.41	0.33	0.49	0.16	0.0
Foreman	0.09	0.10	0.08	-0.02	0.0
Executives	0.08	0.09	0.07	-0.02	0.0
Driving Licenses					
Car Driving License	0.05	0.08	0.02	-0.06	0.0
Truck driving license	0.03	0.05	0.00	-0.05	0.0
Number of driving licenses	0.73	0.89	0.58	-0.31	0.0
Sector					
Agriculture	0.04	0.05	0.03	-0.03	0.0
Arts and Crafts	0.01	0.01	0.01	0.01	0.0
Banking, Insurance and Real Estate	0.01	0.01	0.01	0.00	0.0
Sales and Retail	0.15	0.10	0.19	0.08	0.0
Communication	0.02	0.02	0.03	0.01	0.0
Construction	0.08	0.14	0.01	-0.14	0.0
Hotel, Restaurant, Tourism	0.10	0.10	0.10	0.00	0.4
Industry	0.07	0.10	0.05	-0.05	0.0
Installation and Maintenance	0.04	0.07	0.01	-0.07	0.0
Health	0.05	0.01	0.08	0.06	0.0
Services to individuals and Communities	0.17	0.08	0.26	0.18	0.0
Arts and Entertainment	0.01	0.01	0.01	-0.00	0.0
Business support	0.12	0.07	0.17	0.10	0.0
Transports and logistics	0.10	0.18	0.03	-0.15	0.0
Number of languages spoken	1.00	0.95	1.04	0.09	0.0
# Months of Experience	53.49	59.93	47.27	-12.67	0.0
Preferences					
Full Time	0.73	0.83	0.64	-0.18	0.0
Commuting Distance	27.74	30.23	25.33	-4.90	0.0
Reservation Wage	1940.50	2057.83	1827.15	-230.68	0.0
Number of hours	30.56	30.75	30.38	-0.37	0.0
Permanent Contract	0.58	0.59	0.57	-0.02	0.0

Table F.1: Sample characteristics

F.2 Additional tables

	Sample size	Number men	Number women	% women
Full Week 2022-14				
Full week	358682	176244	182438	0.509
Full week (overlap Z)	293579	138824	154755	0.527
Full week (overlap Z_p)	291870	137982	153888	0.527
Hires				
Hires	41787	19496	22291	0.533
Hires (overlap Z)	34622	15465	19157	0.553
Hires (overlap Z_p)	34532	15380	19152	0.555
Applications				
Applications	97179	39238	57941	0.596
Applications (overlap Z)	80542	32670	47872	0.594
Applications (overlap Z_p)	80263	32472	47791	0.595

Notes: The first column presents the total sample size for the different datasets used in the analysis. “Full week" and “Full week (overlap)" present the sample size for a week in the test set before and after restriction to job seekers satisfying the overlap condition required in the AIPW method of Section 7.2. “Hires", “Hires (overlap)", and “Hires & Applications (overlap)" respectively present the sample sizes for the subsamples of job seekers in the test set who have been hired, hired and for whom the overlap condition holds, and the subset of the latter one where we also observe applications.

Table F.2: Size of datasets used for the analysis

	Observed				Recommendations			
	Men	Women	Diff.	p-value	Men	Women	Diff.	p-value
Total Population								
Wage (log)					7.491	7.470	-0.021	0.0
Distance					7.052	6.702	-0.350	0.0
Executive Position					0.029	0.024	-0.005	0.0
Indefinite duration					0.472	0.432	-0.040	0.0
Full time					0.881	0.684	-0.197	0.0
Experience					6.739	5.044	-1.696	0.0
Fit to job search parameters					0.503	0.472	-0.031	0.0
% Observations					176244	182438		
Hirings								
Wage (log)	7.454	7.436	-0.019	0.00	7.446	7.430	-0.016	0.000
Distance	25.59	21.69	-3.894	0.00	7.499	7.359	-0.140	0.012
Executive Position	0.013	0.012	-0.001	0.31	0.013	0.013	0.000	0.903
Indefinite duration	0.462	0.423	-0.039	0.00	0.435	0.399	-0.037	0.000
Full time	0.868	0.678	-0.190	0.00	0.865	0.673	-0.192	0.000
Experience	7.754	6.569	-1.185	0.00	7.240	5.885	-1.355	0.000
Fit to job search parameters	0.516	0.481	-0.034	0.00	0.516	0.482	-0.034	0.000
% Observations	19496	22291			19496	22291		
Applications								
Wage (log)	7.489	7.460	-0.029	0.0	7.456	7.434	-0.023	0.0
Distance	32.50	23.15	-9.332	0.0	7.296	7.126	-0.170	0.0
Executive Position	0.040	0.021	-0.019	0.0	0.021	0.014	-0.007	0.0
Indefinite duration	0.549	0.492	-0.058	0.0	0.454	0.394	-0.060	0.0
Full time	0.863	0.694	-0.169	0.0	0.856	0.667	-0.189	0.0
Experience	10.453	9.575	-0.877	0.0	8.289	7.326	-0.963	0.0
Fit to job search parameters	0.514	0.475	-0.038	0.0	0.502	0.465	-0.037	0.0
% Observations	39238	57941			39238	57941		

