Análise Comparativa entre Abordagens de Aprendizado de Máquina para Classificação Automática de Currículos de Profissionais de TIC

Renato Santos Pereira[1] **, Hilário Tomaz Alves de Oliveira**[1]

[1]Programa de Pós-graduação em Computação Aplicada (PPComp)
Instituto Federal do Espírito Santo (IFES), Serra, Brasil

renato_stosp@gmail.com, hilario.oliveira@ifes.edu.br

Abstract. *Resume Screening plays a crucial role in recruiting talent in companies. However, dealing with a large volume of resumes can be time-consuming and complex. In order to automate this task, several works have explored natural language processing techniques and machine learning algorithms. In this context, this paper presents a comparative analysis of different approaches for automatically classifying resumes of Information and Communication Technology (ICT) professionals. The investigated approaches include traditional machine learning algorithms, models based on deep neural networks, and pre-trained neural language models. Experiments were conducted using a set of 27,405 resumes divided into eight categories related to ICT professionals. The results show that, in general, the pre-trained models achieved the best performances, especially the RoBERTa-base model, which obtained a performance superior to 93.00% in all the evaluation measures used.*

Resumo. *A triagem de currículos desempenha um papel crucial no recrutamento de talentos nas empresas. Contudo, lidar com um grande volume de currículos pode ser demorado e complexo. Com o objetivo de automatizar essa tarefa, diversos trabalhos têm explorado técnicas de processamento de linguagem natural e algoritmos de aprendizado de máquina. Nesse contexto, este trabalho apresenta uma análise comparativa de diferentes abordagens para a classificação automática de currículos de profissionais de Tecnologia da Informação e Comunicação (TIC). As abordagens investigadas incluem algoritmos tradicionais, modelos baseados em redes neurais profundas e modelos neurais de linguagem pré-treinados. Foram realizados experimentos utilizando um conjunto de 27.405 currículos, distribuídos em oito categorias relacionadas aos profissionais de TIC. Os resultados obtidos revelam que, de maneira geral, os modelos pré-treinados alcançaram os melhores desempenhos, especialmente, o modelo RoBERTa-base, que obteve resultados superiores a 93,00% em todas as medidas de avaliação utilizadas.*

1. Introdução

As atividades relacionadas à Tecnologia da Informação e Comunicação (TIC) continuam em constante crescimento, contribuindo direta e indiretamente para a criação de diversos postos de trabalho [Silveira and Tonini 2021]. O setor de TIC emprega diferentes categorias de profissionais, cada um com sua própria escolaridade, conhecimentos técnicos, grau

de instrução acadêmica e habilidades cognitivas. De acordo com um estudo realizado pela Associação Brasileira de Empresas de Tecnologia da Informação e Comunicação (Brasscom[1]), estima-se que as empresas de tecnologia necessitem de 797 mil profissionais qualificados entre 2021 e 2025. Em geral, a demanda por profissionais no campo de TIC requer conhecimentos variados, abrangendo áreas administrativas, habilidades linguísticas, engenharias, contabilidade, direito, entre outras [Jorge and Costa 2022].

Identificar competências e habilidades requeridas a um profissional é um processo complexo e desafiador, especialmente na área de TIC, porque além das profissões deste segmento serem relativamente novas, estas ainda carecem de uma regulamentação em países como o Brasil [Silveira and Tonini 2021]. Para obter Recursos Humanos (RH) adequados e de qualidade, um dos processos mais importantes em uma empresa é o recrutamento, que consiste na seleção de potenciais colaboradores. A aquisição de talentos é um processo dinâmico, complexo e um aspecto importante da gestão de RH nas empresas [Roopesh and Babu 2021], e de acordo com um levantamento realizado pela divisão de pesquisas da empresa Glassdoor[2], este processo dura em torno de 40 dias no Brasil [Chamberlain 2017]. Ao longo do tempo, o processo de recrutamento tem passado por diversas transformações, sendo a etapa inicial geralmente a triagem dos currículos. Essa triagem pode ser realizada de diferentes formas, desde a análise de candidaturas em papel até a utilização de formulários *online* [Najjar et al. 2021, Ransing et al. 2021]. No contexto atual, em que há uma ampla diversidade de funções de trabalho e um grande número de candidaturas recebidas, tem se tornado cada vez mais difícil realizar o processo de triagem manual de currículos de maneira rápida e eficiente [Cabrera-Diego et al. 2019].

Diante desse cenário, é possível identificar na literatura uma variedade de trabalhos que buscam desenvolver abordagens para a automação da triagem de currículos, utilizando técnicas de Inteligência Artificial (IA), como Processamento de Linguagem Natural (PLN) e Aprendizado de Máquina (AM). Alguns desses trabalhos, como [Satheesh et al. 2020, Najjar et al. 2021, Ransing et al. 2021], abordam esse tema atribuindo uma pontuação aos currículos com base em uma descrição de vaga fornecida pelo sistema, utilizando algoritmos de AM. Outras pesquisas, como em [Roopesh and Babu 2021, Rajath et al. 2021, Ali et al. 2022], exploram algoritmos de AM para classificação, buscando determinar se um currículo possui ou não os elementos necessários que o credenciam para uma determinada vaga.

