Desenvolvimento de uma Ferramenta para Reconhecimento de Entidades Nomeadas em Certificados de Atividades Complementares de Curso utilizando spaCy

Bernardo Gularte Kirsch[1], Árton Pereira Dorneles[1]

[1] Curso de Bacharelado em Ciência da Computação do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha (IFFar) – Frederico Westphalen – RS – Brasil

bernardogulartekirsch@gmail.com, arton.dorneles@iffarroupilha.edu.br

Abstract. *This work proposes the development of a Python tool to recognize entities named in certificates for complementary course activities. The tool was implemented with the help of the spaCy library and evaluated using a corpus of certificate data from courses in the IFFar/FW information and communication axis. After conducting computational experiments, the results obtained demonstrate through metrics that the proposed tool is promising for recognizing names, titles, periods and workload of a certain class of certificates for complementary course activities.*

Resumo. *Este trabalho propõe o desenvolvimento de uma ferramenta em Python para realizar o reconhecimento de entidades nomeadas de certificados de atividades complementares de curso. A ferramenta foi implementada com o auxílio da biblioteca spaCy e avaliada por meio de um corpus de dados de certificados provenientes dos cursos do eixo de informação e comunicação do IFFar/FW. Após a condução de experimentos computacionais, os resultados obtidos demonstram por meio de métricas que a ferramenta proposta é promissora para reconhecer nomes, títulos, períodos e carga horária de uma determinada classe de certificados de atividades complementares de curso.*

1. Introdução

Com o avanço da tecnologia e da internet, a quantidade de dados produzidos e armazenados nas mais diversas aplicações tem crescido significativamente nos últimos anos, envolvendo principalmente as mídias de texto, áudio, imagem e vídeo. Além do armazenamento desses formatos se constituir em um desafio por si só, existe ainda o desafio de processá-los para extrair informações relevantes para cada aplicação como, por exemplo, a identificação de pessoas, organizações, locais, data e hora presentes nos dados brutos de cada mídia. Esse processo de extração, quando realizado de forma manual por um operador humano, se torna oneroso e repetitivo, exigindo um significativo investimento de tempo, além de potencializar a ocorrência de erros.

Como uma proposta de solução para esse problema de extração de dados, foi apresentado em 1996, na 6ª edição do evento Message Understanding Conference (MUC-6), uma técnica de Processamento de Linguagem Natural denominada Reconhecimento de Entidades Nomeadas - REN (do inglês, Named Entity Recognition - NER) que, na ocasião, tinha como exemplo de aplicação principal a extração automática de informações relevantes em mensagens militares. Desde então, esta técnica tem sido utilizada para extração de dados nas mais diversas áreas. O processo de extração de dados utilizando REN consiste em uma implementação de software com o objetivo de identificar um conjunto específico de entidades

no texto como, por exemplo, nomes de pessoas, organizações, locais, data, hora, entre outras informações de interesse de cada aplicação. Para essa implementação, normalmente é utilizado um *corpus* de dados particular da aplicação para treinamento da ferramenta, buscando maior êxito e precisão no reconhecimento das entidades de interesse. Uma vez que a ferramenta tenha sido treinada, é possível utilizá-la para obter maior agilidade no processo de extração, otimizando o uso de tempo e de recursos de uma determinada aplicação.

Na literatura científica são encontrados vários trabalhos utilizando REN que buscam a extração automática de informações em diversos formatos de arquivos, mas, em particular, os arquivos do tipo PDF (*Portable Document Format*) se destacam como um dos principais formatos de interesse para extração de informações por ser muito utilizado para intercâmbio e disseminação de diversas fontes de informações na internet, como por exemplo, registros do Diário Oficial da União, documentos jurídicos, editais, notas fiscais eletrônicas, textos legislativos e documentos de coleta de dados como formulários e questionários. Para todas essas fontes é possível encontrar propostas correspondentes para extração de informações utilizando REN.

Arquivos no formato PDF também são muito utilizados para criar documentos de certificados usados para comprovar, por exemplo, a conclusão de um curso de curta duração ou participação em um evento científico. Esse tipo de certificado é bastante utilizado para validação de atividades complementares de cursos de instituições de ensino no Brasil e, até onde se sabe, não existe nenhuma proposta de ferramenta na literatura com o objetivo de extrair informações de forma automatizada de certificados armazenados em PDF.

Nesse contexto, este trabalho propõe uma ferramenta desenvolvida em Python para realizar o reconhecimento de entidades nomeadas de certificados de atividades complementares de curso utilizando como base de treinamento um *corpus* de dados de certificados provenientes dos cursos do eixo de informação e comunicação do IFFar/FW. Desta forma, espera-se que o desenvolvimento deste projeto possa contribuir não somente com o desenvolvimento científico na área de computação, mas também em fornecer uma ferramenta que possa ser utilizada para agilizar, facilitar e minimizar erros de entrada de dados em sistemas informatizados para validação e gerenciamento de atividades complementares de curso.