Table F.3: Average recommended job characteristics and observed behavior with respect to job ads for the total population, the subsample of hired jobseekers and the subsample of applicants

	Differences between women and men				Difference of Differences	
A. In hirings	$\tau_{\text{Hire}}(Observed)$	p-value	τ (MUSE)	p-value	τ_{DifH} (MUSE)	p-value
Wage (log)	-0.009	0.002	-0.006	0.000	0.003	0.312
Distance	-2.352	0.013	0.588	0.000	2.801	0.002
Executive Position	-0.003	0.454	-0.001	0.450	0.003	0.783
Indefinite duration	-0.045	0.001	-0.012	0.010	0.031	0.066
Full time	-0.078	0.000	-0.046	0.000	0.032	0.020
Experience	0.323	0.579	0.142	0.561	-0.263	0.730
Fit to job search parameters	-0.021	0.000	-0.019	0.000	0.003	0.560
B. In applications	τ_{App} (Observed)	p-value	τ (MUSE)	p-value	τ_{DifA} (MUSE)	p-value
Wage (log)	-0.008	0.000	-0.004	0.000	0.005	0.004
Distance	-5.341	0.000	0.493	0.000	5.725	0.000
Executive Position	-0.004	0.076	-0.001	0.029	0.003	0.246
Indefinite duration	-0.029	0.000	-0.014	0.000	0.016	0.167
Full time	-0.057	0.000	-0.045	0.000	0.011	0.008
Experience	-0.142	0.493	-0.033	0.899	0.063	0.500
Fit to job search parameters	-0.015	0.000	-0.015	0.000	-0.002	0.199

Notes: The results are presented in the subsample of hired job seekers. Due to different data sources, we study the sub-population of job seekers with hires in the testing weeks for which we observe applications (all weeks taken together). The first column presents the conditional estimates for the gender gaps on observed hirings (resp. observed applications) between women and men for the population with common support. The third one presents the same difference on the characteristics of the algorithm's recommendations. For hirings, the differences with the doubly robust effects presented in the fifth column of Table 7.2 are due to the restriction on the subsample of hired job seekers. The fifth column reports the difference of two latter differences, i.e., the conditional estimates for the differences between a hire's characteristics (resp application's) and the algorithm's recommendation.

Table F.4: Conditional gender gaps in hires and applications and in the algorithm's recommendations on the subsample of hired job seekers (Doubly Robust estimators)