Este trabalho tem como objetivo contribuir nesse contexto, por meio de uma análise comparativa entre diferentes abordagens de AM, na tarefa de triagem automática de currículos de profissionais da área de TIC. A análise realizada abrange desde algoritmos tradicionais, como Árvores de Decisão e Regressão Logística, até algoritmos de aprendizado profundo e modelos neurais de linguagem pré-treinados, como o *Bidirecional Encoder Representations from Transformers* (BERT) [Devlin et al. 2019]. Foram realizados diversos experimentos utilizando uma base de dados composta por 27.405 currículos, classificados em 9 categorias de trabalho. Os resultados experimentais demonstram que, de maneira geral, os modelos pré-treinados baseados no BERT apresentaram os melhores desempenhos. Em maior destaque, o modelo *RoBERTa-base* treinado por 10 épocas alcançou uma assertividade superior a 93,00% em todas as medidas de avaliação utiliza-

[1]https://brasscom.org.br/

[2]https://www.glassdoor.com/research/

das, incluindo Acurácia, Precisão, Cobertura e Medida-F.

2. Trabalhos Relacionados

Os primeiros sistemas de triagem de currículos foram criados no fim da década de 1990, como uma solução autossuficiente para empresas que procuravam uma forma de se concentrar em seus negócios principais [Sinha et al. 2021]. Dentre esses sistemas, pode-se destacar os sistemas criados pelas empresas Sovren[3] em 1996 e TextKernel, em 2001 [Tosik 2014, Buttiker et al. 2021].

Nos anos 2000, os Modelos Ocultos de Markov, do inglês *Hidden Markov Models* (HMMs), eram amplamente utilizados como método de extração de informação [Hetzner 2008]. Um dos trabalhos que aplicaram essa técnica foi o desenvolvido por [Yu et al. 2005], que propôs um modelo híbrido que empregava HMM em conjunto com o algoritmo de Máquinas de Vetores de Suporte, do inglês *Support Vector Machines* (SVM) para extração de informações presentes nos currículos. Posteriormente, o trabalho de [Singh et al. 2010] propôs um sistema denominado PROSPECT, que utilizava os Campos Aleatórios Condicionais, do inglês *Conditional Random Fields* (CRF) para extrair aspectos relevantes dos perfis dos candidatos, como habilidades, experiência em cada habilidade, detalhes educacionais e experiência anterior.

A partir de 2010, começaram a surgir trabalhos na área de triagem de currículos, cujo motor de extração de informações era baseado no conceito de Web Semântica. Dentre eles, é válido citar a pesquisa de [Çelik and Elçi 2013] que propôs um sistema de extração de informações orientados a ontologias, visando converter os currículos para uma versão estruturada e enriquecida semanticamente, a fim de utilizá-los em processos de recrutamento. Além desse trabalho, destaca-se a pesquisa de [Kumaran and Sankar 2013], que propôs um sistema que faz a correspondência entre os requisitos para uma determinada vaga e as habilidades dos candidatos expressas nos currículos utilizando mapeamento de ontologias. Outro trabalho nessa mesma linha de pesquisa foi desenvolvido por [Silva et al. 2018], que propôs um sistema que possibilita anotar automaticamente entidades nos Currículo Lattes[4] de pesquisadores por meio de bases de dados abertas (*Linked Open Data*).

Nos últimos anos, as técnicas de classificação de texto baseadas em algoritmos de Aprendizado de Máquina (AM) têm sido amplamente utilizadas em diversos domínios [Ali et al. 2022]. Nesse contexto, o trabalho de [Gopalakrishna and Vijayaraghavan 2019] propôs um sistema de classificação de currículos que utiliza um conjunto de seis algoritmos tradicionais de AM (Naive Bayes, Naive Bayes multinomial, Bernoulli Naive Bayes, Máquinas de Vetores de Suporte, Regressão Logística e K-vizinhos Mais Próximos). Outro trabalho que utiliza algoritmos de AM nesse contexto é o desenvolvido por [Ali et al. 2022], que compara o desempenho de nove algoritmos para classificação de currículos.

Embora os algoritmos tradicionais de AM tenham alcançado sucesso em várias tarefas importantes de PLN, eles enfrentam dificuldades em abranger todas as regularidades linguísticas por meio do projeto manual de atributos com conhecimento especializado no

[3]https://www.textkernel.com/textkernel-acquires-sovren

[4]https://lattes.cnpq.br/

domínio [Deng and Liu 2018]. Nesse contexto, os algoritmos de aprendizagem profunda, do inglês *Deep Learning* (DL), têm sido usados para preencher essa lacuna. Dentre as técnicas baseadas em DL, pode-se citar a utilização de arquiteturas baseadas em redes neurais recorrentes, como a rede *Long Short-Term Memory* (LSTM), as redes convolucionais, e mais recentemente nos modelos neurais de linguagem pré-treinados, como o *Bidirectional Encoder Representations from Transformer* (BERT) [Li et al. 2022].

Dentre os estudos que fazem uso de tais técnicas, destaca-se o trabalho apresentado por [Jiechieu and Tsopze 2021], que propôs um modelo de arquitetura de classificação multi-rótulo baseado em redes neurais convolucionais para identificar profissões a partir do conjunto de habilidades expressas em currículos em formato de texto não estruturado. Um outro trabalho que utiliza algoritmos baseados em redes neurais foi apresentado por [Bhatia et al. 2019], onde foi proposto um sistema que utiliza o modelo BERT na classificação e adaptação de currículos para o formato utilizado na plataforma LinkedIn[5].

Diante do cenário apresentado, este trabalho se diferencia dos anteriores por apresentar uma análise comparativa entre o desempenho de diferentes abordagens de classificação, levando em consideração abordagens baseadas em algoritmos de AM tradicionais em conjunto com a clássica medida de *Term Frequency–Inverse Document Frequency* (TF-IDF), representações multidimensionais, algoritmos baseados em redes neurais profundas e modelos neurais de linguagem pré-treinados.

