O Problema de Escassez de *Matchings* em Recomendações nos Domínios de Recrutamento

Alan Cardoso[1], Fernando Mourão[2], Leonardo Rocha[1]
[1]Universidade Federal de São João del-Rei
[2]Seek AIPS
alanc@ufsj.edu.br,fmourao@seek.com.au,lcrocha@ufsj.edu.br

ABSTRACT

Candidates and job vacancies may remain for long periods without real opportunities in Recommendation Systems (RSs) for online recruitment. We refer to these scenarios as the **Matching Scarcity Problem** (**MaSP**). We formalize the MaSP and propose a strategy to identify candidates and vacancies suffering from it. We also propose five heuristics, which suggest changes in CVs and jobs descriptions, to mitigate the MaSP. The best heuristic was able to reduce up to 50% the number of CVs and jobs suffering from MaSP.

KEYWORDS

Job Recommendation, Matching Scarcity, User Modeling

1 Introdução

Os serviços de recrutamento on-line são especializados em encontrar casamentos (*matchings*) entre vagas de trabalho e candidatos qualificados e têm atraído um número grande de usuários [7]. Uma das principais preocupações em Sistemas de Recomendação (SsR) para aplicações de recrutamento on-line é garantir *matchings* reais para todos. De fato, longos períodos aguardando por *matchings* implicam em oportunidades de negócios perdidas. Denominamos esses cenários onde candidatos e vagas sofrem com a ausência de oportunidades de *matchings* como **Matching Scarcity Problem** (**MaSP**). Consideramos o MaSP uma ameaça real, dado os desbalanceamentos sazonais entre oferta e demanda em diferentes áreas profissionais, em que algumas áreas podem ter mais candidatos do que vagas, e vice-versa, em diferentes momentos [1].

A partir do contexto acima, definimos nossa primeira hipotese: **(H1) Existe um conjunto identificável de candidatos/vagas que passam longos períodos sem conseguirem um número adequado de *matchings* em sistemas de recrutamento on-line.** Para validá-la, formalizamos o MaSP e propomos um método heurístico para identificar automaticamente esses candidatos e vagas que sofrem de escassez de oportunidades. Em seguida, definimos nossa segunda hipótese: **(H2) Existe um conjunto identificável de *features* contidas em currículos/vagas que trazem maior ganho de informação para SsR, as quais, quando presentes, favorecem a tarefa de identificação de *matchings*.** Para validar **H2**, propomos um segundo método heurístico capaz de identificar características que potencialmente acentuam o MaSP e as que o amenizam. Assim, definimos nossa terceira hipótese: **(H3) Auxiliar candidatos/recrutadores durante o processo de descrição dos currículos/vagas, adicionando *features* mais relevantes (identificadas em H2) é uma forma eficaz para se mitigar cenários de escassez.** Para validar **H3**, propomos cinco estratégias capazes de sugerir mudanças nas descrições dos currículos/vagas, aproximamos candidatos sofrendo com o MaSP de vagas de trabalho semanticamente relacionadas a eles. Avaliações em amostras de dados reais providas pela empresa Catho, apontam que nossas estratégias podem reduzir em até 50% o número total de vagas e CVs sofrendo com o MaSP.

Dessa forma, como principais contribuições desse trabalho destacamos: *(i) a formalização e caracterização de um problema novo (MaSP); (ii) a proposta de estratégias para identificar automaticamente CVs e vagas que sofram com MaSP; (iii) proposta de estratégias para mitigar o MaSP; e (iv) avaliação de todas as metodologias e estratégias propostas em dados reais.* **Essa dissertação foi fruto de um projeto de P&D entre a empresa Catho e a UFSJ e resultou na publicação de dois artigos no WebMedia [1, 2] e outro no periódico Expert Systems with Applications (qualis A1 - Factor Impactor 6.954) [3]. Essa dissertação também contribuiu com outro artigo no WebMedia [4]**.