O restante do trabalho está organizado como segue. Na Seção 2 é apresentado o referencial teórico do trabalho. Na Seção 3, a metodologia do projeto da ferramenta proposta é detalhada. A Seção 4 apresenta experimentos computacionais e resultados, e por fim, na Seção 5, são apresentadas as considerações finais e opções de trabalhos futuros.

2. Referencial Teórico

Nesta seção são apresentados os principais conceitos e tecnologias necessárias para a compreensão deste trabalho, bem como um conjunto de trabalhos relacionados.

2.1. Reconhecimento de Entidades Nomeadas (REN)

De acordo com SILVA (2020), o Reconhecimento de Entidades Nomeadas (REN) é uma técnica de Processamento de Linguagem Natural (PLN) que surgiu em 1996, na 6ª edição do evento Message Understanding Conference (MUC-6), e teve seu foco na extração automática de informações em mensagens militares. Entidades nomeadas são artefatos de texto presentes em documentos descritos em linguagem natural. O processo de Reconhecimento de Entidades

Nomeadas busca identificar e classificar esses artefatos de acordo com as necessidades de cada aplicação. Um sistema de REN recebe como entrada um texto livre, não estruturado, e devolve como saída um conjunto de textos estruturados destacando as entidades de interesse.

As entidades nomeadas podem ser classificadas em categorias pré-definidas como Pessoa, Organização, Local, Data, Moeda, entre outras, esse tipo de entidade é chamado de entidade pré-construída (*Prebuilt Entity*). As entidades podem ter classificações específicas de uma determinada aplicação, mas para que isso seja possível é necessário que um algoritmo de aprendizado de máquina seja aplicado para treinamento prévio destas entidades.

Em resumo, um sistema de REN possui quatro etapas: pré-processamento, identificação de palavras pertinentes, classificação e pós-processamento. Na primeira etapa, a etapa de pré-processamento, o texto é corrigido e preparado para análise. Na etapa de identificação de palavras pertinentes, as palavras candidatas a entidades são identificadas. Na etapa de classificação, as palavras candidatas são classificadas em suas respectivas categorias e por fim, na última etapa, a etapa de pós-processamento, é onde as entidades são refinadas e, se necessário, agrupadas em entidades compostas ou relacionadas.

Atualmente, existem várias bibliotecas e sistemas de REN de uso geral que estão disponíveis para extração de dados em língua portuguesa e fornecem interfaces de programação em diversas linguagens de programação. Na linguagem Python, existe o sistema NERP-CRF, que possui código aberto e utiliza aprendizado de máquina. Em Java, foi proposto o LanguageTasks que é livre para fins acadêmicos e pago para fins comerciais, sendo capaz de reconhecer e classificar entidades por meio de um ambiente web. Outro sistema implementado em Java é o Apache OpenNLP que suporta tokenização e análise sintática. Desenvolvido em C++, o sistema FreeLing também possui código aberto e tem como foco o reconhecimento em textos na língua portuguesa, assim como o sistema 'Palavras' (FONSECA et. al, 2015). Outra opção disponível é o spaCy, uma biblioteca Python de processamento de linguagem natural de força industrial.

2.2. spaCy

O spaCy é uma biblioteca de código aberto desenvolvida em Cython que tem seu foco no uso em ambientes de produção, fornecendo um bom desempenho em tarefas de extração de informações de textos em grande escala. Esta biblioteca disponibiliza diversos *pipelines* pré-treinados para diferentes idiomas, incluindo o Português, fornecendo recursos adequados para produção e tokenização, bem como componentes para REN com fácil extensão e personalização, marcação de classes de gramática, segmentação de frases, classificação de texto, lematização, análise morfológica, vinculação de entidades e muitos outros. Além disso, também dispõe de suporte para modelos do PyTorch e TensorFlow.

A biblioteca utiliza modelos de *deep learning* e *machine learning* para Processamento de Linguagem Natural. Os modelos utilizados pelo spaCy são das mais diversas arquiteturas de redes neurais, como a arquitetura Transformer, Rede Neural Convolucional (CNN), Rede Neural Recorrente (RNN), Bidirectional Long Short-Term Memory (BiLSTM), entre outras. Para o Reconhecimento de Entidades Nomeadas, o componente de REN utiliza um modelo de rede neural convolucional que tem seu algoritmo baseado em transição que codifica suposições eficazes para tarefas usuais de REN, identificando extensões de tokens rotulados não sobrepostos. O componente visa oferecer um equilíbrio entre eficiência e precisão (SPACY, 2023).

