UNIVERSIDADE
FEDERAL RURAL
DE PERNAMBUCO

Lucas Fernandes Lins

Comparação de arquiteturas de redes neurais em NER para E-commerce brasileiro

Recife

2023

Lucas Fernandes Lins

Comparação de arquiteturas de redes neurais em NER para E-commerce brasileiro

Monografia apresentada ao Curso de Bacharelado em Ciências da Computação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Ciências da Computação.

Universidade Federal Rural de Pernambuco – UFRPE

Departamento de Computação

Curso de Bacharelado em Ciências da Computação

Orientador: André Câmara

Recife

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal Rural de Pernambuco
Sistema Integrado de Bibliotecas
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

L759c Lins, Lucas
Comparação de arquiteturas de redes neurais em NER para E-commerce brasileiro / Lucas Lins. - 2023.
28 f. : il.

Orientador: Andre Camara.
Inclui referências.

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal Rural de Pernambuco, Bacharelado em Ciência da Computação, Recife, 2023.

1. REN. 2. compreensão da consulta. 3. comércio eletrônico. I. Camara, Andre, orient. II. Título

CDD 004

MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO (UFRPE)
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

http://www.bcc.ufrpe.br

FICHA DE APROVAÇÃO DO TRABALHO DE CONCLUSÃO DE CURSO

Trabalho defendido por Lucas Fernandes Lins às 16 horas e 0 minutos do dia 19 de abril de 2023, no link http://meet.google.com/ntm-rgrk-uqa, como requisito para conclusão do curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco, intitulado "*Comparação de arquiteturas de redes neurais em NER para E-commerce brasileiro*", orientado por André Câmara Alves do Nascimento e aprovado pela seguinte banca examinadora:

André Câmara Alves do Nascimento
DC/UFRPE

Rafael Ferreira Leite de Mello
DC/UFRPE

Agradecimentos

Gostaria de começar agradecendo a minha família, que sempre foi minha base para tudo na vida, servindo de inspiração e me dando uma base para seguir em frente.

Mais especificamente, gostaria de agradecer a minha mãe, Ana Claudia Fernandes, por estar sempre ao meu lado me apoiando e me ajudando a conquistar meus objetivos e indiscutivelmente foi a responsável por qualquer sucesso que eu tenha alcançado.

Ao meu pai Lenilson Lins que batalhou para me dar todas as oportunidades na vida, a força de vontade dele é um exemplo para mim todos os dias.

Gostaria também de agradecer ao meu irmão mais novo Luan Lins que está sempre ao meu lado para tudo e para mim é uma das pessoas mais inteligentes que conheço, que me incentiva a sempre aprender mais e procurar um melhor versão de mim mesmo.

Também quero lembrar do meu padrinho Maurício Marques, que foi o primeiro a me apresentar a área de tecnologia e me inspirar a entrar no curso e me dando minhas primeiras oportunidades na área de tecnologia.

A minha prima Florrie, minha tia Avani e minha irmã Stephanie que sempre me apoiaram e me ensinaram muito durante toda minha vida.

Gostaria de destacar meu professor orientador André Câmara que esteve comigo durante todo o processo do TCC e dura re a faculdade, servindo de exemplo essencial para meu desenvolvimento como profissional e pesquisador.

Ao professor Pericles Miranda que foi meu primeiro contato com as linguagens de programação e também com a área de IA, fazendo parte como orientador na minha iniciação científica.

Não posso deixar de fora todos os amigos que fiz durante o curso e que por muitas vezes foram o diferencial e me marcaram muitos durante todos esses anos de curso, em todos os momentos, sejam eles momentos felizes, tristes ou até mesmo de grande pressão, em especial um agradecimento a Rodrigues, Fernando e Giulia.

Agradeço novamente a todos os que estiveram envolvidos direta e indiretamente no meu desenvolvimento pessoal e profissional.

“Talento é 1% inspiração e 99% transpiração.”
(Thomas Edison)

Resumo

Nas plataformas de e-commerce a busca é a etapa mais importante para achar seus produtos, por isso a importância de pesquisas visando sua melhoria, uma das formas de fazer essa melhoria é o uso de *query understanding* (QU). O QU compreende um conjunto de processos que geralmente ocorrem antes das etapas principais de recuperação de documentos. Assim, o QU é fundamental para classificar e reescrever consultas quando se trata de sistemas de grande volume, como é o caso do domínio de e-commerce, especialmente quando seu foco é enriquecer a entrada fornecida. O Reconhecimento de Entidades Nomeadas (NER) é uma das partes mais importantes do entendimento de consultas, dentre outras etapas de processamento de linguagem natural (PLN). O principal objetivo do NER é possibilitar saber quais entidades ou classes estão presentes em uma consulta. Neste artigo, diferentes tipos de técnicas para NER são avaliados em um conjunto de dados de e-commerce em português, com foco em aspectos práticos para uso industrial. Neste trabalho, serão avaliadas diferentes redes neurais, como CNN do framework Spacy, as redes BI-LSTM-CRF, BI-GRU-CRF do framework PyTorch e diferentes métodos de *embbedings*, alcançando resultados satisfatórios como 0,97 de f1-score na base de testes.

Palavras-chave: REN, compreensão da consulta, comercio eletrônico.

Abstract

On e-commerce platforms, the search is the most important step to find your products, hence the importance of seeking to improve them. One of the ways to do this is to use query understanding (QU). The QU comprises a set of processes that usually occur before the main document recovery steps. Thus, QU is critical to sorting and rewriting queries. when it comes to large-volume systems, such as the domain of e-commerce, especially when your focus is on enriching the provided input. Named Entity Recognition (NER) is one of the most important parts of understanding queries, among other processing steps of natural language (NLP). The main objective of NER is to make it possible to know which entities or classes are present in a query. In this article, different types of techniques for NER are evaluated on a dataset of e-commerce in Portuguese, focusing on practical aspects for industrial use. For this experiment, the CNN network from the Spacy framework, the BI-LSTM-CRF, BI-GRU-CRF networks from the PyTorch framework and different embedding methods were used, achieving satisfactory results such as 0.97 f1-score in the test base.