3. Materiais e Métodos

3.1. Base de Dados

A base de dados utilizada neste trabalho foi proposta por [Jiechieu and Tsopze 2021] e está disponível publicamente na plataforma GitHub[6]. Essa base de dados é composta de 28.707 currículos públicos e anonimizados, coletados a partir do motor de busca de empregos Indeed[7]. Os currículos estão classificados em dez categorias de empregos, e cada currículo pode ser categorizado em mais de uma classe. Além disso, os currículos variam em tamanho, indo de uma até seis páginas. Todos os currículos pertencem ao domínio de Tecnologia da Informação (TI) e estão relacionados a habilidades como "Desenvolvedor(a) de *software*", "Desenvolvedor(a) *front-end*", "Desenvolvedor(a) Python", "Desenvolvedor(a) Java", "Desenvolvedor(a) *Web*, "Gerente de projetos", "Administrador(a) de rede", "Administrador(a) de banco de dados", "Administrador(a) de sistemas" e "Analista de segurança".

Após uma análise inicial da base de dados, foi identificado que alguns currículos não possuíam uma categoria atribuída, além de terem sido encontradas duplicatas. Para garantir a qualidade dos dados, foi necessário remover os currículos duplicados e aqueles sem categoria definida. Além disso, neste estudo, considerou-se apenas uma única categoria atribuída a cada currículo. Por essa razão, a categoria "Desenvolvedor(a) *front-end*" não foi incluída nos experimentos. Após esse processo de filtragem, a base de dados utilizada neste trabalho consiste em 27.405 currículos, agrupados em nove categorias.

[5]https://www.linkedin.com/

[6]https://github.com/florex/resume_corpus

[7]https://www.indeed.com/

Na Figura 1 é apresentada a distribuição do número de currículos por categoria. Conforme observado no gráfico de barras, a classe com o maior número de currículos é "Desenvolvedor(a) de *Software*" com 5.377 exemplos, enquanto a categoria "Administrador(a) de rede" possui a menor quantidade de exemplos, com 2.145 currículos. De maneira geral, a base de dados possui um número de currículos por classe balanceado.

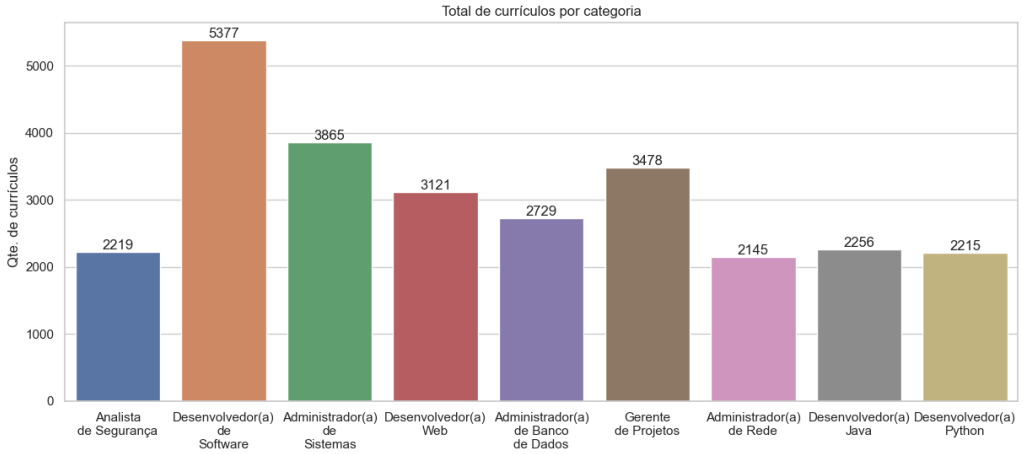

Figura 1. Quantidade de currículos por categoria.

A Figura 2 ilustra a distribuição do número de palavras por currículo, agrupados por categoria. É possível perceber que existe uma uniformidade nas distribuições de palavras por currículo em cada uma das categorias. Embora os currículos tenham um tamanho predefinido, destaca-se uma tendência de maior extensão nos currículos das classes "Desenvolvedor(a) Java" e "Desenvolvedor(a) Python".

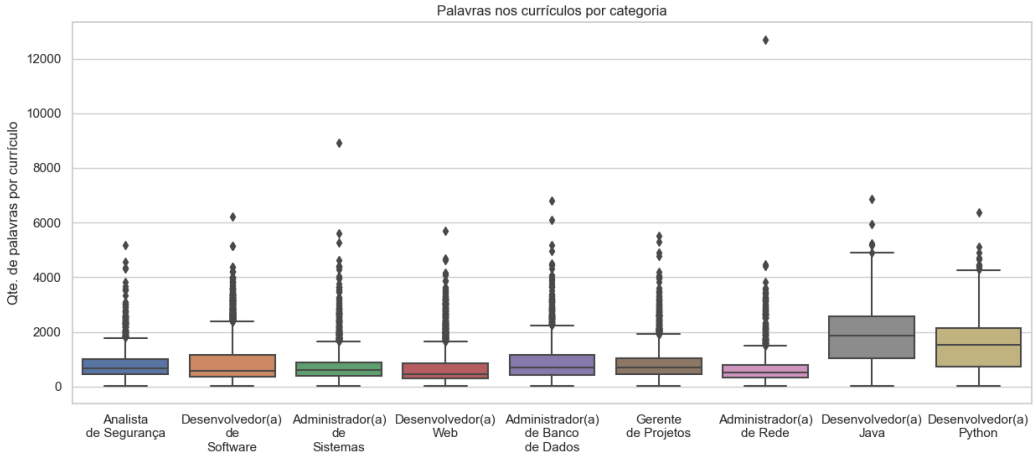

Figura 2. Distribuição da quantidade de palavras por currículo, agrupada por categoria.