2.3. Certificados de Atividades Complementares

Conforme SOUTHIER e DORNELES (2022) as atividades complementares de curso constituem um requisito obrigatório para a formação acadêmica e profissional de um estudante em diversas instituições de ensino do país. Especificamente, no Instituto Federal Farroupilha (IFFar), Campus Frederico Westphalen, as atividades podem ser realizadas em diversas categorias, como participação em eventos, realização de cursos de curta duração, estágios, bem como atividades de pesquisa, ensino e extensão. Além disso, cada curso da instituição define uma carga horária mínima de atividades que o estudante precisa cumprir e comprovar mediante a apresentação de certificados na coordenação de curso.

Ao receber um certificado, além do tipo de atividade, o coordenador de curso precisa extrair quatro informações principais: nome do estudante, título da atividade, período de realização e carga horária. Cada uma destas informações pode ser considerada uma entidade nomeada diferente. Na Figura 1 é apresentado um exemplo de um certificado onde estas quatro entidades estão destacadas em amarelo.


XI Encontro Anual de Tecnologia da Informação
VIII JASI - Jornada Acadêmica de Sistemas de Informação
de 25 a 28 de janeiro de 2021

CERTIFICADO

Certificamos para os devidos fins que Bernardo Gularte Kirsch participou do XI Encontro Anual de Tecnologia da Informação (EATI), ocorrido entre 25/01/2021 e 28/01/2021, perfazendo um total de 16 horas.

André Fiorin
Coordenação do Curso de Sistemas para Internet-IFFar

Solange Pertile
Coordenação do curso Sistemas de Informação-UFSM

Este certificado foi entregue a Bernardo Gularte Kirsch e registrado à fl: 2 do livro respectivo número 16 sob o número de registro 46.

Chave de Verificação: 8767.AC55.BA1Z.9Z89.72AC
Verificação: www2.fw.iffarroupilha.edu.br/autenticacao
Frederico Westphalen, 28 de janeiro de 2021.

Figura 1. Exemplo de um certificado de participação em evento

2.4. Trabalhos Relacionados

Até o momento, não há registro na literatura acadêmica a proposta de uma ferramenta que extrai informações relevantes de certificados de atividades complementares de curso, mas há trabalhos semelhantes que extraem informações de outras fontes. Como o trabalho desenvolvido por ALLES (2018) que teve como proposta a utilização de aprendizado supervisionado para extração de entidades nomeadas do Diário Oficial da União (DOU), onde é apresentado um estudo explorando os conceitos e aplicações de 4 ferramentas de

processamento de linguagem natural, a OpenNLP e a CoreNLP para reconhecimento de entidades nomeadas e a NLTK e a Syntaxnet para reconhecimento morfossintático.

Para explorar os conceitos de ferramentas que realizam PLN, o autor propõe uma metodologia que consiste na construção de um *corpus* específico para auxiliar no reconhecimento de entidades do DOU, realizado o seu processamento visando o entendimento linguístico das palavras em um texto, e em seguida, a quantidade de entidades presentes no texto é comparada com a quantidade de entidades reconhecidas e a qualidade das entidades reconhecidas é verificada. O *corpus* foi construído com a OpenNLP, utilizando aprendizado supervisionado, para que fosse elaborada uma proposta de construção de um *corpus* específico para extrair entidades nomeadas com melhor qualidade, comparando com os resultados dos *corpus* disponíveis. Os resultados quantitativos e qualitativos obtidos pelo DOU-Corpus se mostraram superiores em comparação com outras estratégias da literatura.

Outro trabalho é o desenvolvido por FONSECA et. al (2015), onde é apresentada a construção de um modelo para o reconhecimento de entidades nomeadas utilizando o NameFinder, uma classe contida no OpenNLP, que tem como objetivo reconhecer e classificar entidades nomeadas para o Português, dada a inexistência de um modelo para língua portuguesa no OpenNLP. Foram utilizados os *corpus* Amazônia e Harem para treinar e avaliar o modelo considerando 10 categorias: Pessoa, Local, Organização, Acontecimento, Obra, Abstração, Coisa, Tempo, Valor e Outro. Na avaliação do modelo é comparado o número de entidades anotadas do Harem com as encontradas e classificadas corretamente pelo modelo considerando as 10 categorias, e apresentando os resultados de *precisão, recall* e *f-measure* para cada uma. Também é apresentada uma comparação da *precisão, recall* e *f-measure* das categorias Pessoa, Local e Organização no OpenNLP em comparação com os resultados de outras ferramentas como NERP-CRF, LTASK, Freeling e PALAVRAS. Os resultados apresentados no trabalho são compatíveis com os demais modelos, podendo ser ligeiramente superior em alguns aspectos.

3. Metodologia

Esta seção apresenta detalhes sobre a metodologia utilizada para o projeto da ferramenta proposta, bem como delimita o escopo dos materiais de dados da pesquisa.