Keywords: NER, Query Understanding, E-commerce.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

NER	Named Entity Recognition
PLN	Processamento de Linguagem Natural
QU	Query Understanding

Sumário

1 Introdução

O crescimento de plataformas de comércio eletrônico tem motiva um número cada vez maior de pesquisas cujo objetivo é refinar a qualidade das buscas nesse cenário em particular (NGUYEN, 2020; GOSWAMI; ZHAI; MOHAPATRA, 2019; ZHANG et al., 2020; BHANGE et al., 2021; PAPENMEIER et al., 2021). Os motores de busca em tais plataformas são sistemas complexos, com um grande número de etapas de processamento e sub-sistemas, normalmente com baixa latência, sempre considerando a melhor experiência possível para o usuário. Tal experiência se reflete tanto na forma em que os produtos devem ser apresentados, bem como a sua ordenação na página de resultados.

Entretanto, com a popularização dos chamados *marketplaces*, i.e., sites em que uma grande variedade de produtos podem ser comercializados, e.g., artigos para limpeza doméstica, roupas, eletrônicos e eletrodomésticos, o número de produtos presentes em um catálogo pode chegar facilmente na casa dos milhões de itens. Tal fato, associado a possibilidade de que um mesmo produto pode ser comercializado por um número arbitrário de vendedores, podendo chegar às centenas de milhares, eleva ainda mais a complexidade de encontrar os produtos corretos para uma busca em específico. Embora na navegação do site normalmente seja possível filtrar os produtos por categoria, a análise de interações demonstra que a grande maioria dos usuários faz pouco ou nenhum uso de tais recursos (VARDASBI; RIJKE; MARKOV, 2020), dependendo quase que inteiramente da busca (i.e., *query*) para realizar suas compras. Devido à dependência dos e-commerces a busca, torna-se interessante o entender o funcionamento desse sistema, que será detalhado a seguir.

A primeira etapa da jornada de busca é o pré-processamento da consulta, onde a consulta do usuário é processada para remover palavras irrelevantes, como artigos, preposições e conjunções. Isso ajuda a reduzir o ruído nos termos de consulta e a melhorar a precisão da busca.

Em seguida, a compreensão da consulta é realizada para identificar o significado subjacente dos termos de busca. Essa etapa envolve técnicas como análise sintática, análise semântica e Reconhecimento de Entidade Nomeada (NER) para melhorar a correspondência dos resultados com a intenção do usuário. O NER é utilizado para identificar entidades nomeadas, como nomes de produtos, marcas, locais e personalidades, o que pode ajudar a melhorar a precisão da busca e fornecer resultados mais relevantes. Como citado anteriormente, muitos sites não utilizam bem seus sistemas de filtros, entretanto, com o auxílio do NER o uso desses filtros pode ser facilitado

ou até mesmo automatizado. Além de ajudar em filtros, o NER pode participar da melhoria de sinônimos e variantes, pois ao reconhecer uma marca é possível sugerir a reescrita, por exemplo, em uma consulta como ”comprar um celular da Apple”, o NER pode identificar ”Apple”como uma entidade nomeada do tipo ”marca”e reconhecer sinônimos como ”iPhone”ou ”iPad”. Entretanto, o uso do NER na compreensão de consulta pode apresentar desafios, como a ambiguidade dos termos de busca. Por exemplo, o termo ”Amazon”pode se referir tanto à empresa de comércio eletrônico quanto ao rio. Nesses casos, é importante que o sistema de busca utilize técnicas adicionais, como análise sintática e semântica, para compreender o contexto da consulta e identificar corretamente a entidade nomeada. Se a consulta do usuário não produzir resultados relevantes, a reescrita da consulta pode ser realizada para gerar novas consultas mais adequadas. A reescrita da consulta pode envolver técnicas como expansão de consulta, redução de consulta ou tradução de consulta.

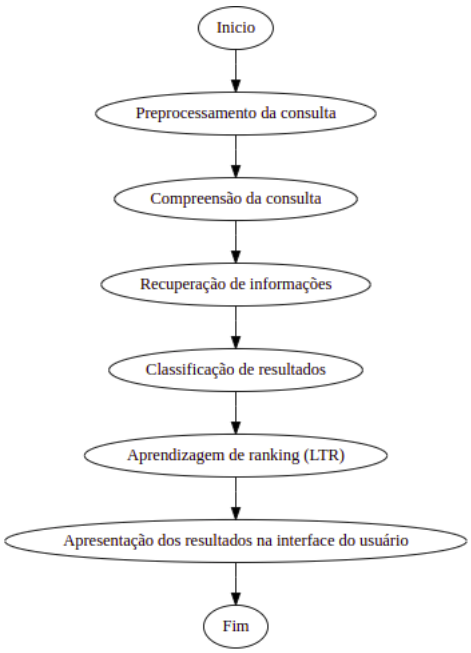
Em seguida, a etapa de recuperação de informações é crucial para a precisão dos resultados de busca. A recuperação de informações envolve a comparação da consulta com as informações indexadas no sistema de busca, para encontrar os documentos que correspondem à consulta. Essa etapa envolve técnicas como modelos de espaço vetorial, modelos probabilísticos e modelos baseados em linguagem para recuperar informações relevantes.

A classificação de resultados é realizada com base em sua pontuação de relevância. A pontuação de relevância é calculada com base em vários fatores, como frequência do termo de busca, proximidade dos termos de busca e popularidade dos produtos. A etapa de classificação dos resultados pode ser realizada com algoritmos como BM25, TF-IDF e Okapi.

Além disso, a aprendizagem de ranking (LTR) é utilizada para melhorar a precisão da classificação dos resultados de busca. O LTR envolve a utilização de um modelo de aprendizado de máquina treinado em um conjunto de dados de treinamento para classificar os resultados de busca com base em várias características, como relevância, popularidade, comentários do usuário, entre outros. Por fim, os resultados da pesquisa são apresentados aos usuários em uma interface de usuário. A interface pode incluir uma lista de resultados, com títulos, descrições e links para os produtos, além de filtros para refinar a pesquisa. Finalizando o processo de busca.