	Pop. Mean	Wage	Distance	Executive Qualif.	Indef. Duration	Full Time	Adequ.	Experience
Other Individual Characteristics								
Age								
Less than 30	0.312	0.221	0.288	0.248	0.272	0.277	0.267	0.180
Between 30 and 50	0.540	0.580	0.577	0.595	0.579	0.562	0.568	0.573
More than 50	0.148	0.199	0.135	0.157	0.149	0.161	0.165	0.247
Family Status								
Is Married	0.368	0.431	0.353	0.432	0.347	0.352	0.396	0.419
Have Children	0.415	0.442	0.412	0.414	0.404	0.397	0.429	0.430
City								
Live in a priority district	0.092	0.069	0.105	0.052	0.087	0.064	0.098	0.085
Unemployment history								
Unemployed > 1 year	0.414	0.453	0.437	0.449	0.426	0.421	0.428	0.442
Qualifications								
Executive qualification	0.094	0.165	0.030	0.360	0.067	0.061	0.106	0.174
Months of Experience	52.887	78.425	37.406	69.499	51.820	61.385	57.257	98.218
Education								
Post Secondary Education	0.614	0.672	0.652	0.866	0.557	0.573	0.685	0.489
Sector								
Agriculture	0.043	0.053	0.029	0.029	0.115	0.023	0.035	0.029
Arts and Crafts	0.010	0.067	0.018	0.037	0.053	0.026	0.009	0.020
Banking, Insurance and Real Estate	0.015	0.025	0.009	0.024	0.008	0.025	0.026	0.006
Sales and Retail	0.165	0.062	0.114	0.057	0.108	0.213	0.164	0.043
Communication	0.026	0.012	0.011	0.025	0.012	0.025	0.020	0.003
Construction	0.040	0.067	0.023	0.043	0.019	0.045	0.084	0.162
Hotel, Restaurant, Tourism	0.117	0.096	0.090	0.089	0.221	0.290	0.076	0.187
Industry	0.079	0.269	0.077	0.161	0.048	0.052	0.103	0.208
Installation and Maintenance	0.027	0.049	0.015	0.019	0.038	0.068	0.030	0.055
Health	0.049	0.052	0.054	0.011	0.042	0.017	0.049	0.014
Services to individuals and Communities	0.159	0.066	0.179	0.112	0.110	0.053	0.122	0.102
Arts and Entertainment	0.015	0.019	0.016	0.032	0.015	0.008	0.005	0.001
Business support	0.123	0.087	0.084	0.289	0.027	0.069	0.116	0.051
Transport and logistics	0.097	0.069	0.264	0.061	0.166	0.055	0.144	0.109
Preferences								
Full Time	0.729	0.788	0.768	0.774	0.818	0.796	0.767	0.794
Commuting Distance	27.510	31.704	30.180	35.405	31.043	26.977	28.034	30.672
Reservation Wage	1923.093	2440.376	1714.099	3248.486	1960.697	2143.989	1957.713	2229.719
Permanent Contract	0.575	0.649	0.568	0.685	0.571	0.596	0.670	0.630
Average Loss of the 10%		-0.025	-2.789	-0.017	-0.084	-0.172	-0.051	-1.825

Notes: The first column presents the population mean (with overlap) on the fourteenth week of 2022.The rest of the columns shows the average characteristics of a subset of the population that would incur the largest costs in terms of wages, distance, executive status, type of contract, type of contract hours, fit to job search parameters and experience (respectively in columns 2,3,4,5,6,7,8) if they were a woman. This population is constructed by taking the 10% most affected according to each corresponding doubly robust estimator.

Table F.5: Sample characteristics of the 10% individuals experiencing the highest loss in job ad characteristics

	Low	Intermediate	High
Wage (log)	0.0050	0.0025	0.0010
Distance	0.5000	0.25	0.1250
Executive Position	0.0025	0.001	0.0005
Indefinite duration	0.0100	0.005	0.0025
Full time	0.0100	0.005	0.0025
Experience	1.0000	0.5	0.2500

Table F.6: Value of the constraint on gender gaps (ϵ) for each intensity level

	Initial		Low			Intermediate			High		
	value	p-value	ϵ	value	p-value	ϵ	value	p-value	ϵ	value	p-value
Unconditional gaps											
Wage (log)	-0.0045	0.0000	0.0050	-0.0048	0.0000	0.0025	-0.0028	0.0003	0.0010	-0.0013	0.0803
Distance	0.7620	0.0000	0.5000	0.5044	0.0000	0.2500	0.2587	0.0064	0.1250	0.1223	0.3248
Executive position	0.0008	0.1529	0.0025	0.0010	0.0874	0.0010	0.0011	0.0639	0.0005	0.0006	0.2959
Indefinite duration	-0.0110	0.0000	0.0100	-0.0108	0.0000	0.0050	-0.0058	0.0094	0.0025	-0.0033	0.1454
Full time	-0.0303	0.0000	0.0100	-0.0105	0.0000	0.0050	-0.0055	0.0090	0.0025	-0.0030	0.1376
Experience	-0.1087	0.0182	1.0000	-0.0704	0.1086	0.5000	-0.0333	0.4359	0.2500	-0.0041	0.8664
Fit to job search parameters	-0.0080	0.0000	0.0000	-0.0056	0.0001	0.0000	-0.0043	0.0021	0.0000	-0.0035	0.0136
Performance indicators											
R@10	0.2387			0.2385			0.2385			0.2386	
R@10 (Women)	0.2504			0.2502			0.2502			0.2503	
R@10 (Men)	0.2243			0.2241			0.2242			0.2242	
Other descriptive statistics											
Gini Index	0.7971			0.7971			0.7971			0.7971	
% Offers modified	0.0000			0.0029			0.0042			0.0050	
% Ranking modified	0.0000			0.0283			0.0403			0.0474	

Notes: The results are presented using recommendations generated for job seekers (and job ads) existing at week 2022-14 for which the estimated propensity score is within the interval [0.01, 0.99]. Propensity scores are estimated through regularized regression. Variables that are used for conditioning (Z) include all job seekers qualifications. We present results for an increase set of conditional constraints ϵ that we call weak, intermediate, and high. The set of ϵ values is given for each intensity level, respectively, in the third, sixth, and ninth columns. The first two columns restate the results of the standard algorithm in terms of conditional gender gaps for comparison purposes. The point estimates (columns 1, 3, 6, 9) are calculated following the methodology in 7.4 (IPW) and associated p-values are calculated using bootstrap estimation (100 randomly selected samples drawn from the initial population).