3.1. Delimitação de Escopo de Certificados e Entidades Nomeadas de Interesse

Devido a existência de inúmeros tipos de certificados de atividades complementares representados em formatos de mídia diversos, é necessário delimitar o escopo desta pesquisa. Desta forma, assume-se que um certificado de atividade complementar está representado no formato de um arquivo PDF e que as informações de interesse estão disponíveis em sua primeira página. Além disso, um certificado pode ter as informações organizadas em qualquer *layout*, mas devem estar disponíveis em algum objeto de texto no arquivo PDF. Essa delimitação implica que estão excluídos do escopo desta pesquisa certificados cujas informações estão representadas por meio de um objeto de imagem inserido no arquivo PDF.

Em relação às entidades nomeadas de interesse que precisam ser extraídas de um certificado, são consideradas quatro categorias: NOME, TITULO, CARGA e PERIODO. A categoria NOME especifica um ou mais nomes atribuídos a certificação da atividade realizada. A categoria TITULO se refere ao título da atividade, podendo ser o nome de um evento ou curso, por exemplo. A categoria CARGA se refere a carga horária da atividade. Por fim, a categoria PERIODO se refere ao período em que a atividade certificada foi realizada,

podendo ser, por exemplo, a data de conclusão ou as datas de início e final da atividade. No certificado da Figura 1, por exemplo, as entidades NOME, TITULO, CARGA e PERIODO correspondem, respectivamente a "Bernardo Gularte Kirsch", "XI Encontro Anual de Tecnologia da Informação (EATI)", "16 horas" e "25/01/2021 e 28/01/2021".

3.2. Procedimento para Converter um Certificado no Formato PDF em Texto

Para que seja possível extrair as entidades nomeadas de interesse de um certificado no formato PDF é necessário, primeiramente, convertê-lo em um formato de texto. Para realizar esse processo de conversão, utiliza-se o utilitário de linha de comando denominado Pdftotext, o qual vem incluído com a biblioteca Poppler de renderização de arquivos PDF (POPPLER, 2023). Após a conversão com o utilitário, o texto é tratado através da função apresentada na Figura 2. A função recebe por parâmetro o texto extraído pelo Pdftottext e retorna uma versão tratada. Este tratamento consiste na conversão em uma única linha por meio da remoção de quebras de linhas, remoção de espaços duplicados e substituição de aspas duplas por aspas simples.

```python
import re

def trata_texto_extraido(texto: str):
    texto = texto.replace('\n', ' ')
    texto = re.sub(r'\s+', ' ', texto)
    texto = texto.replace('"', "'")
    return texto
```

Figura 2. Exemplo do código da função de tratamento do texto extraído

Desta forma, a aplicação do procedimento descrito nesta seção no certificado da Figura 1, resulta no certificado em formato texto exibido na Figura 3.

```
CERTIFICADO Certificamos para os devidos fins que Bernardo Gularte Kirsch participou do XI Encontro Anual de Tecnologia da Informação (EATI), ocorrido entre 25/01/2021 e 28/01/2021, perfazendo um total de 16 horas. André Fiorin Coordenação do Curso de Sistemas para Internet-IFFar Solange Pertile Coordenação do curso Sistemas de Informação-UFSM Este certificado foi entregue a Bernardo Gularte Kirsch e registrado à fl: 2 do livro respectivo número 16 sob o número de registro 16. Chave de Verificação: 8767.AC55.BA1Z.9Z89.72AC Verificação: www2.fw.iffarroupilha.edu.br/autenticacao Frederico Westphalen, 28 de janeiro de 2021.
```

Figura 3. Exemplo de um certificado convertido em formato texto

3.3. Procedimento de Treinamento com a Biblioteca spaCy

A biblioteca spaCy fornece recursos para a criação de um modelo para extração de entidades nomeadas específicas da aplicação por meio de um processo de treinamento. Este processo requer a especificação das entidades nomeadas que serão aprendidas, um conjunto de dados de treinamento rotulados em uma lista Python seguindo uma estrutura específica do spaCy e a escolha de parâmetros para o treinamento. O processo de rotulação de dados envolve a

adequação do texto de cada certificado presente no conjunto de dados em uma estrutura que contém o texto do certificado e a posição inicial e final para cada entidade nomeada desejada identificada no texto fornecido. Um exemplo desta estrutura, obtido a partir dos dados de texto da Figura 3, é apresentado na Figura 4.

```
("CERTIFICADO Certificamos para os devidos fins que
Bernardo Gularte Kirsch participou do XI Encontro Anual
de Tecnologia da Informação (EATI), ocorrido entre
25/01/2021 e 28/01/2021, perfazendo um total de 16
horas. André Fiorin Coordenação do Curso de Sistemas
para Internet-IFFar Solange Pertile Coordenação do curso
Sistemas de Informação-UFSM Este certificado foi
entregue a Bernardo Gularte Kirsch e registrado à fl: 2
do livro respectivo número 16 sob o número de registro
16. Chave de Verificação: 8767.AC55.BA1Z.9Z89.72AC
Verificação: www2.fw.iffarroupilha.edu.br/autenticacao
Frederico Westphalen, 28 de janeiro de 2021.",
{"entities": [(50, 73, "NOME"), (88, 140, "TITULO"),
(205, 213, "CARGA"), (157, 180, "PERIODO")]})
```

