

Profile Matcher

Ferramenta de auxílio na procura de emprego e seleção de candidatos

José Cardoso^{1,2}[0000–0003–4252–5449] and Nuno Escudeiro^{1,3}

¹ Instituto Superior de Engenharia do Porto (ISEP), Porto
Rua Dr. António Bernardino de Almeida 431, 4200-072 Porto, Portugal

<http://www.dei.isep.ipp.pt>

² 1140466@isep.ipp.pt

³ nfe@isep.ipp.pt

Resumo A seleção de candidatos a um emprego é na sua essência um processo de ajustamento de um currículo a um dado perfil descrito pela oferta de emprego. É um processo moroso e oneroso que frequentemente exige a análise e avaliação individual e manual de centenas de currículos. Este mesmo cenário, ajustamento de competências a um dado perfil profissional, verifica-se também, embora em circunstâncias substancialmente distintas, quando um estudante pretende definir o seu percurso académico direcionando-o para uma determinada profissão.

A utilização de ferramentas automáticas pode trazer benefícios, como maior celeridade e eficiência, a estes processos. Este artigo retrata o trabalho desenvolvido na implementação da aplicação Profile Matcher, uma aplicação web que permite criar *templates* de emprego ou ocupação profissional e compará-los automaticamente com um conjunto de currículos com vista a identificar os melhores candidatos para uma oferta de emprego ou as melhores ofertas de emprego para um dado currículo fornecendo indicações adicionais sobre as competências que um dado estudante deve desenvolver para um melhor posicionamento face a uma dada profissão. Espera-se que esta aplicação permita guiar o percurso académico de um estudante assim como diminuir o tempo dos processos de recrutamento.

Palavras-chave: Ferramentas automáticas, Seleção de candidatos, Estudantes, Aprendizagem automática, Rede neuronal, Doc2Vec

1 Introdução

O Curriculum Vitae (CV) é provavelmente a ferramenta mais utilizada por recém-licenciados e jovens para expressar as suas competências, quando procuram emprego ou estágios. Um CV com as competências necessárias para uma oferta de emprego é um fator crucial para entrar no mercado laboral.

Atualmente, as empresas cada vez mais necessitam de contratar pessoas, o mais rapidamente e eficazmente possível. Por isso, o tempo e a eficiência no processo de seleção são fatores bastante importantes. Não é viável que um currículo

cumpra os requisitos para todas as oportunidades de emprego. Idealmente, um CV deve ser adaptado de forma a servir especificamente para cada oferta de emprego que o estudante se candidate.

Por outro lado, os recrutadores investem tempo e recursos valiosos no processo de recrutamento. A análise dos currículos dos candidatos, de modo a perceber o quão apropriados são para uma determinada oferta de emprego é uma tarefa crucial, mas trabalhosos e que consomem tempo.

De modo a satisfazer as necessidades de ambas as partes — dos recrutadores e de quem procura emprego, seria interessante a utilização de ferramentas automáticas. Seria assim possível identificar quais as competências mais relevantes para um emprego, para alguém que procure emprego. Os recrutadores poderiam também comparar as competências de candidatos mais rapidamente.

Decidiu-se desenvolver uma aplicação com o objetivo de facilitar o processo de seleção, tanto para os candidatos a emprego, como para os recrutadores, através do uso de aprendizagem automática. A facilitação na inserção de dados, como a extração de dados de um currículo de um candidato, foi um dos requisitos abordados.

Pretende-se com esta aplicação, identificar o quão adequado um candidato é para uma determinada oferta de emprego. Para tal, a aplicação deve obter dados relevantes do perfil do candidato, assim como de uma determinada oferta de emprego, de modo que seja possível aplicar um algoritmo adequado para fazer o “matching” entre ambos.

Neste artigo, são descritas as metodologias aplicadas e os resultados obtidos durante a análise e implementação da aplicação Profile Matcher.

2 Metodologias aplicadas

Esta secção refere o trabalho realizado para a implementação de um algoritmo, que seja capaz de fazer o “matching” entre um candidato a emprego e uma oferta de emprego.

2.1 Modelo de emprego

Existe uma grande diversidade na estrutura e na informação que é colocada nas ofertas de emprego, assim como nos currículos dos candidatos. Isto dificulta o processo de analisar a informação fornecida pelo utilizador. Com isto em conta, decidiu-se moldar um modelo genérico que abstraia essa informação. Este modelo teria várias categorias (e.g. Experiência profissional, educação, etc.) e subcategorias de dados considerados relevantes, e que habitualmente são publicadas nas ofertas de emprego, assim como nos currículos de candidatos.