2 Trabalhos Relacionados

Sistemas de recrutamento on-line fazem recomendações bilaterais, as quais precisam levar em conta a satisfação mútua do candidato e do recrutador [9]. Trata-se de um cenário com desafios e características que o diferenciam dos cenários tradicionais de recomendação [12]. A qualidade e utilidade das informações inseridas por candidatos e recrutadores nesses sistemas devem ser cuidadosamente avaliadas antes destas serem exploradas por SsR. Ademais, trata-se de um cenário dinâmico, no qual oferta e demanda por vagas podem variar significativamente ao longo do tempo. A falha em lidar com esses desafios pode levar os usuários a longos períodos sem *matchings*. Nesse artigo, nos referimos a esses cenários como o **Matching Scarcity Problem** (**MaSP**).

Os trabalhos focados em SsR para serviços de recrutamento podem ser classificados em três classes. A primeira classe contêm trabalhos focados em adaptar SsR utilizados em domínios tradicionais, como entretenimento, para recrutamento. Por exemplo, a combinação de diferentes estratégias de recomendação tem sido estudada com o foco em tornar os SsR mais robustos ao cenário de recrutamento [8]. A segunda classe de trabalhos foca em uma melhor compreensão das dinâmicas do comportamento dos usuários em sistemas de recrutamento on-line, para explorar padrões úteis para as tarefas de recomendação e busca. Em [11], os autores procuram entender a evolução das carreiras dos candidatos, fornecendo insumos para os SsR detectarem quais as próximas vagas de trabalho que seriam interessantes para eles. Por fim, a terceira classe de trabalhos é composta por esforços em pré-processamento dos dados de entrada, para melhor estruturar a informação fornecida

In: III Concurso de Teses e Dissertações (CTD 2021), Minas Gerais, Brasil. Anais Estendidos do Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia). Porto Alegre: Sociedade Brasileira de Computação, 2021.

ISSN 2596-1683

por usuários. Dessa forma, a informação pode ser aplicada posteriormente em estratégias de recomendação tradicionais de maneira mais eficiente [5]. Considerando essa organização da literatura, essa dissertação está mais relacionada aos trabalhos da terceira classe, uma vez que também propomos estratégias focada em pré-processar a informação de vagas e currículos fornecidos pelos usuários. Contudo, diferente de todos esses esforços, estamos focados no MaSP.

3 O Problema da Escassez de Matchings

De forma sucinta, o MaSP pode ser formalizado pela sequência de definições abaixo:

Definição 3.1. Screening feature. *Qualquer feature*[1] *útil para recrutadores/candidatos tomarem uma decisão ou julgar um currículo/vaga. Elas são: (1) interpretáveis, pois os atores do domínio as reconhecem como significativas; (2) populares, sendo frequentemente usadas por recrutadores ou candidatos; (3) discriminativas, sendo observadas com mais frequência em matchings de CVs/vagas do que em pares sorteados aleatoriamente.*

Definição 3.2. Appendant feature. *Features não contempladas pela Definição 3.1.*

Definição 3.3. Matching. *Estado no qual a similaridade*[2] *entre o conjunto de features F_v e F_c extraídas, respectivamente, de uma dada vaga v e CV c é acima de um threshold τ.*

Definição 3.4. Escassez. *Estado de uma vaga v ou CV c no qual há menos que μ matchings distintos abrangendo v ou c em um período de tempo T.*

Definição 3.5. The Matching Scarcity Problem (MaSP). *Cenários nos quais uma vaga v ou um CV c apresenta escassez por período de tempo maior que o threshold T_M.*

Note na Definição 3.5 que, apesar do MaSP ser um problema personalizado, uma vez que candidatos e recrutadores distintos podem tolerar diferentes valores de T_M, na prática, definir T_M representa uma decisão de negócio. O serviço de recrutamento pode determinar um valor capaz de atender suas necessidades e o perfil geral dos seus usuários.

4 Mitigando o MaSP

Propomos um processo com três etapas para mitigar o MaSP: (1) identificar itens sofrendo com o MaSP; (2) identificar *features* que potencialmente intensifiquem o MaSP; (3) sugerir alterações nos CVs e vagas com o objetivo de melhorar suas descrições.