Figura 4. Exemplo de uma estrutura de rotulação de dados utilizada no treinamento do modelo

Depois de realizar a configuração das etiquetas das entidades de interesse e a rotulação dos dados, o procedimento de treino pode ser iniciado. Durante cada época, os dados de treinamento são embaralhados visando evitar padrões de aprendizado. Após o fim do treinamento, o modelo treinado é salvo em um arquivo e pode ser invocado para realização de testes.

Para este trabalho, o processo de treinamento com o spaCy foi implementado por meio da função `treinamento`, codificada em Python conforme código apresentado na Figura 5. Essa função recebe como entrada três parâmetros: uma lista de dados (`dados`) rotulados de treinamento organizados conforme a estrutura apresentada na Figura 4, uma quantidade de épocas de treinamento (`epocas`) e a taxa de *dropout* que é um valor definido entre 0 e 1, útil para evitar *overfitting* (`dropout`).

Na linha 6, o algoritmo inicia criando um objeto usando o spaCy com um modelo em branco para o idioma Português e segue adicionando um componente de REN ao *pipeline* na linha 8. Nas linhas 9-12, quatro etiquetas são especificadas para as entidades nomeadas. As etiquetas adicionadas se referem às entidades NOME, TITULO, CARGA e PERIODO, que foram definidas na Seção 3.1. Em seguida, nas linhas 14-15 o modelo é inicializado obtendo um otimizador associado e obtém a referência para o componente de NER que foi adicionado ao *pipeline* anteriormente.

Após esta etapa de configuração, nas linhas 17-23, é realizado um laço que implementa o procedimento de treinamento iterando em um determinado número de épocas. Em cada época, os dados de treinamento são embaralhados através de uma função da biblioteca random (linha 18) e outro laço (linhas 20-23) é iniciado iterando sobre cada par texto-anotações nos dados de treinamento. Este laço converte o texto em um objeto “doc”, cria um exemplo a partir do “doc” e das anotações e atualiza o modelo com o exemplo criado, usando a taxa de *dropout* especificada e registrando as perdas durante o treinamento. Por fim, o modelo é salvo no disco com um nome definido (linha 25).

```python
import spacy
from spacy.training.example import Example
import random

def treinamento(dados: list, epocas: int, dropout: float):
    nlp = spacy.blank("pt")

    ner = nlp.add_pipe("ner")
    ner.add_label("NOME")
    ner.add_label("TITULO")
    ner.add_label("CARGA")
    ner.add_label("PERIODO")

    otimizador = nlp.initialize()
    ner = nlp.get_pipe("ner")

    for epoca in range(epocas):
        random.shuffle(dados)
        perdas = {}
        for texto, anotacoes in dados:
            doc = nlp.make_doc(texto)
            exemplo = Example.from_dict(doc, anotacoes)
            nlp.update([exemplo], drop=dropout, losses=perdas)

    nlp.to_disk("nome_modelo_treinado")
```

Figura 5. Código da função utilizada para treinamento do modelo de REN do spaCy

3.4. Procedimento de Avaliação do Modelo

A validação de um modelo é uma etapa importante para avaliar a capacidade de precisão do modelo treinado. Para a validação do modelo foi utilizada a técnica de validação cruzada *K-fold*. Segundo CUNHA (2019) o método *K-fold* consiste em dividir a amostra dos dados em K partes iguais e utilizar uma parte K dos dados para teste e as demais partes para treinamento do modelo. Antes da divisão dos dados em K partes, é realizado um embaralhamento dos dados de forma aleatória. Por fim, para avaliação do modelo podemos realizar uma média aritmética das K métricas de cada divisão, mensurando a capacidade do modelo. Na Figura 6, há um exemplo da validação cruzada *K-fold* com o K = 5, ou seja, a divisão dos dados em 5 partes iguais.

Figura 6. Exemplo da técnica de validação cruzada *K-fold*, com uma divisão em 5 partes

Para avaliação do modelo, utilizamos um método do spaCy para avaliar os componentes de um pipeline, o método *language.evaluate()*, passando por parâmetro um lote de *Example* da parte dos dados de teste. O método retorna um dicionário Python com as pontuações de avaliação do modelo. As métricas de avaliação retornadas pelo método do spaCy são de *precisão*, *recall* e *F-score*, sendo para cada entidade de interesse e uma total sendo a média aritmética dos resultados de cada entidade. No caso deste trabalho, o método retorna as métricas para cada uma das quatro entidades de interesse, bem como uma média das métricas considerando todas as quatro entidades.