Este modelo deve ser o mais genérico possível, de modo a ser adaptável para diversas áreas de negócio diferentes. Assim sendo, e após a análise de várias ofertas de emprego e de currículos, ficou definido o modelo representado na tabela seguinte.

Tabela 1. Modelo de emprego

Categorias	Subcategorias
Work experience	Field Duration Role Responsibilities/tasks
Education	Study Field Year Degree level
Hard Skills	Knowledge on Experience with Communication skills
Soft Skills	Team skills Leadership skills Organizational skills Interpersonal skills
Certifications	Field

Decidiu-se definir as categorias e subcategorias em inglês, visto os currículos e ofertas de emprego analisados estarem em inglês, assim como o Profile Matcher.

Desta forma, as características do candidato e da oferta de emprego seriam adaptadas conforme este modelo, o que facilitaria a identificação do quão adequado é um candidato a uma oferta de emprego.

2.2 Técnicas de processamento de linguagens naturais

Após a definição do modelo descrito na secção anterior, pretende-se comparar duas instâncias desse modelo, uma instância para um candidato e outra para uma oferta de emprego, e concluir o quão semelhantes são.

Uma possível solução seria comparar as categorias de ambas as instâncias, frase a frase e calcular a frequência de frases ou de palavras que fossem iguais. No entanto, esta hipótese não seria viável, pois duas palavras que fossem sinónimas ou semelhantes, seriam identificadas como se fossem totalmente diferentes.

Uma solução mais adequada, seria ter em conta o quão semelhantes são as frases e assim, calcular a média do grau de semelhança de todos os pares de frases. Um valor mais elevado significaria que o candidato tinha maior probabilidade de ser adequado para uma oferta de emprego.

Para tal, decidiu-se utilizar uma técnica de processamento de linguagem natural, chamada Doc2Vec, baseada noutra técnica Word2Vec, ambos criados por equipas de investigação da Google [1].

A técnica Word2Vec consiste em gerar vetores que representem palavras. O objetivo é gerar estes vetores, de modo que a distância entre eles, represente a semelhança das palavras [2]. Por exemplo, um vetor que represente a palavra “man” deverá ter uma distância relativamente curta a outro vetor que represente

a palavra “woman”. Por outro lado, um vetor “king”, deverá estar mais distante de um vetor “woman” e, ao mesmo tempo próximo de “man”. A Fig. 1 demonstra este exemplo numa representação gráfica com três dimensões.

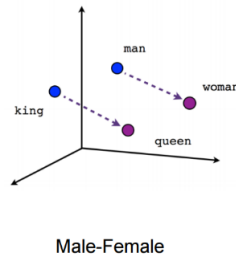


Fig. 1. Exemplo gráfico de Word2Vec

A técnica Doc2Vec é semelhante ao Word2Vec. Para além dos vetores que representam cada uma das palavras de uma frase, nesta técnica também é considerado mais um vetor. Este vetor representa uma identificação única do parágrafo onde se insere a frase [3].

Não se sabe inicialmente o quão semelhantes as palavras são. É por isso necessário treinar previamente um modelo de Doc2Vec através de uma rede neuronal. Pretende-se com este treino ajustar as posições dos vetores, de modo que a distância vetorial entre eles indiquem a semelhança das palavras que esses mesmos vetores representam.

Este treino ao modelo de Doc2Vec é feito fornecendo uma lista de frases, relacionadas com ofertas de emprego, visto neste caso em particular, fazer mais sentido descobrir relações entre palavras deste tema. Foi aplicado o Skip-Gram Model [4]. Este algoritmo, dada uma palavra de uma frase, tenta prever as palavras vizinhas na mesma frase. Isto significa que o modelo vai ser ajustado de modo que esta previsão seja o mais correta possível.

Após este treino, é possível comparar pares de frases e identificar o quão semelhantes elas são. Pode-se assim aplicar a solução sugerida, de calcular a média do grau de semelhança de todos os pares de frases.

2.3 Preenchimento automático de perfis

O Profile Matcher permite, quando possível, extrair informação de um CV, e preencher o perfil do candidato automaticamente. Posteriormente, o candidato pode adicionar ou editar as características do perfil que entender manualmente.

Existem ferramentas capazes de fazer a extração de CV em certas categorias definidas. Foram utilizadas algumas dessas ferramentas para implementar parcialmente esta funcionalidade. No entanto, estas categorias não correspondem exatamente às categorias definidas no Modelo de emprego (Tabela 1).