3.2. Abordagens Avaliadas

As abordagens avaliadas neste trabalho podem ser categorizados em quatro perspectivas distintas: **(i)** a aplicação de algoritmos de AM tradicionais em conjunto com a medida

Term Frequency-Inverse Document Frequency (TF-IDF); **(ii)** o uso de algoritmos de AM tradicionais em conjunto com representações contextuais multidimensionais (*contextual word embeddings*); **(iii)** a utilização de redes neurais convolucionais e redes neurais recorrentes; e **(iv)** a execução de modelos neurais de linguagem pré-treinados baseados no BERT.

Na primeira abordagem, foram aplicados algoritmos tradicionais de AM para a classificação dos currículos. Inicialmente, os textos dos currículos passaram por um processo de pré-processamento para remover símbolos de pontuação e palavras classificadas como *stop words*[8]. Em seguida, foi construído um vocabulário contendo as cinco mil palavras mais frequentes com base no conjunto de treinamento. Para representar os currículos numericamente, utilizou-se a técnica de TF-IDF, que atribui pesos às palavras com base em sua frequência nos currículos e sua raridade no corpus geral. Essa abordagem permite capturar a importância das palavras em cada currículo [Ali et al. 2022].

Na segunda abordagem, também foram utilizados algoritmos tradicionais de AM, mas em conjunto com uma representação contextual multidimensional dos currículos. As representações multidimensionais são um tipo de codificação numérica que busca capturar as relações semânticas e sintáticas entre as palavras de um texto. Essa técnica permite que palavras com significados relacionados tenham representações semelhantes no espaço vetorial, facilitando a compreensão das relações e significados contextuais das palavras. Essa abordagem é especialmente útil para modelos de aprendizado mais complexos, incluindo aqueles baseados em aprendizado profundo [Neelima and Mehrotra 2023].

Na terceira abordagem, foram utilizados algoritmos de aprendizado profundo baseados em redes neurais convolucionais, do inglês *Convolutional Neural Network* (CNN), e a rede neural do tipo *Long Short-Term Memory* (LSTM). As CNNs têm a vantagem de extrair características dominantes dos dados processados, atuando como um seletor de palavras relevantes. Por outro lado, as LSTMs são capazes de aprender as características linguísticas e padrões ao analisar sequencialmente os textos de entrada [Patil et al. 2023]. Neste trabalho, foi adotado um modelo de classificação baseado em CNNs, seguindo uma arquitetura similar à investigada por [Jiechieu and Tsopze 2021]. Além desse modelo, também foi utilizado uma arquitetura híbrida composta por uma CNN, seguida por de uma rede LSTM bidirecional (BiLSTM). Essa abordagem visa tirar proveito das vantagens distintas de cada tipo de rede. A CNN tem o papel de selecionar as características dominantes presentes nos currículos, enquanto a BiLSTM é responsável por capturar os padrões presentes nas sequências de palavras, considerando a dependência temporal das palavras tanto à esquerda quanto à direita. A ideia é que essa análise contextualizada permita uma compreensão mais abrangente dos currículos.

Por fim, na quarta abordagem, foram empregados modelos neurais de linguagem baseados no BERT. Os modelos de linguagem pré-treinados têm se mostrado eficazes para melhorar o desempenho de diversas tarefas de PLN, reduzindo a necessidade de desenvolver arquiteturas específicas para cada tarefa [Devlin et al. 2019]. O BERT realiza um pré-treinamento com representações bidirecionais profundas, levando em consideração o contexto tanto à esquerda quanto à direita em todas as camadas. Como resultado, as representações pré-treinadas do BERT podem ser ajustadas com apenas uma

[8]São palavras muito frequentes em um idioma, como artigos, preposições, entre outras.

camada densa adicional de saída, tornando possível criar modelos capazes de desempenhar uma tarefa específica sem a necessidade de modificações substanciais na arquitetura [Devlin et al. 2019]. Neste trabalho, foram utilizados os seguintes modelos: **(i)** O *BERT-base* original proposto por [Devlin et al. 2019]; **(ii)** Uma versão compactada do modelo *BERT-base*, chamada de *DistilBERT-base* [Sanh et al. 2019]; **(iii)** O modelo *ALBERT-base* proposto por [Lan et al. 2019]; e **(iv)** O *Robustly Optimized BERT Pretraining Approach* (RoBERTA) base [Liu et al. 2019].

4. Experimentos

4.1. Configurações dos Experimentos

Todas as implementações utilizadas para executar os experimentos deste trabalho foram desenvolvidas na linguagem de programação Python e estão disponíveis em um repositório da plataforma GitHub[9]. O fluxo de trabalho adotado seguiu a mesma metodologia para as quatro abordagens analisadas, envolvendo a aplicação de algoritmos tradicionais de AM com as representações baseadas na medida TF-IDF e as representações multidimensionais, o uso de algoritmos de aprendizado profundo e a utilização de modelos neurais de linguagem pré-treinados baseados no BERT.

Além das etapas de filtragem mencionadas na Seção 3.1, os textos dos currículos foram pré-processados para a remoção de marcações na linguagem *HyperText Markup Language* (HTML), substituição de espaços em branco desnecessários, eliminação de menções a endereços de páginas *web*, substituição de quebras de linha e fragmentação do texto em palavras. Essas etapas foram realizadas utilizando expressões regulares e a ferramenta spaCy[10].

Foi adotada a validação cruzada estratificada com cinco subconjuntos como metodologia de avaliação para todos os algoritmos analisados. Em cada execução, foi separado 10% do conjunto de treinamento como conjunto de validação. Para avaliar o desempenho dos algoritmos, foram computadas as seguintes medidas: Acurácia e as médias macro das métricas de Precisão, Cobertura e Medida-F. Por fim, foram computadas as médias dos valores obtidos pelos algoritmos avaliados nas cinco execuções da metodologia de validação cruzada.