A métrica de *precisão*, mede a proporção de ocorrências positivas identificadas corretamente pelo modelo em relação ao número total de ocorrências que o modelo previu como positivas, incluindo verdadeiros positivos e falsos positivos, em outras palavras, a métrica informa a quantidade de acerto do modelo em relação aos resultados dos dados de teste informados. A métrica *recall*, também conhecida como sensibilidade, mede a proporção de ocorrências positivas que foram previstas corretamente pelo modelo em relação ao total de ocorrências positivas dos dados, incluindo os verdadeiros positivos e falsos negativos. Por fim, a métrica *F-score,* que combina o resultado das métricas de *precisão* e *recall* em um resultado único, calculando a média harmônica entre ambas as métricas, como resultado temos uma medida balanceada do desempenho do modelo. Neste trabalho, todas as métricas são dispostas em valores entre 0% e 100%.

4. Experimentos Computacionais e Resultados

Nesta seção são apresentados os experimentos computacionais realizados para parametrizar e validar a ferramenta de REN proposta neste trabalho. Mais especificamente, os experimentos têm o objetivo de responder às seguintes questões de pesquisa:

i) Quais valores de parâmetros são adequados para o processo de treinamento?
ii) Quanto tempo computacional é investido no processo de treinamento?
iii) Qual o desempenho do modelo treinado?

Os resultados dos experimentos foram obtidos em um computador com processador Intel® Core™ i7-8565U 1.8GHz e 8GB de memória RAM, executando o *Windows 11 Home Insider Preview Single Language* como sistema operacional. A implementação da ferramenta foi realizada em Python 3.11.3, utilizando a biblioteca spaCy 3.6.1, no IDE PyCharm 2023.1 (Community Edition) e foi utilizada a versão 3.03 do utilitário Pdftotext.

Para validar a ferramenta proposta neste trabalho são utilizados certificados de atividades complementares de curso obtidos da base de dados do Sistema Integrado de Validação de Atividades Complementares (SIVAC), um sistema web desenvolvido em Django por SOUTHIER e DORNELES (2022), que é utilizado pelos cursos de Bacharelado em Ciência da Computação e Técnico em Informática Integrado do IFFar/FW. A base total de certificados do SIVAC possui em torno de 530 certificados dos mais variados tipos, de cursos, de eventos, palestras, de estágio, entre outros, sendo grande parte dos arquivos em formato PDF. Após aplicar as restrições consideradas na Seção 3.1, foram utilizados 430 certificados que atendiam as delimitações deste estudo e, a este conjunto chamaremos de *Corpus* de Dados do SIVAC (CDS).

Para validação do modelo, foi utilizada a técnica de validação cruzada *K-fold*, conforme descrito na Seção 3.4, dividindo o conjunto de dados CDS em 5 partes iguais com 86 certificados cada.

O primeiro conjunto de experimentos realizados teve como objetivo identificar os parâmetros adequados para o treinamento dos modelos de REN conforme método apresentado na Seção 3.3, mais especificamente, o número de épocas e a taxa de *dropout.* Enquanto uma quantidade de épocas maior geralmente oferece um melhor resultado, esse parâmetro também interfere diretamente no tempo computacional para produção de um modelo.

Desta forma, decidimos utilizar um número de épocas relativamente pequeno para descobrir um valor adequado para a taxa de *dropout*. Assim, fixamos inicialmente o número de épocas em 200 e avaliamos 9 diferentes taxas de *dropout.* No gráfico da Figura 7 são apresentados os resultados médios da métrica *F-score* utilizando a validação cruzada para cada uma das taxas de *dropout* avaliadas. Analisando os resultados é possível perceber que as taxas de *dropout* entre 0.1 e 0.6 possuem desempenho comparáveis, enquanto valores superiores a 0.6 degradam significativamente o desempenho do modelo. Como conclusão deste conjunto de experimentos, entendemos que uma taxa de *dropout* de 0.3 é apropriada.