Sendo assim, a correta interpretação das (poucas) palavras fornecidas pelo usuário na caixa de pesquisa é fundamental para identificar os produtos mais relevantes a serem retornados. *Query Understanding* (QU) (CHANG; DEMG, 2020) é uma área de pesquisa na qual o principal objetivo é detectar a intenção do usuário. Uma vez identificada a intenção do usuário, a consulta não será mais tratada apenas como um conjunto de palavras a serem buscadas em um índice invertido, pois os significados

Figura 1 – Processo de busca.



extraídos das palavras-chave da consulta serão usados para fornecer o resultado de pesquisa sensato e relevante. QU envolve diversas tarefas, como por exemplo, classificação, transformação e reescrita de queries, bem como o reconhecimento de entidades nomeadas, o foco do presente trabalho.

A grande maioria das pesquisas em reconhecimento de entidades nomeadas (NER) concentra-se em um pequeno conjunto de tipos de entidades proeminentes, como pessoas, organizações, ou ainda em domínios específicos, como saúde e biologia (doenças, genes, proteínas, compostos químicos)(KRALLINGER et al., 2015; CHOI et al., 2016), direito (Luz de Araujo et al., 2018), para os quais conjuntos de dados rotulados estão disponíveis. Recentemente, alguns trabalhos tem se dedicado a analisar o contexto específico de comércio eletrônico, com suas especificidades: sentenças em geral muito curtas (2 a 5 palavras) e ambiguidade contextual (e.g., “32” seria o tamanho em polegadas? Tamanho da calça? Do sapato? Memória?) (WEN et al., 2019; PAPENMEIER et al., 2021). No entanto, poucos trabalhos são dedicados à queries escritas em português brasileiro (SILVA et al., 2021; BARBIRATO; REAL; CASELI, 2021).

Além disso, trabalhos mais atuais geralmente usam as mais recentes arquiteturas neurais (DEVLIN et al., 2019) e modelos de linguagem (ZHANG et al., 2020) cujo foco é a melhoria marginal das métricas de avaliação, enquanto a viabilidade em aplicações do mundo real nem sempre é considerada. Até onde temos conhecimento, não existem trabalhos que avaliem arquiteturas de NER disponíveis em ferramentas de PLN industriais em pesquisas e comércio eletrônico. Dessa forma, este trabalho avalia o desempenho de uma arquitetura de rede neural com foco em eficiência e tempo de

inferência, considerando diferentes modelos de representação de linguagem.

2 Trabalhos relacionados

Nos últimos anos, têm-se observado um crescente avanço nas pesquisas aplicando técnicas de redes neurais profundas e modelos de linguagem em problemas de NER nos mais diversos contextos. Muitos desses trabalhos baseiam-se na arquitetura de redes neurais recorrentes (*recurrent neural network* - RNN), especialmente redes BiLSTM (*Bidirectional Long Short Term Memory*) associada ao método probabilístico *Conditional Random Fields* (CRF), uma vez que tal combinação têm obtido excelentes resultados em outros domínios (MOTA et al., 2021; BHANGE et al., 2021).

Outros têm explorado arquiteturas alternativas, visando um melhor desempenho em tempo de inferência. (BHANGE et al., 2021) propôs o uso de uma rede neural BiGRU-CRF (*Bidirectional Gated Recurrent Unit* - BiGRU). A fim de reduzir o esforço de anotação manual, os autores propõem a adoção de uma base de dados a partir da combinação de três sub-conjuntos de queries anotadas, uma com grande quantidade de entidades anotadas, e outra cujo foco seria na qualidade das entidades anotadas. Outro trabalho que buscou superar as dificuldades de obtenção de dados anotados de boa qualidade, (WEN et al., 2019) propôs uma solução para obter anotações humanas sem a necessidade esforço humano extra. Sua solução consistiu em juntar anotações de diferentes sistemas legados para criação de um sistema independente para NER.

Poucos são os trabalhos que exploram aplicações de técnicas e modelos de linguagem específicas para o idioma português brasileiro no contexto de comércio eletrônico. Recentemente, (SILVA et al., 2021) apresenta uma comparação e vários experimentos envolvendo a plataforma *MIT Information Extraction tool* (MITIE) e o modelo de linguagem contextual *Bidirectional Encoder Representations from Transformers* (BERT), na tarefa de extração de entidades nomeadas em títulos de produtos em português (na categoria *smartphones*/celulares). Foram consideradas 10 entidades distintas: a definição do produto, marca, modelo, memória interna, tamanho da tela, processador, cor, dentre outras. Os resultados indicaram que ambas as estratégias alcançaram resultados satisfatórios, com uma ligeira vantagem para a abordagem clássica (MITIE) devido a sua maior simplicidade e menor custo computacional.

Outro trabalho recente que aborda um problema correlato ao NER também no domínio de comércio eletrônico em português é proposto em (BARBIRATO; REAL; CASELI, 2021). A tarefa em questão é a extração de relações (*Relation Extraction* - RE), que objetiva identificar relações entre termos ou entidades no texto. Os autores avaliaram dois modelos BERT para o português, ajustados para o contexto de títulos e descrições de produtos da categoria *smartphone* e celular. Foram considerados 8 tipos

de relações entre as entidades descritas no modelo de NER desenvolvido em (SILVA et al., 2021). Ambos os trabalhos (SILVA et al., 2021; BARBIRATO; REAL; CASELI, 2021) tem como principal objetivo o enriquecimento do catálogo de produtos, uma vez que foi possível notar a existência de informações relevantes presentes nos campos não estruturados que não estavam evidentes nos cadastros dos campos estruturados acerca do produto.

(PAPENMEIER et al., 2021) pontua o fato de que a escassez de bases anotadas e abertas ao público está diretamente ligada à questões de privacidade e competitividade no mercado. Os autores apresentam uma base de dados de 3,540 queries em inglês, abertas ao público, para aumentar as possibilidades de pesquisa na área. Dessa forma, (PAPENMEIER et al., 2021) apresenta um conjunto de dados anotado por 1.818 pessoas, contendo queries relativas a dois produtos (*laptops* e jaquetas), que dentre os diversos usos, incluem possíveis aplicações de NER. Entretanto, até o presente momento, não temos conhecimento de nenhum trabalho que tenha avaliado ferramentas e modelos de NER voltados a aplicações industriais que possam ser utilizados em tempo de execução nas etapas de pre-processamento de QU em português.