Foram avaliados dez algoritmos de AM tradicionais em conjunto com as abordagens utilizando TF-IDF e representações multidimensionais. Os seguintes algoritmos, disponíveis na biblioteca Scikit-learn[11], foram utilizados: Árvore de Decisão, *Extremely Randomized Trees* (*Extra Trees*), Floresta Aleatória, K-Vizinhos mais Próximos, do inglês *K-Nearest Neighbors* (KNN), Máquinas de Vetores de Suporte, do inglês *Support Vector Machine* (SVM), Perceptron de Múltiplas Camadas, do inglês *Multilayer Perceptron* (MLP) e Regressão Logística. Além desses, também foram considerados nos experimentos os seguintes algoritmos: XGBoost[12], CatBoost[13] e LightGBM[14]. Todos os algoritmos foram utilizados com seus parâmetros padrões definidos nas respectivas bibliotecas. Essa

[9]https://github.com/renatostosp/mpca_triagem_curriculo
[10]https://spacy.io/
[11]https://scikit-learn.org/
[12]https://github.com/dmlc/xgboost/
[13]https://catboost.ai/
[14]https://github.com/Microsoft/LightGBM/

seleção abrangente de algoritmos busca explorar diferentes técnicas para encontrar a melhor solução para a classificação dos currículos.

As representações contextuais multidimensionais foram obtidas utilizando a ferramenta *Sentence Transformers* [Reimers and Gurevych 2019], usando o modelo pré-treinado *all-MiniLM-L6-v2*[15]. Esse modelo é capaz de mapear frases para um espaço vetorial denso de 384 dimensões. Sua utilização tem sido amplamente adotada em diversas tarefas de PLN, como agrupamento de textos e recuperação semântica de documentos.

As implementações das redes CNN e BiLSTM na abordagem baseada em aprendizado profundo foram desenvolvidas utilizando a biblioteca Keras[16]. Cada modelo foi treinado por 20 épocas durante as etapas da validação cruzada. Ao final de cada época, o modelo resultante foi aplicado ao conjunto de validação. Somente o modelo com o melhor desempenho no conjunto de validação, considerando a Medida-F, foi mantido. Ao concluir as 20 épocas de treinamento, apenas o modelo com melhor desempenho salvo foi utilizado para avaliação no conjunto de testes.

Por fim, na abordagem baseada nos modelos pré-treinados do BERT, foram utilizados os modelos disponibilizados na plataforma Hugging Face[17]. A implementação desses modelos foi realizada utilizando a biblioteca *Transformers*[18]. Foram realizados treinamentos com três configurações diferentes para o número total de épocas: uma, cinco e dez. O objetivo foi avaliar se um treinamento mais longo resultaria em melhor desempenho. Assim como na abordagem anterior, ao final de cada época, o modelo resultante é aplicado no conjunto de validação e somente aquele com melhor desempenho com base na Medida-F é mantido. Os modelos pré-treinados utilizados foram os seguintes: **(i)** BERT-base[19]; **(ii)** DistilBERT-base[20]; **(iii)** ALBERT-base[21]; e **(iv)** RoBERTA-base[22].

4.2. Resultados

Na Tabela 1, são apresentados os resultados dos experimentos utilizando a abordagem baseada na representação usando a medida TF-IDF. O algoritmo *LightGBM* demonstrou o melhor desempenho em quase todas as métricas de avaliação (Acurácia, Cobertura e Medida-F), exceto pela medida de Precisão, em que o algoritmo *XGBoost* obteve o melhor resultado. De maneira geral, esses dois algoritmos apresentaram desempenhos muito similares. É importante ressaltar que mesmo o algoritmo com pior desempenho (KNN) alcançou uma Medida-F próxima a 70,00%. Esses resultados evidenciam que, apesar da simplicidade da abordagem utilizando TF-IDF, ela apresenta um desempenho competitivo como um *baseline*.

Os resultados dos experimentos utilizando a abordagem baseada nas representações multidimensionais são apresentados na Tabela 2. O algoritmo SVM demonstrou o melhor desempenho em três das quatro medidas de avaliação (Acurácia, Cobertura e Medida-F), enquanto o algoritmo *CatBoost* obteve o melhor valor na medida

[15]https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2
[16]https://keras.io/
[17]https://huggingface.co/
[18]https://huggingface.co/docs/transformers/index
[19]https://huggingface.co/bert-base-uncased
[20]https://huggingface.co/distilbert-base-uncased
[21]https://huggingface.co/albert-base-v2
[22]https://huggingface.co/roberta-base

Tabela 1. Resultados do experimentos (%) utilizando a abordagem baseada na medida TF-IDF. O melhor resultado com base em cada medida de avaliação é destacado em negrito.

Algoritmos	Acurácia	Precisão	Cobertura	Medida-F
Árvore de Decisão	75,38	76,18	76,04	76,10
CatBoost	84,13	84,60	84,54	84,53
Extra Trees	76,57	80,18	75,43	77,03
Floresta Aleatória	78,96	82,00	78,21	79,57
KNN	67,82	68,02	70,16	68,86
LightGBM	**84,96**	85,01	**85,82**	**85,37**
MLP	77,73	78,97	78,11	78,50
Regressão Logística	82,14	81,95	83,71	82,69
SVM	82,11	82,03	83,49	82,67
XGBoost	84,80	**85,31**	85,20	85,22

de Precisão. É importante destacar que os resultados alcançados usando as representações multidimensionais extraídas do modelo *all-MiniLM-L6-v2* foram inferiores aos obtidos usando a medida TF-IDF. No entanto, vale ressaltar que, na representação TF-IDF, cada currículo é representado por cinco mil palavras, enquanto na representação multidimensional são utilizadas apenas 384 palavras, resultando em uma taxa de compressão mais de dez vezes menor. Dessa forma, informações importantes dos currículos podem ter sido desconsideradas. Apesar dessa codificação reduzida, o algoritmo SVM apresentou um desempenho superior a 81,00% considerando a Medida-F.