Table F.7: Conditional gender gaps in job ads characteristics for post-processed recommendations

F.3 Additional figures

Figure F.1: **Distribution of propensity scores among job seekers**

F.4 Details regarding the post-processing approach

In this Appendix, we begin by formulating the integer linear program of recommendation under gender gap constraints in explicit fashion (Section F.4.1). We then then provide more insights on how the dualized program $L(\lambda)$ can be efficiently evaluated for a given value of λ (Section F.4.2).

F.4.1 Formulation of gender gap constraints

Recall the notations of section 7.5: there are n users, m job ads, and for each user i, k recommendations to generate. The variable to be optimized are the nm binary variables $\{\gamma_{ij}\}, i \in [[1,n]], j \in [[1,m]]$ where $\gamma_{ij} = 1$ denotes that job ad j is shown to job seeker i. The quality of a given recommendation policy is measured by $\sum_{i=1}^{n}\sum_{j=1}^{n} s_{ij}\gamma_{ij}$, where scores s_{ij}'s are the pairwise scores provided by the recommender system.

Let f be a bijection from $[[1,n]] \times [[1,m]]$ to $[[1,nm]]$, and let $i : \mathbf{R}^{nm} \to [[1,n]], i(p) = \mathrm{pr}_1(f^{-1}(p))$, $j : \mathbf{R}^{nm} \to [[1,m]], j(p) = \mathrm{pr}_2(f^{-1}(p))$.

The problem of interest can be rewritten as the choice of $\gamma \in \mathbb{R}^{nm}$, such that $\gamma_p \in \{0,1\}$ corresponds to whether to whether job ad $j(p)$ is recommended to job seeker $i(p)$. Let $\mathbf{s} \in \mathbb{R}^{nm}$ be a vector representations of scores s_{ij}, such that $\mathbf{s}_p = s_{i(p),j(p)}$ The unconstrained recommendation problem then formally becomes:

$$\max_{\gamma \in \mathbb{R}^{nm}} \mathbf{s}^T\gamma$$

subject to:

$$\gamma_p \in \{0,1\} \quad \forall p \in [[1,nm]]; \qquad \sum_{j=1}^{m} \gamma_{f(i,j)} = k \quad \forall i$$

Now, recall for instance the constraint on unconditional average gender gaps:

$$-\epsilon \leq \frac{1}{k}\sum_{i:g_i=1}\sum_{j}\frac{w_{ij}}{n_{g_i}}\gamma_{ij} - \frac{1}{k}\sum_{i:g_i=0}\frac{w_{ij}}{n_{g_i}}\gamma_{ij} \leq \epsilon$$

It may be rephrased as:

$$a^T\gamma \leq \epsilon \qquad \text{and} \qquad -a^T\gamma \leq \epsilon$$

where $a \in \mathbb{R}^{nm}$ has components

$$a_p = \frac{1}{k}(2 \times g_{i(p)} - 1)\frac{w_{i(p)j(p)}}{n_{g_{i(p)}}}$$

And thus as

$$A\gamma \leq b$$

where A is obtained by stacking vertically a and $-a$, and $b = (\epsilon, \epsilon)^T$.

F.4.2 Efficiently evaluating L

Recall that

$$L(\lambda) = \max_{\gamma \in \mathbb{R}^{nm}} s^T \gamma - \lambda(\mathbf{b} - \mathbf{A}\gamma)$$

subject to:

$$\gamma_p \in \{0,1\} \quad \forall p \in [[1, nm]]; \qquad \sum_{j=1}^{m} \gamma_{f(i,j)} = k \quad \forall i$$