Esta secção demonstra como foi implementado o preenchimento automático de perfis.

O preenchimento automático de um perfil é feito em três passos:

- Extrair o texto de um currículo em formato pdf.
- Dividir e organizar porções do texto em categorias distintas.
- Adaptar as categorias definidas no modelo previamente definido (Tabela 1)

Extração de texto Em primeiro lugar, o currículo fornecido pelo utilizador (em formato pdf), é lido com o auxílio da ferramenta *npm-pdfreader* [5]. Esta ferramenta lê o texto do documento, linha a linha, assim como a posição de cada palavra lida.

Divisão do texto Depois de obtido o texto do documento, o mesmo é dividido em diversas categorias (e.g. competências, informação pessoal, etc.). Este passo foi feito com recurso a técnicas de aprendizagem automática. Adicionalmente, foi aproveitado o facto de existirem ferramentas *open-source* que resolvem parcialmente este problema.

Foram utilizadas as ferramentas *ResumeParser* [6] e *Pyresparser* [7]. Ambas transformam o conteúdo de um currículo, num objeto JSON, dividido em várias categorias. As categorias mais relevantes dessas ferramentas para o *Profile Matcher* são: *gender*, *email*, *work experience*, *skills* e *education and training*.

Algumas das categorias referidas coincidem com as do modelo descrito na Fig. 1. Nestes casos, as características relativas a estas categorias podem ser atribuídas diretamente no modelo. Noutros casos, é necessário fazer algumas adaptações após esta extração, de modo a poderem ser utilizadas no modelo.

Adaptação das categorias Resta adaptar as categorias que não são equivalentes ao modelo. Um dos casos é a categoria *skills*, que deve ser dividida nas subcategorias de *hard skills* e *soft skills* (categorias que existem no modelo). Para identificar qual a categoria que uma determinada característica pertence, foram obtidas algumas das características mais utilizadas em cada uma das subcategorias de *hard skills* e *soft skills* [8][9]. A tabela seguinte contém um excerto das características recolhidas, para a subcategoria *communication skills*.

O objetivo é comparar o grau de semelhança de cada uma destas características, de todas as subcategorias referidas, com cada uma das características relativas à categoria *skills* obtidas no processo de divisão de texto. O grau de semelhança entre as características é feito com o modelo *Doc2Vec* referido na secção anterior.

A categoria da característica que tenha um maior grau de semelhança, é assumida ser a categoria mais adequada. Por exemplo, se uma dada característica obtida no processo de divisão de texto tiver um maior grau de semelhança relativamente a uma característica da subcategoria *leadership skills*, então assume-se que essa característica pertença a essa subcategoria.

Tabela 2. Excerto de características (communication skills)

Answering Telephones
Business Correspondence
Calling Clients
Client Relations
Communication
Correspondence
Customer Service
Directing Clients
(...)

2.4 Redes neuronais

As redes neuronais contêm camadas. Cada uma das camadas contêm nós, que têm um peso atribuído a eles. Um peso maior tem mais influência em ativar nós das camadas seguintes. Os pesos dos nós vão sendo modificados consoante os *inputs* e *outputs* fornecidos na fase de treino [10].

Quando o *output* fornecido é semelhante ao *output* calculado pela rede neuronal, os nós com pesos maiores tendem a não se alterar. Por outro lado, quando o *output* fornecido não corresponde ao calculado, os pesos tendem a ser alterados. Isto significa que a rede neuronal é ajustada de modo que os nós tenham pesos mais eficazes para calcular o *output*.

Está em progresso o desenvolvimento de uma abordagem alternativa. Esta abordagem consiste em treinar uma rede neuronal que seja capaz de classificar o texto dado. Para tal, seria utilizada uma parte das características recolhidas como treino e a parte restante como teste. Espera-se que esta abordagem tenha resultados mais fiáveis que a primeira.

3 Profile Matcher

O Profile Matcher é a aplicação Web que é usada diretamente pelos utilizadores. Esta aplicação foi implementada em React.js, e tira partido do trabalho descrito nas secções anteriores. A Fig. 2 demonstra um excerto da aplicação, após o preenchimento automático do perfil de um utilizador.

The image shows a web form for a profile. It is divided into three main sections: Soft Skills, Personal Information, and Hard Skills.