Tabela 2. Resultados do experimentos (%) utilizando a abordagem baseada em representações multidimensionais. O melhor resultado com base em cada medida de avaliação é destacado em negrito.

Algoritmos	Acurácia	Precisão	Cobertura	Medida-F
Árvore de Decisão	57,665	59,085	58,928	58,980
CatBoost	79,734	**80,921**	80,004	80,392
Extra Trees	75,428	79,553	74,725	76,265
Floresta Aleatória	75,603	79,782	74,881	76,523
KNN	75,202	75,356	76,628	75,839
LightGBM	79,256	80,204	79,809	79,986
MLP	78,847	79,337	79,525	79,395
Regressão Logística	79,610	79,444	81,468	80,276
SVM	**80,558**	80,323	**82,169**	**81,098**
XGBoost	79,142	80,746	79,222	79,892

A Tabela 3 apresenta os resultados dos experimentos com os modelos neurais de

linguagem pré-treinados, utilizando três configurações diferentes no número de épocas de treinamento. Destaca-se que o modelo *RoBERTa-base* obteve as melhores métricas entre todos os modelos analisados, alcançando valores superiores a 93,00% em todas as medidas de avaliação. O modelo treinado por dez épocas apresentou melhores resultados nas medidas de Acurácia, Precisão e Medida-F, enquanto o modelo treinado por cinco épocas demonstrou melhor desempenho na medida de Cobertura. Com exceção do *RoBERTa-base*, foi observado que os demais modelos (*BERT-base*, *ALBERT-base* e *DistilBERT-base*) obtiveram melhor desempenho quando treinados por cinco épocas. Isso sugere que o treinamento por 10 épocas pode resultar em um possível sobreajuste desses modelos aos dados de treinamento, prejudicando o desempenho na etapa de teste. Mesmo o modelo *ALBERT-base* apresentando o pior resultado, ele ainda foi capaz de atingir valores superiores a 90,00% em todas as métricas de avaliação.

Tabela 3. Resultados (%) dos experimentos utilizando os modelos neurais de linguagem pré-treinados. O melhor resultado com base em cada medida de avaliação é destacado em negrito.

Modelos	Épocas	Acurácia	Precisão	Cobertura	Medida-F
ALBERT-base	1	91,823	91,894	91,993	91,900
	5	93,275	93,449	93,322	93,369
	10	92,399	92,658	92,367	92,492
BERT-base	1	92,906	93,117	93,036	93,051
	5	93,552	93,604	93,721	93,645
	10	93,370	93,442	93,401	93,406
DistilBERT-base	1	93,012	93,209	93,109	93,139
	5	93,592	93,737	93,665	93,684
	10	93,490	93,535	93,577	93,543
RoBERTa-base	1	92,695	92,874	92,869	92,842
	5	93,914	93,913	**94,035**	93,951
	10	**93,943**	**94,181**	93,949	**94,051**

Na Tabela 4, são apresentados os resultados da análise comparativa entre os algoritmos que apresentaram melhor desempenho utilizando a medida TF-IDF, as representações multidimensionais e os modelos neurais pré-treinados. Além disso, são incluídos os resultados obtidos utilizando duas arquiteturas adicionais: uma rede do tipo CNN, similar à utilizada em [Jiechieu and Tsopze 2021], e um modelo híbrido que combina uma camada de rede CNN com uma camada de rede do tipo BiLSTM. Observa-se que o modelo *RoBERTa-base* manteve o melhor desempenho em todas as medidas de avaliação, quando comparado a todos os algoritmos analisados neste trabalho. Essa superioridade pode ser atribuída à sua arquitetura mais complexa e ao processo de pré-treinamento em larga escala, que envolve o uso de grandes volumes de dados textuais. Como resultado, o *RoBERTa-base*, assim como os outros modelos baseados no BERT, é capaz de identificar padrões mais complexos no significado das palavras com base em seu contexto, contribuindo para um melhor desempenho na classificação dos currículos.

Os modelos baseados nas arquiteturas *CNN* e *CNN+BiLSTM* demonstraram desempenhos superiores em relação aos algoritmos que utilizam a medida TF-IDF e as representações multidimensionais (*embeddings*). Além disso, é importante ressaltar o desempenho aprimorado do modelo *CNN+BiLSTM* em comparação ao modelo que utiliza apenas a *CNN*. Acreditamos que a combinação da camada convolucional e das recorrentes em uma única arquitetura possibilitou que o modelo filtrasse as palavras mais relevantes dos currículos por meio da convolução, ao mesmo tempo em que representava informações sequenciais e de longo alcance por meio da recorrência. Essa combinação permitiu obter um desempenho mais eficaz na tarefa de classificação dos currículos, aproveitando as características complementares desses dois tipos de redes neurais.

Tabela 4. Resultados (%) da análise de desempenho entre os melhores algoritmos nas abordagens TF-IDF, representações multidimensionais e modelos pré-treinados, em comparação com as arquiteturas baseadas nas redes CNN e LSTM. O melhor resultado para cada medida de avaliação está destacado em negrito.