3 Materiais

Para o contexto desta pesquisa, foram consideradas um total de 2.000 queries distintas, categorizadas (por um modelo de classificação de queries) na categoria de moda, extraídas a partir de um grande portal de comércio eletrônico brasileiro. As queries selecionadas foram separadas em 2 subconjuntos de forma aleatória, os quais foram submetidos a procedimentos diferentes para anotação: o primeiro anotado manualmente, o segundo anotado a partir de um processo semi-automático baseado em *matches* textuais de dados de catálogo (marcas, produtos, etc).

As queries utilizadas no modelo podem ser separadas em três conjuntos distintos conforme proposto por (BHANGE et al., 2021) de forma que cada base tenha um objetivo. O primeiro anotado manualmente, o segundo gerado sinteticamente a partir de um conjunto de exemplos de cada entidade, e por último uma marcação através de *match* textual. Cada um desses datasets podem acrescentar um grande valor ao modelo pelas suas características particulares (BHANGE et al., 2021).

O dataset manual, por ser anotado por humanos, possui o nível de interpretação das queries mais confiável e próximo da realidade. Apesar disso, é o mais difícil, lento e exaustivo de obter. O sintético é prático, pode criar diversos exemplos interessantes para o modelo, com diversas maneiras diferentes de escrever uma query, além de uma capacidade de gerar uma imensa quantidade de dados em um curto espaço de tempo. O terceiro dataset, de queries anotadas por *match* textual, são queries feitas por humanos e anotadas através do casamento de padrões entre os termos e possíveis valores já catalogados para entidades. Os três tipos de anotação em conjunto podem ser usados para balancear a base de dados, criando um bom volume sem perder informação. Para os experimentos feitos neste trabalho foram considerados 939 anotações manuais, 561 anotações sintéticas e 500 anotações por *match* em dados de catalogo.

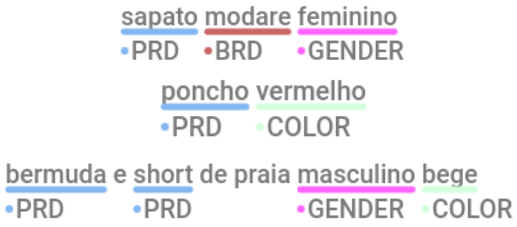
Tabela 1 – Estatísticas dos conjuntos de dados analisados

	Núm.	**Núm.**	**Entidade**			
Dataset	**Queries**	**Tokens**	**PRD**	**BRD**	**GENDER**	**COLOR**
Manual	939	3262	1009	91	454	650
Sintético	561	2038	561	561	0	561
Baseado em catálogo	500	1492	390	102	41	231

3.0.1 Manual

Para as anotações manuais foi usado o sistema de anotação *Doccano* (NAKAYAMA et al., 2018), em que uma amostra de 1000 queries foi aleatoriamente selecionada e marcada com auxílio do software (Figura 2). De 1000 queries, 61 foram excluídas por fatores como erros de escrita ou ausência de entidades na query, reforçando a ideia da qualidade da anotação manual, pois esse nível de interpretação ocorre apenas em anotação humana.

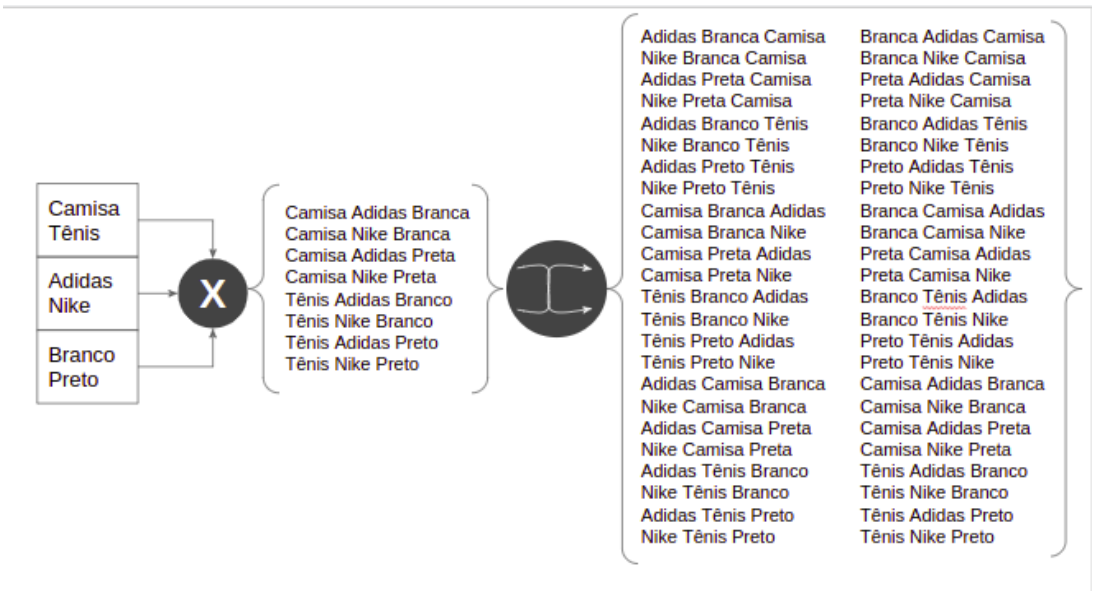
Figura 2 – Exemplos de queries anotadas manualmente.



3.0.2 Sintético

Para a obtenção das queries sintéticas, foram utilizadas listas pré-catalogadas de produtos, cores e marcas. Foi realizado um produto cartesiano entre as três listas, gerando queries com combinações dos valores disponíveis nas entidades {PRD, BRD, COLOR, GENDER}. Além disso, para cada query obtida foi executada uma permutação dos termos para aleatorizar as posições em que as entidades apareciam e impedir que todos os exemplos fossem gerados na mesma ordem. Ilustrado na figura 3

Figura 3 – *Pipeline* para geração de queries sintéticas.