The value of γ minimizing $L(\lambda)$ also minimizes

$$\max_{\gamma} (s^T + \lambda \mathbf{A})\gamma \qquad \text{s.t.} \qquad \gamma_p \in \{0,1\} \quad \forall p \in [[1, nm]]; \qquad \sum_{j=1}^{m} \gamma_{f(i,j)} = k \quad \forall i$$

$$= \max_{\gamma} \sum_i \sum_j (s_{ij} + \sum_d \lambda_d \mathbf{A}_{f(i,j),d}) \gamma_{ij} \qquad \text{s.t.} \qquad \gamma_{ij} \in \{0,1\} \forall i,j \qquad \sum_j \gamma_{ij} = k \quad \forall i$$

This simply amounts, for each job seeker i, to choosing the top j job ads in terms of $(s_{ij} + \sum_d \lambda_d \mathbf{A}_{f(i,j),d})$. Finding the top k elements in an array of size m can be done efficiently, for instance in average complexity $O(m + k \log k)$, or worst case complexity $O(m + k \log m)$.

Appendix G

Thesis summary in French (*résumé substantiel*)

En apprenant des appariements passés, les systèmes de recommandation ont le potentiel de réduire les frictions informationnelles sur le marché du travail et d'améliorer l'appariement entre demandeurs d'emploi et recruteurs.

Cette thèse étudie la question de la conception et de l'évaluation d'algorithmes de recommandation d'offres d'emploi, en s'appuyant sur des données détaillées fournies par le service public de l'emploi français (*France Travail*).

Premièrement, nous proposons une nouvelle architecture neuronale pour la recommandation d'offres d'emploi, visant à répondre au problème du démarrage à froid (présentation de recommandations à des nouveaux utilisateurs) tout en passant à l'échelle. Dans une première étape, offres et demandeurs d'emplois sont représentés par des plongements dans un espace latent appris des embauches, dont la structure prend en compte les spécificités du domaine d'application (représentations dédiées à la géographie, aux compétences et aux métiers, à des informations générales). Ces représentations permettent la sélection rapide d'un sous-ensemble d'offres pour un demandeur d'emploi. Ces offres sont ensuite reclassées lors d'une deuxième étape par une architecture exploitant des informations plus détaillées au niveau de la paire demandeur-offre et une architecture plus complexe comportant des interactions multiplicatives, apprise des embauches et candidatures. L'approche proposée est validée de manière comparative en termes de performance hors-ligne et de rapidité de génération des recommandations sur des données publiques et propriétaires. Elle est également évaluée sur le terrain en termes de satisfaction des utilisateurs au moyen d'expériences randomisées à grande échelle, conduites en mars 2022 et juin 2023. L'approche s'avère compétitive à l'état de l'art comme au système expert actuel de *France Travail*.

Deuxièmement, nous examinons les objectifs possibles qu'un concepteur pourrait assigner à un algorithme de recommandation d'offres d'emploi. Cette analyse est réalisée dans le cadre d'un modèle économique, qui nous permet de discuter les mérites et limites de différentes approches plausibles (satisfaire les critères de recherche exacts des demandeurs, apprendre des candidatures ou des embauches), et de les confronter aux besoins des demandeurs d'emploi. Un algorithme aligné à ces besoins combinerait deux quantités cruciales, la probabilité d'embauche en

cas de candidature et l'utilité associée à un emploi, dont l'estimation est cependant complexe.

Troisièmement, nous étudions le problème de la congestion qui peut survenir si les recommandations se concentrent sur un ensemble excessivement restreint d'offres, créant des conséquences nuisibles au niveau agrégé. A l'aide d'outils issus du transport optimal computationnel, nous proposons une approche algorithmique pour limiter la congestion dans des recommendations, et étudions ses performances sur des données publiques et propriétaires.

Enfin, comme les algorithmes de recommandations sont entraînés sur des données issues du monde réel, ils peuvent reproduire ou aggraver certains comportements indésirables (notamment, de discrimination) existants sur le marché du travail. Afin de répondre à ces inquiétudes, nous réalisons un audit de l'algorithme de recommandation (entraîné à partir des embauches) en termes d'inégalités de genre. En s'inspirant de la littérature en économie du travail, nous proposons des mesures des écarts genrés en termes de caractéristiques des recommandations (salaire, type de contrat...), en moyenne ou conditionnellement aux qualifications et préférences des demandeurs d'emploi. Selon nos résultats, l'algorithme reproduit, sans aggraver, les biais de genre présents dans les données d'entraînement. Nous proposons également une approche dite de "post-traitement", s'appuyant sur la relaxation lagrangienne d'un problème d'optimisation linéaire en nombre entiers, dont l'objectif est de réduire les écarts femmes-hommes en termes de caractéristiques des offres recommandées. Nous décrivons les arbitrages entre performance et équité que cette intervention implique.