Modelos	Acurácia	Precisão	Cobertura	Medida-F
CNN	90,078	90,374	90,197	90,183
CNN+BiLSTM	92,169	92,114	92,322	92,169
LightGBM + TF-IDF	84,963	85,006	85,820	85,372
RoBERTa-base	**93,943**	**94,181**	**93,949**	**94,051**
SVM + *Embeddings*	80,558	80,323	82,169	81,098

Na Figura 3, é apresentada a matriz de confusão com todas as predições realizadas pelo modelo *RoBERTa-base* após dez épocas de treinamento. É possível observar que existe uma maior incidência de falsos positivos na categoria "Desenvolvedor(a) de Software", identificada pelo Código 6 na matriz de confusão. Esses falsos positivos contribuem para uma degradação das métricas de avaliação adotadas nas diferentes abordagens analisadas. Esse comportamento pode ser atribuído à semelhança existente entre os currículos de candidatos dessa categoria e os de outras classes que são, na verdade, especializações do posto de desenvolvedor(a) de *software*, como "Desenvolvedor(a) Java" (Código 1), "Desenvolvedor(a) Python" (Código 4) e "Desenvolvedor(a) Web" (Código 8). Por outro lado, para as demais categorias, o *RoBERTa-base* apresentou uma baixa ocorrência de erros, o que demonstra que o modelo foi capaz de extrair padrões eficientes para diferenciar os currículos de cada uma das classes de trabalho.

5. Considerações Finais e Trabalhos Futuros

Neste trabalho, foi realizada uma análise comparativa entre diferentes abordagens de Aprendizado de Máquina para a triagem automática de currículos no domínio de TIC. As abordagens investigadas incluem a representação tradicional usando a medida TF-IDF, a aplicação de representações contextuais multidimensionais, o uso das arquiteturas CNN e BiLSTM baseadas em redes neurais profundas, e, por fim, a utilização de modelos neurais de linguagem pré-treinados baseados no BERT. Foram conduzidos diversos experimentos utilizando um conjunto de 27.405 currículos, contendo nove categorias de trabalho distintas. Os resultados experimentais obtidos revelaram que os modelos pré-treinados obtive-

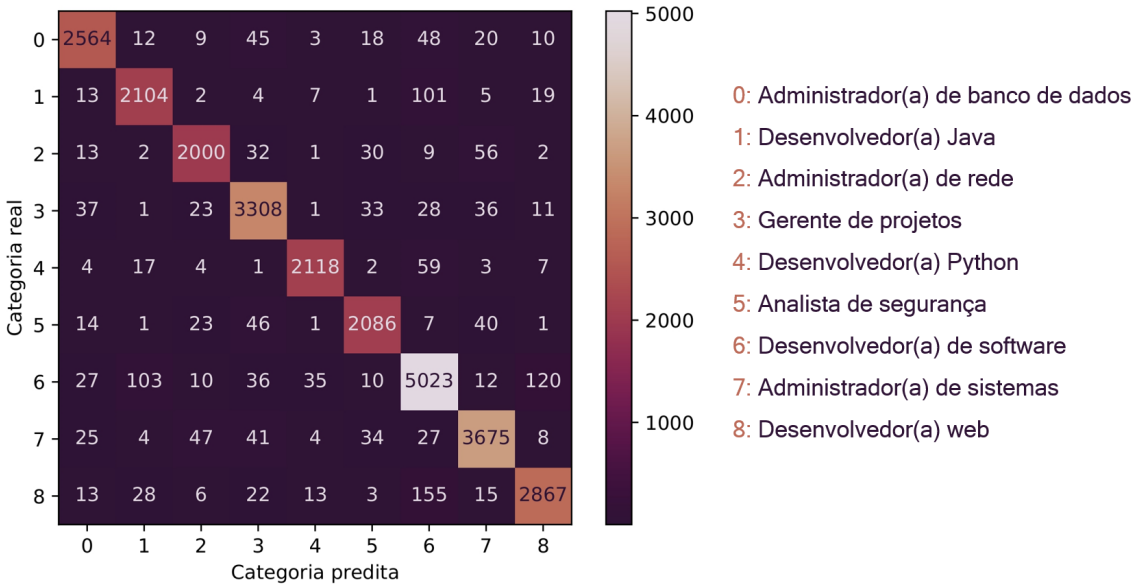

Figura 3. Matriz de confusão do modelo RoBERTA.

ram o melhor desempenho de forma geral, com destaque para o modelo *RoBERTa-base*, que alcançou valores superiores a 93,00% em todas as medidas de avaliação utilizadas.

Como trabalhos futuros, pretendemos investigar a aplicação de técnicas de Processamento de Linguagem Natural e Aprendizado de Máquina para identificar habilidades técnicas (*hard skills*) e interpessoais (*soft skills*) presentes nos currículos, assim como o estudo proposto por [Fareri et al. 2021], que utilizou o reconhecimento de entidades nomeadas para essa tarefa. Além disso, vislumbramos a expansão dos experimentos realizados para incluir a utilização de algoritmos de classificação multirrótulo. Essa abordagem possibilitará a atribuição de múltiplas categorias às amostras de currículos, tornando a classificação mais abrangente e precisa.

Por fim, outra linha de pesquisa de interesse é a investigação de vieses, como o de gênero, presentes nas representações multidimensionais, conforme abordado em [Caliskan et al. 2022]. Em estudos futuros, é importante explorar de forma mais aprofundada como esses vieses podem impactar a classificação de currículos, pois as representações de palavras influenciadas por vieses podem introduzir tendências indesejadas nos resultados. Compreender a extensão desses vieses e como eles podem afetar a seleção de candidatos é essencial para garantir a imparcialidade e a igualdade de oportunidades no processo de recrutamento.

Agradecimentos

Os autores agradecem ao Ifes, apoio da FAPES e CAPES (processo 2021-2S6CD, nº FAPES 132/2021) por meio do PDPG (Programa de Desenvolvimento da Pós-Graduação, Parcerias Estratégicas nos Estados).