4.1 Identificação de Itens Sofrendo do MaSP

Para abordar **H1**, definimos um *threshold* τ (Definição 3.3) como o limite usado para determinar a existência de um *matching* entre CVs e vagas. Precisamos também encontrar o número mínimo de *matchings* μ utilizado para definir escassez (Definição 3.4). Propomos uma solução baseada em ***Bayesian Inference*** [6]: observamos como o domínio se comportou no passado para inferir como ele se comportará no futuro. Inicialmente, definimos μ como o menor valor observado no domínio e obtemos o total de itens identificados como escassos. μ é gradualmente incrementado e, a cada incremento, é avaliado o impacto no número de itens identificados como sofrendo de escassez. Quando um dado incremento não causar diferenças significativas no números de itens apontados como em cenário de escassez, μ é o valor do incremento imediatamente anterior ao atual.

Propomos agora encontrar o valor de τ utilizando o ***Maximum Likelihood Estimation (MLE)*** [10]. Dado o valor mínimo de μ, encontramos o valor de τ que gere uma distribuição de *matchings* que melhor se aproxime da distribuição observada. A estratégia se inicia definindo τ como o menor valor possível, incrementando-o iterativamente. A cada novo valor de τ, avaliamos a probabilidade de ocorrência de μ *matchings* na coleção de itens. Quando para um dado valor de τ, a probabilidade de μ cai, significa que a aproximação da distribuição de probabilidade observada a priori diminuiu. Portanto, o valor de τ imediatamente anterior é o escolhido. Tendo definido os valores de τ e μ, propomos um classificador binário para identificar CVs/vagas que sofram com o MaSP, como apresentado no Algoritmo 1. Esse algoritmo classifica um CV, considerando todas as vagas do conjunto de dados e o período de tempo T de análise. O mesmo processo se aplica a vagas ao trocarmos a entrada para vaga e conjunto de CVs.

Algoritmo 1 Classificador Binário de Escassez

ENTRADA: CV, vagas, T
SAÍDA: True/False para cenário de escassez

```
μ, τ = getParametros()
totalMatchings = 0, escassez = False
vagasValidas ← filtraItensPorTempo(vagas, T)
for all vaga ∈ vagasValidas do
    similaridade ← getSimilaridade(CV, vaga)
    if similaridade >= τ then
        totalMatchings ← totalMatchings + 1
    end if
end for
if totalMatchings < μ then
    escassez ← True
end if
Return escassez
```

4.2 Identificação de *Appendant Features*

CVs e vagas possuem características que podem ser divididas em dois grupos, *Screening* e *Appendant features*. As *Screening* são caracterizadas como interpretáveis, populares e discriminativas. Visando abordar **H2**, propomos um Classificador de *Features* (Algoritmo 2), que considera esses três aspectos para discriminar *Screening* e *Appendant features*.

O algoritmo recebe como entrada o item a ser processado, o conjunto com todas as *features* distintas extraídas previamente de todos os CVs e vagas da coleção, e um *thesaurus* de *features* do domínio gerado por especialistas da área. O método gera inicialmente a distribuição de frequência de todas as *features* para encontrar o *threshold* de popularidade ω para *Screening features*. Então, o método verifica se cada *feature* é: (1) interpretável ao verificar se ela pertence ao *thesaurus* do domínio; (2) popular ao ter sua frequência comparada ao *threshold* ω da distribuição de frequência global; e (3) discriminativa ao checar se a probabilidade de ocorrência da mesma

[1]Uma feature é qualquer palavra ou número usado na descrição de CVs ou vagas.
[2]Nesse trabalho adotamos similaridade cosseno, mas outras métricas poderiam ser utilizadas.

Algoritmo 2 Classificador de Features

ENTRADA: CV, FeaturesDominio, ThesaurusDominio
SAÍDA: Lista de *Screening* e *Appendant features* extraídas de um CV

```
ScreeningFeatures ← listaVazia()
FeaturesItem ← getFeaturesItem(CV, FeaturesDominio)
ω ← getThresholdPopularidade(FeaturesDominio)
for all feature ∈ FeaturesItem do
    if (feature.nome ∈ ThesaurusDominio) and
       (feature.frequencia >= ω) and
       isFeatureDiscriminative(feature) then
        ScreeningFeatures.insere(feature)
    end if
end for
AppendantFeatures ← FeaturesItem − ScreeningFeatures
Return (ScreeningFeatures, AppendantFeatures)
```

é maior em pares de $(CV, vaga)$ com contatos do que em pares selecionados aleatoriamente. Todos as *features* candidatas que não se encaixarem nesses critérios são consideradas como *Appedant*.