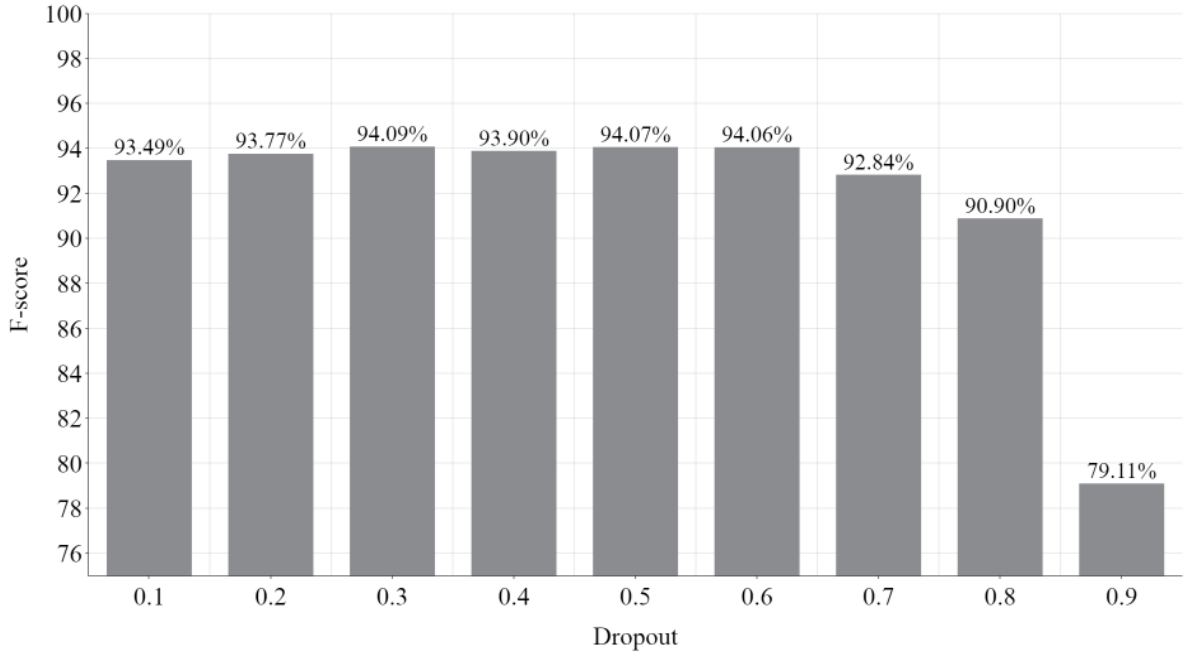

Figura 7. Resultados de *F-score* obtidos para diferentes taxas de *dropout* com 200 épocas.

O segundo conjunto de experimentos teve como objetivo avaliar o desempenho geral do modelo por meio de diferentes métricas, utilizando uma quantidade maior de épocas, bem como a taxa de *dropout* estabelecida no experimento anterior. Na Tabela 1, são apresentados resultados detalhados de cada parte da validação cruzada dos experimentos de validação do modelo REN, utilizando 1000 épocas e uma taxa de *dropout* de 0.3.

Para cada parte são reportados o tempo de treinamento em minutos e valores percentuais para as métricas de *precisão*, *recall* e *F-score*. As colunas NOME, TITULO, CARGA e PERIODO apresentam os resultados das métricas das entidades nomeadas de interesse. A coluna "Média" apresenta a média aritmética de cada métrica das entidades nomeadas de interesse. Finalmente, a última linha da tabela apresenta os resultados médios de cada coluna.

Tabela 1. Resultados gerais de avaliação com 1000 épocas e taxa de *dropout* de 0.3

Parte	Tempo	Métrica	NOME	TITULO	CARGA	PERIODO	Média
1	324	*Precisão*	100.00%	87.34%	97.62%	91.95%	94.23%
		Recall	96.51%	80.23%	95.35%	93.02%	91.28%
		F-score	98.23%	83.64%	96.47%	92.49%	92.71%
2	310	*Precisão*	96.47%	98.70%	97.62%	97.56%	97.59%
		Recall	95.35%	88.37%	95.35%	93.02%	93.02%
		F-score	95.91%	93.25%	96.47%	95.24%	95.22%
3	337	*Precisão*	98.81%	87.65%	97.67%	98.80%	95.73%
		Recall	96.51%	82.56%	97.67%	95.35%	93.02%
		F-score	97.65%	85.03%	97.67%	97.04%	94.35%
4	307	*Precisão*	93.02%	89.29%	98.84%	98.80%	94.99%
		Recall	93.02%	87.21%	98.84%	95.35%	93.61%
		F-score	93.02%	88.24%	98.84%	97.04%	94.29%
5	307	*Precisão*	97.67%	84.52%	94.25%	95.29%	92.93%
		Recall	97.67%	82.56%	95.35%	94.19%	92.44%
		F-score	97.67%	83.53%	94.80%	94.74%	92.69%
Média	317	*Precisão*	97.19%	89.50%	97.20%	96.48%	95.09%
		Recall	95.81%	84.19%	96.51%	94.19%	92.67%
		F-score	96.50%	86.74%	96.85%	95.31%	93.85%

Analisando os resultados médios apresentados na linha final da Tabela 1, observamos que em relação às entidades NOME, CARGA e PERIODO o modelo obteve excelentes resultados de reconhecimento com as métricas de *precisão, recall* e *F-score*. Em menor medida, os resultados da entidade TITULO, embora bons, são significativamente inferiores aos obtidos pelas outras 3 entidades.