Referências

Ali, I., Mughal, N., Khan, Z. H., Ahmed, J., and Mujtaba, G. (2022). Resume Classification System using Natural Language Processing and Machine Learning Techniques.

Mehran University Research Journal of Engineering and Technology, 41(1):65–79.

Bhatia, V., Rawat, P., Kumar, A., and Shah, R. R. (2019). End-to-End Resume Parsing and Finding Candidates for a Job Description using BERT.

Buttiker, F., Roth, S., Steinacher, T., and Hanne, T. (2021). Comparative analysis of tools for matching work-related skill profiles with cv data and other unstructured data. *University of South Florida (USF) M3 Publishing*, 5(2021):97.

Cabrera-Diego, L. A., El-Bèze, M., Torres-Moreno, J.-M., and Durette, B. (2019). Ranking résumés automatically using only résumés: A method free of job offers. *Expert Systems with Applications*, 123:91–107.

Caliskan, A., Ajay, P. P., Charlesworth, T., Wolfe, R., and Banaji, M. R. (2022). Gender bias in word embeddings. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*. ACM.

Çelik, D. and Elçi, A. (2013). An ontology-based information extraction approach for résumés. In *Pervasive Computing and the Networked World: Joint International Conference, ICPCA/SWS 2012, Istanbul, Turkey, November 28-30, 2012, Revised Selected Papers*, pages 165–179. Springer.

Chamberlain, A. (2017). How long does it take to hire? interview duration in 25 countries. *Retrieved from Glassdoor. com website: https://www. glassdoor. com/research/time-to-hire-in-25-countries*.

Deng, L. and Liu, Y. (2018). A joint introduction to natural language processing and to deep learning. *Deep learning in natural language processing*, pages 1–22.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

Fareri, S., Melluso, N., Chiarello, F., and Fantoni, G. (2021). Skillner: Mining and mapping soft skills from any text. *Expert Systems with Applications*, 184:115544.

Gopalakrishna, S. T. and Vijayaraghavan, V. (2019). Automated tool for resume classification using sementic analysis. *International Journal of Artificial Intelligence and Applications (IJAIA)*, 10(1).

Hetzner, E. (2008). A simple method for citation metadata extraction using hidden markov models. In *Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '08, page 280–284, New York, NY, USA. Association for Computing Machinery.

Jiechieu, K. F. F. and Tsopze, N. (2021). Skills prediction based on multi-label resume classification using cnn with model predictions explanation. *Neural Computing and Applications*, 33:5069–5087.

Jorge, T. V. and Costa, E. C. D. (2022). Análise das modalidades de contratações CLT E PJ para os profissionais de Tecnologia da Informação. *Revista Interface Tecnológica*, 18(2):91–104.

Kumaran, V. S. and Sankar, A. (2013). Towards an automated system for intelligent screening of candidates for recruitment using ontology mapping (EXPERT). *International Journal of Metadata, Semantics and Ontologies*, 8(1):56.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P. S., and He, L. (2022). A survey on text classification: From traditional to deep learning. *ACM Trans. Intell. Syst. Technol.*, 13(2).

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Najjar, A., Amro, B., and Macedo, M. (2021). An intelligent decision support system for recruitment: resumes screening and applicants ranking. *Informatica*, 45(4).

Neelima, A. and Mehrotra, S. (2023). A comprehensive review on word embedding techniques. In *2023 International Conference on Intelligent Systems for Communication, IoT and Security (ICISCoIS)*, pages 538–543.

Patil, P., Raul, S., Raut, D., and Nagarhalli, T. (2023). Hate speech detection using deep learning and text analysis. In *2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 322–330.

Rajath, V., Fareed, R. T., and Kaganurmath, S. (2021). Resume Classification and Ranking using KNN and Cosine Similarity. *IJERTV10IS080057*, 10(08).

Ransing, R., Mohan, A., Emberi, N. B., and Mahavarkar, K. (2021). Screening and Ranking Resumes using Stacked Model. In *2021 5th International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques (ICEECCOT)*, pages 643–648, Mysuru, India. IEEE.

Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Roopesh, N. and Babu, C. N. (2021). Robotic Process Automation for Resume Processing System. In *2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, pages 180–184, Bangalore, India. IEEE.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Satheesh, K., Jahnavi, A., Iswarya, L., Ayesha, K., Bhanusekhar, G., and Hanisha, K. (2020). Resume Ranking based on Job Description using SpaCy NER model.

Silva, W. D. d., Parreiras, F. S., Maia, L. C. G., and Brandão, W. C. (2018). Anotação semântica automática do currículo Lattes utilizando Linked Open Data. *Perspectivas em Ciência da Informação*, 23(4):53–72.

Silveira, A. C. J. d. and Tonini, A. M. (2021). Análise sobre a regulamentação do profissional do setor de tecnologia da informação e comunicação no Brasil. *Revista HISTEDBR On-line*, 21:e021022.

Singh, A., Rose, C., Visweswariah, K., Chenthamarakshan, V., and Kambhatla, N. (2010). PROSPECT: A System for Screening Candidates for Recruitment. In *Proceedings of the 19th ACM international conference on Information and knowledge management - CIKM '10*, page 659, New York, New York, USA. ACM Press.

Sinha, A. K., Amir Khusru Akhtar, M., and Kumar, A. (2021). Resume Screening Using Natural Language Processing and Machine Learning: A Systematic Review. In *Machine Learning and Information Processing: Proceedings of ICMLIP 2020*, pages 207–214.

Tosik, M. (2014). Internship report: Sequence labelling using distributional semantic vectors and conditional random fields.

Yu, K., Guan, G., and Zhou, M. (2005). Resume information extraction with cascaded hybrid model. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 499–506.