Soft Skills Section: This section has a header "Soft Skills" and a dropdown menu. Below the dropdown are four categories of skills, each with a list of specific skills in a box:

- Team Skills:** Banking
- Leadership Skills:** Analysis, Engineering, Presentations
- Organizational Skills:** Financial statements
- Interpersonal Skills:** Research

There are "Edit" and "Remove" buttons at the top right of this section.

Personal Information Section: This section has a header "Personal Information". It contains four fields:

- Age:** (empty)
- Naturality:** (empty)
- Gender:** male
- E-mail address:** adityabhartia@yahoo.com

Hard Skills Section: This section has a header "Hard Skills" and an "Edit" button. It contains one category of skills with a list of specific skills in a box:

- Knowledge on:** Writing, Forecasts, Accounting, Reports

Fig. 2. Excerto do Perfil após o preenchimento automático

Com o fim de concluir de uma forma objetiva o quão fiável é o algoritmo desenvolvido, foi abordado um método de avaliação do mesmo. Dada uma amostra de CV de candidatos que foram aceites num emprego, assim como o cargo desse mesmo emprego, verifica-se se a aplicação aconselha ao candidato ofertas de emprego deste mesmo cargo. Se a aplicação aconselhar ao candidato outro tipo de ofertas de emprego, assume-se que é um falso positivo.

Com esta abordagem, e com uma amostra de 191 CV, concluiu-se que a taxa de resultados positivos no "matching" de candidatos com ofertas de emprego é de 59%. Um valor que, neste momento, não deve ser suficiente para ser utilizado viavelmente no mercado de trabalho.

4 Conclusões

O Profile Matcher facilita a procura de ofertas de emprego ou de candidatos adequados, tanto para os recrutadores ou para os candidatos. Facilita também o preenchimento de perfis ao fornecer um currículo em formato pdf. O desenvolvimento desta aplicação utilizou algumas ferramentas ou técnicas previamente desenvolvidas, em conjunto com novas técnicas aplicadas especificamente para o Profile Matcher. Esta versatilidade no uso de técnicas revelou-se eficaz, pois permitiu tirar proveito de cada um dos seus propósitos em diversas funções da aplicação.

A eficácia desta aplicação está em grande parte dependente da qualidade e quantidade de dados nos treinos de modelos de aprendizagem automática. Uma melhoria nestes fatores permitirá que a aplicação tenha resultados mais precisos, o que significa que a aplicação está sempre sujeita a melhorias no futuro.

Em suma, o Profile Matcher serve um propósito relevante para o mercado de trabalho e apesar de poder ainda ser melhorado, é uma ferramenta que poderá vir a ser útil para candidatos a emprego, assim como para recrutadores.

Referências

1. Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space, 2013; arXiv:1301.3781.
2. Emeric Dymant, Stéfan J. Darmoni, Émeline Lejeune, Gaëtan Kerdelhué, Jean-Philippe Leroy, Vincent Lequertier, Stéphane Canu and Julien Grosjean. Doc2Vec on the PubMed corpus: study of a new approach to generate related articles, 2019; arXiv:1911.11698.
3. Jey Han Lau and Timothy Baldwin. An Empirical Evaluation of doc2vec with Practical Insights into Document Embedding Generation, 2016, Proceedings of the 1st Workshop on Representation Learning for NLP, Berlin, Germany, pp. 78–86; arXiv:1607.05368.
4. Xuedong Huang and Fileno Alleva and Hsiao-wuen Hon and Mei-yuh Hwang and Ronald Rosenfeld. The SPHINX-II Speech Recognition System: An Overview, 1992.
5. adrienjoly/npm-pdfreader: Read text and parse tables from PDF files. Includes automatic column detection, and rule-based parsing., <https://github.com/adrienjoly/npm-pdfreader> Visitado em Junho de 2020
6. antonydeepak/ResumeParser: Resume Parser using rule based approach. Developed using framework provided by GATE, <https://github.com/antonydeepak/ResumeParser> Visitado em Julho de 2020
7. OmkarPathak/pyrespaser: A simple resume parser used for extracting information from resumes, <https://github.com/OmkarPathak/pyrespaser> Visitado em Julho de 2020
8. Pesquisa grátis de ofertas de emprego — Indeed <https://pt.indeed.com/?r=us> Visitado em Julho de 2020
9. Sample Job Descriptions <https://hiring.monster.com/employer-resources/job-description-templates/sample-job-descriptions/> Visitado em Julho de 2020
10. Hopfield JJ. Neural networks and physical systems with emergent collective computational abilities. Proc Natl Acad Sci U S A. 1982;79(8):2554-2558. doi:10.1073/pnas.79.8.2554