Para a entidade NOME, os resultados referentes a *precisão* são de 97.19%, *recall* de 95.81% e *F-score* de 96.50%, que sugere uma boa capacidade do modelo em identificar nomes, mas com uma pequena margem de erro. No caso da entidade TITULO, é apresentado os resultados mais baixos entre as 4 entidades, com uma *precisão* indicando um acerto de aproximadamente 89.50% das predições positivas, mas com um *recall* ligeiramente inferior de 84.19%, o que resultou em um *F-score* de 86.74%.

Já para a entidade CARGA, o modelo também obteve um desempenho sólido parecido com o da entidade NOME, apresentando uma *precisão* de 97.20%, *recall* de 96.51% e *F-score* de 96.85%. E por fim, a última entidade, a PERIODO apresentou resultados ligeiramente inferiores às entidades NOME e CARGA, mas mesmo assim ótimos resultados contando com a *precisão* de 96.48%, *recall* de 94.19% e um *F-score*, a média harmônica entre as duas primeiras, de 95.31%, mostrando resultados bem consistentes, com uma superioridade considerável em relação a TITULO, que detém os menores valores.

Analisando o resultado geral do modelo para todas as entidades apresentado na célula da última coluna e última linha, o modelo alcançou a *precisão* de 95.09%, *recall* de 92.67% e um *F-score* de 93.85%, o que configura resultados promissores na identificação das entidades de interesse avaliadas. Finalmente, em relação ao tempo computacional, podemos observar que, no total, os 5 treinamentos levaram aproximadamente 1585 minutos de tempo computacional, com uma média de 317 minutos em cada parte.

5. Considerações Finais

Neste trabalho foi apresentada uma proposta de ferramenta para a realizar o reconhecimento de entidades nomeadas de certificados de atividades complementares de curso, considerando quatro entidades: nome, título, carga horária e período. Foi apresentada uma implementação de um modelo REN treinado com a biblioteca spaCy, cujo desempenho foi avaliado por meio da técnica de validação cruzada *K-fold* utilizando um *corpus* de dados composto de 430 certificados provenientes dos cursos do eixo de informação e comunicação do IFFar/FW. Os resultados computacionais indicam que o modelo de REN implementado apresentou um bom desempenho geral na tarefa de identificação das entidades de interesse. Em especial, o modelo se destacou na extração das entidades de nomes, carga horária e período atingindo valores superiores à 94% para as métricas avaliadas. Em relação a extração de títulos dos certificados, mesmo sendo uma informação menos estruturada, o modelo apresentou um bom desempenho, sendo superior a 84% para a menor métrica, de *recall*.

Com base nesses resultados, conclui-se que o modelo proposto é promissor para auxiliar na melhoria de sistemas informatizados específicos para validação e gerenciamento de atividades complementares de curso como o SIVAC, ou ainda ser integrado em sistemas acadêmicos pré-existentes em instituições de ensino públicas e privadas.

Este trabalho apresenta ainda as seguintes sugestões de trabalhos futuros: (i) comparar o desempenho do modelo proposto com outras ferramentas; (ii) extração de novas entidades nomeadas em outros tipos de certificados; (iii) avaliação do modelo proposto com dados de testes coletados de outras instituições públicas.

Referências

ALLES, V. J. Construção de um Corpus para Extrair Entidades Nomeadas do Diário Oficial da União Utilizando Aprendizado Supervisionado. Dissertação de Mestrado em Engenharia Elétrica, Publicação 714/2018, Departamento de Engenharia Elétrica, Faculdade de Tecnologia Universidade de Brasília. Brasília, DF. dez. 2018.

CUNHA, J. P. Z. Um estudo comparativo das técnicas de validação cruzada aplicadas a modelos mistos. Dissertação de Mestrado em Ciências. Universidade de São Paulo - USP. São Paulo, SP, fev. 2019.

FONSECA, E. V; CHIELE, G. C; VIEIRA, R; VANIN, A. A. Reconhecimento de Entidades Nomeadas para o Português Usando o OpenNLP. PUCRS. Porto Alegre, RS, Anais do ENIAC 2015, 2015.

SILVA, A. V. e. Um modelo de classificação para o Reconhecimento de Entidades Nomeadas. Dissertação de Mestrado. Faculdade de Filosofia, Letras e Ciências Humanas. USP. São Paulo, SP, dez. 2020.

SOUTHIER, P. H; DORNELES, A. P. Desenvolvimento de uma Plataforma Web em Django para Gerenciamento de Atividades Complementares de Curso. Instituto Federal de Educação, Ciência e Tecnologia Farroupilha (IFFar), Frederico Westphalen, RS, Brasil. Ano 11, n. 1. Anais do XIII Encontro Anual de Tecnologia da Informação - EATI. nov. 2022.

SPACY. spaCy. Disponível em: <https://spacy.io/>. Acesso em: 15 nov. 2023.

POPPLER. Poppler. Disponível em: <https://poppler.freedesktop.org/>. Acesso em: 07 out. 2023.