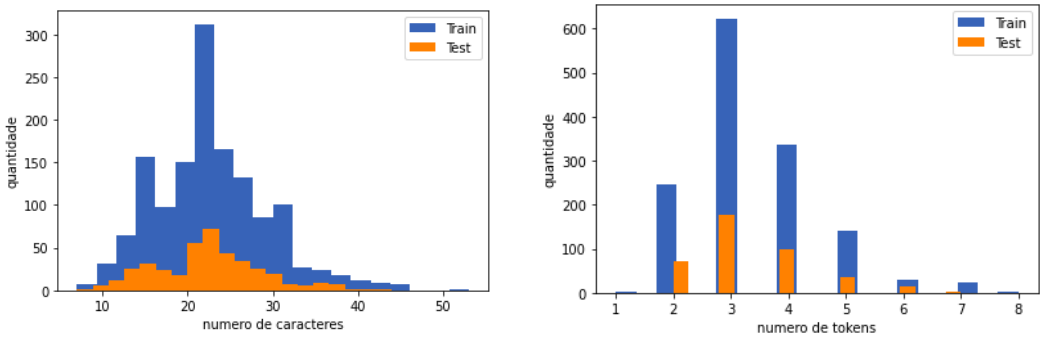
3.0.3 Baseado em catálogo

As anotações baseadas no catálogo são as mais simples, foram utilizadas listas pré-catalogadas de produtos, cores, gêneros e marcas para realizar a marcação através do *match* textual em queries presentes no conjunto considerado. O contexto é um fator importante em queries de comércio eletrônico, porém o tamanho reduzido faz com que o contexto seja escasso e difícil de compreender. Isso causa problemas de interpretação nas queries anotadas via match textual. Um bom exemplo são as queries “camisa vinho” e “vinho tinto”, em que simplesmente anotando a palavra “vinho” como o produto, ocasionará um erro quando a palavra for utilizada como uma cor. Entretanto, como mostra (BHANGE et al., 2021) a anotação baseada em catálogo é válida e importante para aumentar a quantidade de exemplos na base de dados.

3.0.4 Distribuição

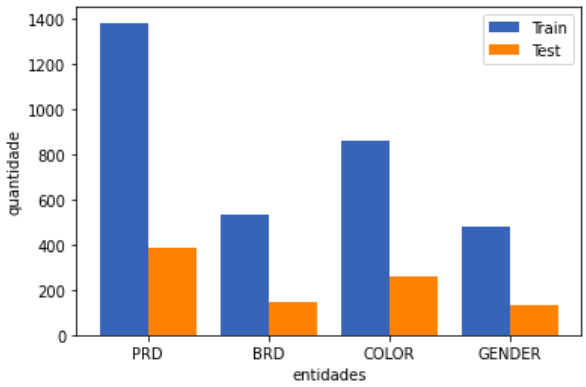
As duas mil queries foram divididas em 1402 para treino, 200 para validação e 398 para testes, esses dados foram os mesmos em todos os testes das redes neurais. Uma característica importante do dataset é a quantidade de caracteres e tokens nas queries, uma vez que textos de busca em comércio eletrônico tendem a ser curtos, como pode ser observado na Figura 4.

Figura 4 – Número de caracteres e tokens respectivamente, por query no conjunto de treino e teste



Outra característica importante é a distribuição de entidades na base de dados, ilustrada na Figura 5 para os conjunto de treino e teste. O produto (PRD) é a entidade dominante, pois a maioria das queries busca por algum produto específico, seguido de cor (COLOR) que foi expandido sinteticamente. Gênero (GENDER) e marca (BRD) apresentam uma quantidade menor de queries pois não receberam tantas queries sintéticas.

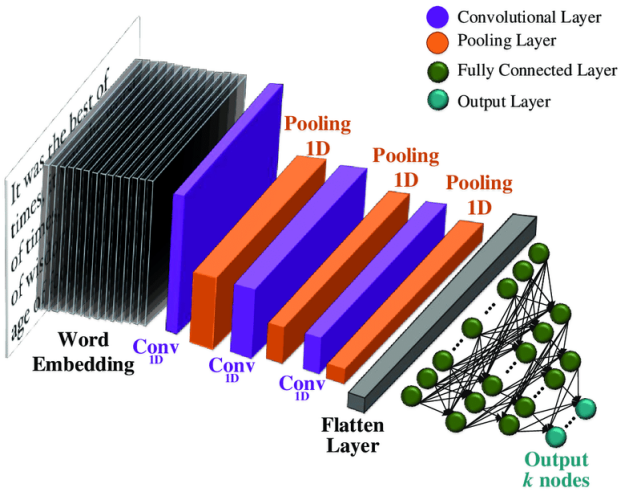
Figura 5 – Distribuição de entidades por query no conjunto de treino e teste



4 Método

Os experimentos neste trabalho concentraram-se na avaliação do impacto do uso de diferentes modelos de linguagem no contexto de NER em queries do domínio de comércio eletrônico. O *framework* Spacy (HONNIBAL et al., 2020) foi adotado como plataforma de comparação dado o seu amplo alcance na comunidade de PLN, disponibilidade de modelos de linguagem em português, além de ter como principal característica ser uma plataforma para uso industrial de aplicações de PLN, gerando modelos compactos e com baixa latência em tempo de inferência (HONNIBAL et al., 2020). O modelo de NER implementado no Spacy baseia-se em uma arquitetura de rede neural convolucional (CNN), com estrutura similar a apresentada na Figura 6. Como etapa de preprocessamento foi adotado apenas o tokenizador do Spacy, cuja saída é apresentada diretamente à camada de entrada da CNN

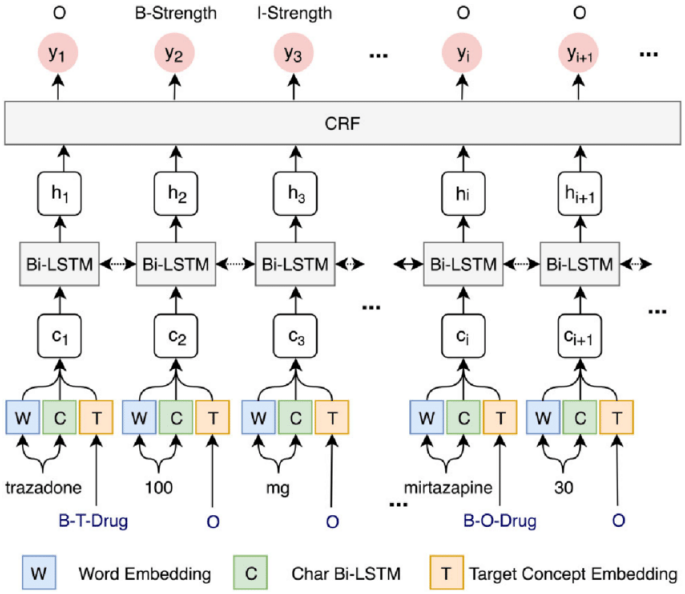
Figura 6 – Rede convolucional em um texto