4.3 Sugestão de Edições para Descrições de Itens

Para tratar **H3**, propomos cinco soluções para mitigar o MaSP. Elas são baseadas na edição de 10 características nas descrições dos itens, removendo *Appendant features* ou trocando-as por *Screening*. Propomos a construção de representações baseadas em *embeddings* para cada *feature* com o objetivo de medir similaridade, usando o algoritmo Word2Vec.

- **Appendant Feature Removal (AR):** esse método remove cada uma das *Appendant features* do currículo/vaga. O objetivo é melhorar o CV/vaga por meio de descrições mais concisas que contenham apenas informações úteis.
- **Appendant Feature Exchange for Screening Features (AES):** realiza a troca de cada *Appendant feature* de um CV/vaga por uma *Screening feature* semanticamente semelhante à removida. Para isso, utiliza *embeddings* treinados com o conjunto de todas as habilidades provenientes de currículos e vagas. O objetivo é encontrar *features* semelhantes às que serão removidas, porém mais interessantes do ponto de vista de um *matching* entre um currículo e uma vaga.
- **Appendant Feature Exchange for Job Screening Features (AEJS):** esse método troca cada *Appendant feature* por uma *Screening feature* semanticamente semelhante a removida. Para isso, usamos *embeddings* treinados com o conjunto de habilidades que aparecem apenas nas descrições de vagas. A intuição por trás dessa estratégia é que a substituição de uma *Appendant feature* por uma *Screening feature* semanticamente próxima e presente nas vagas tende a melhorar a descrição de um currículo.
- **Appendant Feature Exchange for Screening Features Contextualized (AESC):** esse método troca cada *Appendant feature* por uma *Screening feature* semanticamente semelhante ao contexto do CV/vaga de forma geral. Nesse caso, todo o currículo/vaga é mapeado no espaço dimensional das representações do *embedding* e calcula-se as *features* mais semelhantes ao currículo/vaga analisado. O *embedding* é treinado com um conjunto completo de habilidades presentes nos currículos/vagas. A intuição por trás dessa estratégia é que a adição de habilidades contextualizadas ao CV/vaga pode melhorar sua descrição, mantendo uma maior consistência com a descrição original.
- **Appendant Feature Exchange for Job Screening Features Contextualized (AEJSC):** esse método é uma combinação dos métodos AEJS e AESC, pois troca cada *Appendant feature* por uma *Screening feature* semanticamente semelhante ao contexto do CV/vaga. Para isso, é treinado um *embedding* usando o conjunto de habilidades presentes nas descrições de vagas, bem como no AEJS. A definição de *features* semelhantes é feita com base em toda a descrição do currículo/vaga, assim como no AESC. Essa estratégia visa inserir as *features* mais relevantes que estejam presentes em Vagas e, além disso, consistentes com a descrição do CV/vaga alvo.

5 Avaliação Experimental

Para avaliar os métodos propostos, utilizamos uma amostragem de dados da Catho com 376,762 CVs e 115,955 vagas relacionadas a um período de 10 meses (janeiro a outubro). A Catho faz parte do grupo Seek[3] e é líder em recrutamento on-line na América Latina.

5.1 Identificação de Itens Sofrendo com o MaSP e Appedant Features

Selecionamos os oito primeiros meses (janeiro a agosto) como conjunto de treino. Definimos os *thresholds* T e T_M, usados para definir a escassez e o MaSP, como um mês. Aplicamos a estratégia presente na Seção 4.1 para identificar itens sofrendo do MaSP no conjunto de treino, encontrando o valor 3 para μ e o valor 0,6 para τ. Por fim, com os valores de μ e τ, aplicamos o Classificador Binário de Escassez no conjunto de testes.