Fonte: (KOWSARI et al., 2019)

Em contrapartida, foi escolhido o modelo Bi-LSTM-CRF (HUANG; XU; YU, 2015) para ser comparado com a CNN do Spacy. Contando com um modelo duplo recorrente (Bi-LSTM) para extrair as nuances do texto para seu vetor interno de arquitetura recorrente, seguido de uma camada de CRF (Conditional Random Field) que usa de artifícios estatísticos para tentar corrigir erros passados pela camada anterior. A arquitetura utilizada foi baseada no código provido pelo pytorch (PASZKE et al., 2017), que foi o framework escolhido para implementação.

Figura 7 – Modelo Bi-LSTM-CRF



Fonte: (WEI et al., 2019)

4.0.1 Modelos e arquiteturas de *Embeddings*

Para este trabalho, foram avaliadas três arquiteturas de *embeddings*: Token2Vec (Spacy), BERT (DEVLIN et al., 2019) e a camada de embeddings disponibilizada pelo pytorch. Na arquitetura Token2Vec, foram avaliadas representações treinadas no próprio conjunto de dados, bem como um modelo pré-treinado em português *pt_core_news_lg* do próprio Spacy. No caso do BERT, foi adotado o modelo pré-treinado *neuralmind-portuguese-cased* (SOUZA; NOGUEIRA; LOTUFO, 2020). Já a camada do pytorch funciona de forma mais simples, ela mapeia um índice inteiro (por exemplo, um ID de palavra) para um vetor de embedding denso. Durante o treinamento, a camada de embeddings aprende a atualizar os vetores de embedding para minimizar a perda da tarefa específica, enquanto no teste, ela mapeia cada índice de entrada para seu vetor de embedding correspondente, que é então passado para as camadas seguintes da rede neural. O Token2Vec e a camada do pytorch são modelos mais simples, com tempo de inferência mais rápido que o BERT, que por sua vez, é muito mais robusto e mais sensível a variações de contexto, uma vez que cada token pode ter uma representação distinta a depender da sua vizinhança. Porém, tal capacidade vem associada a um maior custo, com maior tempo de inferência quando comparado as duas alternativas .

4.0.2 Análise

As bases de dados separadas em anotações manuais, por catálogo e sintético são usadas separadamente para fazer os testes e calcular as métricas de F-score, precisão e *recall*, de forma que seja possível comparar o desempenho do modelo nos diferentes tipos de anotação. O objetivo da comparação em diferentes bases de dados

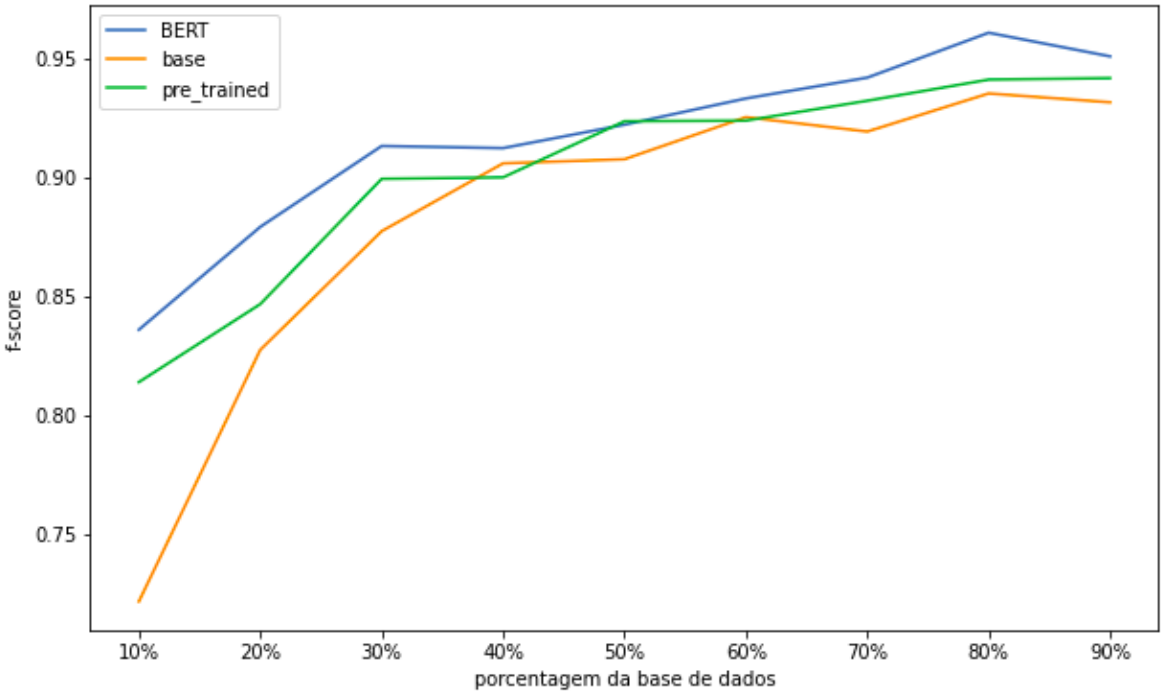


Figura 8 – Resultados obtidos com diferentes percentuais de uso do conjunto de treinamento

é mensurar a capacidade do modelo em contextos distintos, permitindo diferenciar um modelo capaz de generalizar diferentes contextos e formatos da query de um modelo que "decora" uma base específica, conhecido como "*overfitting*".

4.0.3 Amostras

Objetivando entender diferentes formas de melhorar a anotação do modelo, foi feito um estudo treinando a rede com diferentes tamanhos da base de dados, tornando possível visualizar a melhora do classificador de forma crescente com o aumento do tamanho da amostra. A média dos resultados obtidos nos três conjuntos de teste (manual, sintético e catálogo) com as três representações avaliadas, considerando um percentual da base de treinamento são apresentados na Figura 8. A partir da análise do gráfico é possível entender se uma expansão nas anotações é necessária para melhoria do modelo. Os resultados obtidos indicaram que o tamanho do conjunto de treinamento considerado é suficiente para obter bons resultados no teste, i.e., a curva de F-score tende a convergir no resultado final após 80% dos exemplos de treinamento serem considerados.

5 Experimentos e Resultados

Os experimentos foram realizados em um hardware com as seguintes configurações: processador Intel(R) Xeon(R) CPU @ 2.30GHz e GPU Titan. Os modelos Tok2Vec e PyTorch foram treinados apenas com o uso de CPU e o BERT com recursos de GPU, o que fez com que todos os modelos apresentassem tempos de treinamento próximos, durando em média 30 minutos. Os hiperparâmetros utilizados nos experimentos são apresentados na Tabela 2. Apos o período de treinamento, iniciaram-se os testes de desempenho, visando testar cada um dos *embeddings* em relação ao seu tempo de inferência.

Tabela 2 – Hiper-parâmetros utilizados nos experimentos.

Parâmetros	**Tok2Vec**	**BERT**	**BI-LSTM-CRF**
Max steps	20000	20000	2000
Batch size	1000	128	1000
Patience	1600	1600	0
Hidden width	64	64	128
L2	0.1	0.1	0
Learning rate	0.001	adaptative	0.0001

Para essa medição, cada modelo foi avaliado 20 vezes no conjunto de testes e em cada uma dessas vezes foi medido o tempo de inferência de cada query. Como esperado, os algoritmos Tok2Vec e Bi-LSTM-CRF foram mais rápidos do que o BERT, que por sua vez, alcança resultados superiores em termos de qualidade das predições (Tabela 3).

Tabela 3 – Tempo de inferência

Modelo	**Tempo de inferencia no conjunto de teste**	**Desvio padrão**
Tok2Vec	0.002275s	0.000081
BERT	0.224208s	0.011668
Tok2Vec Pre-trained	0.009278s	0.000283
Bi-LSTM-CRF	0.001507	0.000067

A avaliação dos resultados dos modelos, exposta na Tabela 4, mostra que os modelos apresentaram resultados similares, principalmente no conjunto de anotações manuais, com o BERT apresentando um resultado um pouco maior que as outras duas configurações utilizando o Tok2Vec.

Tabela 4 – Média dos resultados (PRD/BRD/GENDER/COLOR).

Dataset	Embedding	F1-Score	Precisão	Recall
Manual	Tok2Vec	**0,956355**	0,965451	0,947761
Sintética	Tok2Vec	0,981818	1,000000	0,982143
Catálogo	Tok2Vec	0,889211	0,840805	0,973684
Manual	BERT	**0,973282**	0,984375	0,962687
Sintética	BERT	0,995556	0,991150	1,000000
Catálogo	BERT	0,933962	1,000000	0,979167
Manual	Tok2Vec Pre-trained	**0,966656**	0,970787	0,962687
Sintética	Tok2Vec Pre-trained	0,977778	0,990741	0,982143
Catálogo	Tok2Vec Pre-trained	0,921384	0,885058	0,973684
Manual	Bi-LSTM-CRF	**0,911619**	0,915217	0,913658
Sintética	Bi-LSTM-CRF	0,956217	0,973809	0,941032
Catálogo	Bi-LSTM-CRF	0,923394	0,933448	0,917526

Ao analisar a tabela apresentada, podemos notar que os modelos com *embeddings* Tok2Vec apresentaram um tempo de inferência consideravelmente menor em comparação aos modelos com BERT. Por exemplo, o modelo com Tok2Vec pré-treinado é 24 vezes mais rápido do que o modelo com BERT, enquanto o modelo com Tok2Vec treinado apenas com o conjunto de dados específico é aproximadamente 100 vezes mais rápido, e o mais discrepante é o modelo Bi-LSTM-CRF que é aproximadamente 150 vezes mais rápido.

Em relação à precisão, o modelo treinado com o BERT apresentou um desempenho superior em relação aos modelos com Tok2Vec, com uma diferença de até 1,77% no F1-Score. Por outro lado, os modelos com Tok2Vec pré-treinados apresentaram resultados próximos ao modelo com BERT, com uma diferença de apenas 0,6% no F1-Score. Já o modelo Bi-LSTM-CRF apresentou um desempenho intermediário em relação aos modelos com BERT e Tok2Vec, com seu desempenho variando entre 0,911619 e 0,956217 no F1-Score, dependendo do conjunto de dados, representando uma diferença de 6% e 3,9% respectivamente quando comparado com o modelo BERT. Dessa forma, a escolha do modelo mais adequado dependerá das necessidades específicas de cada caso. Se a velocidade de inferência for um fator crítico, os modelos com *embeddings* Tok2Vec e Bi-LSTM-CRF podem ser mais indicados. Já se a precisão for mais importante, os modelos com BERT podem ser preferíveis. Quando o tempo de inferência é importante, o Tok2Vec pre-trained e o Bi-LSTM-CRF são fortes opções, pois tem bons resultados equilibrados com alta velocidade, os resultados variam, de forma que a primeira rede pode ser 5% melhor como no exemplo manual ou até mesmo 0,2% pior que a segunda, como no caso de catálogo. A tabela apresentada pode ser útil para a escolha do modelo mais apropriado para cada situação.

6 Conclusão

Este trabalho apresentou uma avaliação de diferentes modelos de linguagem em um contexto de aplicação industrial em um domínio específico, i.e., reconhecimento de entidades nomeadas em queries de comércio eletrônico da categoria de moda. O processo de produção do conjunto de dados para treinamento proposto envolve a combinação de diferentes estratégias de anotação, gerando três conjuntos separados de dados de treinamento.

Os resultados indicaram que o processo proposto foi eficiente, e que a plataforma Spacy e PyTorch foram capazes de produzir resultados satisfatórios no conjunto de testes analisado. Além disso, os *embeddings* provenientes do Tok2Vec e PyTorch mostraram-se candidatos desejáveis para uso dentro de um cenário que busca melhor desempenho e uma boa capacidade assertiva.