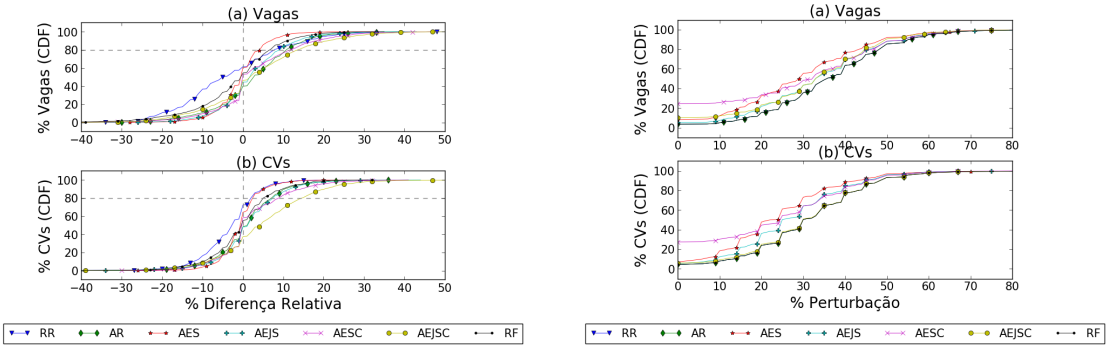
Na segunda etapa, identificamos as *Appendant Features* no conjunto de treino usando o Classificador de *Features* (com um *thesaurus* privado mantido pela Catho). Verificamos que há 20% mais *Appedant features* em CVs e vagas que sofrem com o MaSP do que nos demais CVs/vagas da amostra analisada.

5.2 Aplicação de Operações de Edição na Descrição de Itens

5.2.1 Avaliação das Estratégias Simulamos a aplicação das edições nas descrições de cada CV/vaga, da seguinte forma:

1 Para cada CV/vaga de teste classificado com o MaSP, calculamos a média da distância semântica para os três *matchings* mais próximos no conjunto de treino;
2 Aplicamos as estratégias para mitigar o MaSP nos CVs/vagas de teste;
3 Para cada CV/vaga modificado por nossas estratégias, calculamos a média da distância semântica para os três *matchings* mais próximos no conjunto de treino;
4 Comparamos a média das distâncias semânticas para os três *matchings* mais próximos de antes e depois da aplicação das estratégias, de acordo com as métricas abaixo:

[3] A maior empresa no setor de recrutamento no mundo (www.seek.com.au).



(a) Distância Relativa **(b) Perturbação**

Figure 1: Distância Relativa e Perturbação alcançadas pelas estratégias.

Métrica 1 - Distância Relativa: Tem como objetivo entender a porcentagem de melhoria (ou piora) que ocorreu em cada CV e vaga de trabalho.

$$\frac{mediaDistanciaDepois - mediaDistanciaAntes}{mediaDistanciaAntes} * 100 \quad (1)$$

Métrica 2 - Perturbação: O objetivo é entender o quanto cada estratégia precisa modificar os CVs e vagas para conseguir melhorias.

$$1 - \frac{numeroHabilidadesInalteradas}{numeroTodasHabilidades} \quad (2)$$

5.2.2 Linhas de Base Como esse trabalho é o primeiro que lida com o MaSP, não identificamos linhas de base diretamente aplicáveis para solucioná-lo. Assim, adaptamos métodos de *Feature Selection* para lidar parcialmente com esse problema novo da seguinte forma:

- **Random Feature Removal (RR):** remove aleatoriamente N_i *features* para cada item de teste i (CV/job). A proposta dessa linha de base é demonstrar que uma remoção de recursos sem critérios não leva ao aprimoramento das descrições.
- **Random Forest (RF):** remove *features* baseando-se no seus *score* ao classificar CVs e vagas (usando um algoritmo Random Forest) como sofrendo ou não com o MaSP. Removemos de cada item i do teste as N_i *features* menos discriminativas.

5.2.3 Resultados A Figura 1 (a) mostra os resultados sobre Distância Relativa para vagas de trabalho e CVs. Para todas as estratégias, pelo menos 65% das descrições das vagas obtiveram melhoria (i.e., apresentaram valores positivos para a Distância Relativa). A estratégia AEJSC exibiu o melhor comportamento e os seus ganhos representaram uma redução de 56% de vagas sofrendo com o MaSP. Considerando os CVs, a AEJSC, apresentou novamente os melhores resultados, com 75% das descrições alcançando melhora e trazendo uma redução de 53% dos CVs sofrendo com o MaSP. Considerando a perturbação na Figura 1 (b) e os resultados anteriores, a solução com os maiores ganhos (AEJSC) foi a terceira/quarta a causar menos perturbação nas descrições de vagas/CVs. Como o objetivo dessas estratégias é ajudar usuários no cadastro de CVs/vagas com melhores descrições, dado o potencial de melhora, a AEJSC é a ferramenta assistiva mais promissora.

6 Conclusões & Trabalhos Futuros

Nesse trabalho, formalizamos o ***Matching Scarcity Problem*** **(MaSP)**, problema no qual candidatos e recrutadores sofrem com a falta de oportunidades de *matching* em sistemas de recrutamento on-line, e propomos estratégias para identificar automaticamente candidatos e vagas sofrendo com ele. Propomos cinco estratégias heurísticas para mitigar o MaSP, as quais realizam alterações nas descrições dos CVs e vagas para aproximar candidatos sofrendo com o MaSP de vagas semanticamente relacionadas a eles, e vice-versa. A melhor estratégia reduziu em 50% o número de CVs e vagas que sofrem do MaSP. Como trabalho futuro, pretendemos analizar descrições de vagas e CVs separadamente por área profissional, visto que algumas áreas sofrem mais com o MaSP do que outras.

7 Acknowledgments

This work is supported by SEEK, Catho Online, CNPq and FAPEMIG.

References

[1] Alan Cardoso, Fernando Mourão, and Leonardo Rocha. 2019. A Characterization Methodology for Candidates and Recruiters Interaction in Online Recruitment Services. In *Proceedings of the 25th WebMedia '19*. 8 pages. https://doi.org/10.1145/3323503.3349541

[2] Alan Cardoso, Fernando Mourão, and Leonardo Rocha. 2020. Mitigating Matching Scarcity in Recruitment Recommendation Domains. In *Proceedings of the 26th WebMedia '20*. https://doi.org/10.1145/3428658.3430968

[3] Alan Cardoso, Fernando Mourão, and Leonardo Rocha. 2021. The Matching Scarcity Problem: When recommenders do not connect the edges in recruitment services. *Expert Systems with Applications* (2021). https://doi.org/10.1016/j.eswa.2021.114764

[4] Diego Carvalho, Nícollas Silva, Alan Cardoso, Elverton Fazzion, Adriano C. M. Pereira, and Leonardo Rocha. [n.d.]. Understanding Users-Contents Interaction in Non-Linear Multimedia Streaming Services. In *Proceedings of the 24th WebMedia 2018*. 229–232. https://doi.org/10.1145/3243082.3264661

[5] Papiya Das, Kashyap Barua, Manjusha Pandey, and Siddharth Swarup Routaray. 2018. Context Level Entity Extraction Using Text Analytics with Big Data Tools. In *IEMIS 2018*.

[6] Arthur P. Dempster. 2008. *A Generalization of Bayesian Inference*. Springer Berlin Heidelberg, Berlin, Heidelberg, 73–104. https://doi.org/10.1007/978-3-540-44792-4_4

[7] Maryam Fazel-Zarandi and Mark Fox. 2009. Semantic Matchmaking for Job Recruitment: An Ontology-Based Hybrid Approach. *International Semantic Web Conference* (01 2009).

[8] Yao Lu, Sandy El Helou, and Denis Gillet. 2013. A Recommender System for Job Seeking and Recruiting Website. In *Proceedings of the WWW 2013*.

[9] J. Malinowski, T. Keim, O. Wendt, and T. Weitzel. 2006. Matching People and Jobs: A Bilateral Recommendation Approach. In *HICSS'06*.

[10] XIAO-LI MENG and DONALD B. RUBIN. 1993. Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika* (1993). https://doi.org/10.1093/biomet/80.2.267

[11] Ioannis Paparrizos, B. Barla Cambazoglu, and Aristides Gionis. 2011. Machine Learned Job Recommendation. In *Proc. of the ACM RecSys 2011*.

[12] Luiz Pizzato, Tomek Rej, Thomas Chung, Irena Koprinska, and Judy Kay. 2010. RECON: A Reciprocal Recommender for Online Dating. In *Proc. of the ACM RecSys 2010*.