Como trabalhos futuros, pretendemos avaliar o modelo desenvolvido em testes A/B, a fim de medir os ganhos em termos de engajamento e conversão de usuários reais, além de testar outros *embeddings* em conjunto com a rede do PyTorch, bem como avaliar o processo de anotação adotado em outras categorias de queries (e.g., *smartphones*, eletrodomésticos, etc).

Referências

BARBIRATO, J. G. M.; REAL, L.; CASELI, H. d. M. Relation extraction in structured and unstructured data: a comparative investigation on smartphone titles in the e-commerce domain. In: . [s.n.], 2021. p. 101–110. Disponível em: ⟨https://github.com/plkmo/BERT-Relation-Extraction⟩. Citado 3 vezes nas páginas 12, 14 e 15.

BHANGE, B. R. et al. Named Entity Recognition for E-Commerce Search Queries. *Data Science for Retail and E-Commerce Workshop*, 2021. Disponível em: ⟨https://sdm-dsre.github.io/pdf/named_entity.pdf⟩. Citado 4 vezes nas páginas 10, 14, 16 e 18.

CHANG, Y.; DEMG, H. *Query Understanding for Search Engines*. [S.l.: s.n.], 2020. ISBN 9783030583330. Citado na página 11.

CHOI, W. et al. A corpus for plant-chemical relationships in the biomedical domain. *BMC Bioinformatics*, v. 17, n. 1, p. 386, 2016. ISSN 1471-2105. Disponível em: ⟨http://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-016-1249-5⟩. Citado na página 12.

DEVLIN, J. et al. BERT: Pre-training of deep bidirectional transformers for language understanding. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 4171–4186. Disponível em: ⟨https://aclanthology.org/N19-1423⟩. Citado 2 vezes nas páginas 12 e 21.

GOSWAMI, A.; ZHAI, C.; MOHAPATRA, P. Learning to diversify for e-commerce search with multi-Armed bandit. *CEUR Workshop Proceedings*, v. 2410, n. 3, 2019. ISSN 16130073. Disponível em: ⟨http://ceur-ws.org/Vol-2410/paper18.pdf⟩. Citado na página 10.

HONNIBAL, M. et al. spacy: Industrial-strength natural language processing in python. Zenodo, Honolulu, HI, USA, 2020. Citado na página 20.

HUANG, Z.; XU, W.; YU, K. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015. Citado na página 20.

KOWSARI, K. et al. Text classification algorithms: A survey. *Information*, MDPI, v. 10, n. 4, p. 150, 2019. Citado na página 20.

KRALLINGER, M. et al. CHEMDNER: The drugs and chemical names extraction challenge. *Journal of Cheminformatics*, Chemistry Central Ltd, v. 7, n. Suppl 1, p. S1, 2015. ISSN 17582946. Disponível em: ⟨http://www.jcheminf.com/content/7/S1/S1⟩. Citado na página 12.

Luz de Araujo, P. H. et al. Lener-Br: A Dataset for Named Entity Recognition in Brazilian Legal Text. In: *International Conference on Computational Processing of*

the Portuguese Language. Cham: Springer, 2018. v. 11122 LNAI, p. 313–323. ISBN 9783319997216. ISSN 16113349. Citado na página 12.

MOTA, C. C. et al. Reconhecimento de entidades nomeadas em documentos jurídicos em português utilizando redes neurais. In: SBC. *Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional*. [S.l.], 2021. p. 130–140. Citado na página 14.

NAKAYAMA, H. et al. *doccano: Text Annotation Tool for Human*. 2018. Software available from https://github.com/doccano/doccano. Disponível em: ⟨https://github.com/doccano/doccano⟩. Citado na página 17.

NGUYEN, D. *Improving Ecommerce Search with Query Named Entity Recognition*. Tese (Doutorado), 2020. Disponível em: ⟨https://www.theseus.fi/bitstream/handle/10024/344778/DangNguyen-thesis.pdf?sequence=2⟩. Citado na página 10.

PAPENMEIER, A. et al. Dataset of Natural Language Queries for E-Commerce. In: *CHIIR 2021 - Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*. [S.l.]: Association for Computing Machinery, Inc, 2021. p. 307–311. ISBN 9781450380553. Citado 3 vezes nas páginas 10, 12 e 15.

PASZKE, A. et al. Automatic differentiation in pytorch. 2017. Citado na página 20.

SILVA, D. F. et al. Named Entity Recognition for Brazilian Portuguese Product Titles. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 13074 LNAI, p. 526–541, 2021. ISSN 16113349. Citado 3 vezes nas páginas 12, 14 e 15.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. BERTimbau: pretrained BERT models for Brazilian Portuguese. In: *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*. [S.l.: s.n.], 2020. Citado na página 21.

VARDASBI, A.; RIJKE, M. de; MARKOV, I. Cascade Model-based Propensity Estimation for Counterfactual Learning to Rank. In: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY, USA: ACM, 2020. p. 2089–2092. ISBN 9781450380164. Disponível em: ⟨https://staff.fnwi.uva.nl/m.derijke/wp-content/papercite-data/pdf/vardasbi-2020-cascade.pdfhttps://dl.acm.org/doi/10.1145/3397271.3401299⟩. Citado na página 10.

WEI, Q. et al. A study of deep learning approaches for medication and adverse drug event extraction from clinical text. *Journal of the American Medical Informatics Association : JAMIA*, v. 27, 05 2019. Citado na página 21.

WEN, M. et al. Building large-scale deep learning system for entity recognition in e-commerce search. In: *BDCAT 2019 - Proceedings of the 6th IEEE/ACM International Conference on Big Data Computing, Applications and Technologies*. [S.l.]: Association for Computing Machinery, Inc, 2019. p. 149–154. ISBN 9781450370165. Citado 2 vezes nas páginas 12 e 14.

ZHANG, H. et al. Bootstrapping Named Entity Recognition in E-Commerce with Positive Unlabeled Learning. p. 1–6, 2020. Disponível em: ⟨https://arxiv.org/pdf/2005.11075.pdf⟩. Citado 2 vezes nas páginas 10 e 12.