

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

DOUGLAS LÁZARO S. P. SILVA

**Hermeneutics: Ferramenta de auxílio para
realização de revisões sistemáticas da
literatura**

Monografia apresentada como requisito parcial
para a obtenção do grau de Bacharel em Ciência
da Computação

Orientador: Prof. Dr. Leandro Krug Wives
Coorientador: Dr. Vinícius Woloszyn

Porto Alegre
2021

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos André Bulhões Mendes

Vice-Reitora: Prof^ª. Patricia Pranke

Vice-Pró-Reitor de Graduação: Prof. Leandro Raizer

Diretor do Instituto de Informática: Profa. Carla Maria dal Sasso Freitas

Coordenador do Curso de Ciência de Computação: Prof. Rodrigo Machado

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

*“Either mathematics is too big for the human mind,
or the human mind is more than a machine“*

— KURT GÖDEL

AGRADECIMENTOS

Primeiramente, agradeço a Deus por ter sustentado e direcionado a longa jornada que foi fazer a graduação. Em segundo lugar, agradeço ao meu irmão, que durante esses anos aguentou e me suportou falando sobre conteúdos das disciplinas, sem nem saber do que se tratava. Também gostaria de agradecer às inúmeras pessoas que passaram alguns momentos dessa trajetória comigo, em particular ao pessoal do projeto de tecnologias educacionais do qual eu fiz parte no início da graduação, aos colegas de curso que se encontravam nos intervalos entre as disciplinas para jogar a famosa *sinuquinha* do DA-COMP, às pessoas que eu conheci através da ABU, se tornaram grandes amigos e me levaram a conhecer o pessoal do FIDES. Por fim, agradeço aos demais familiares e amigos que sempre me apoiaram e incentivaram meus estudos.

SUMÁRIO

LISTA DE FIGURAS	7
LISTA DE TABELAS	8
LISTA DE ABREVIATURAS E SIGLAS	9
RESUMO	11
ABSTRACT	12
1 INTRODUÇÃO	13
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 Revisão Sistemática da Literatura	16
2.1.1 Detalhamento	17
2.2 Técnicas de mineração textual	19
2.3 Natural Language Processing	19
2.3.1 <i>Named-Entity Recognition</i>	20
2.3.2 <i>Semantic Role Labeling</i>	21
2.4 Resumo do Capítulo	21
3 TRABALHOS RELACIONADOS	22
3.1 Ferramentas de Revisão Sistemática da Literatura	22
3.1.1 Técnicas de mineração textual utilizadas.....	24
3.1.2 Sobek.....	24
3.2 Frameworks de reconhecimento NER/SRL	26
3.2.1 SciERC e SciIE	27
3.2.1.1 Categoria Entidade.....	27
3.2.1.2 Categoria Relacionamento	28
3.2.2 SpERT	29
4 ESTUDO DESENVOLVIDO	31
4.1 Solução Proposta	31
4.1.1 Requisitos.....	32
4.1.1.1 Funcionais	32
4.1.1.2 Não-Funcionais	32
4.1.2 User Stories	32
4.1.2.1 Casos de Uso.....	33
4.2 Implementação	34
4.2.1 Fluxo principal da ferramenta.....	34
4.2.2 Estruturas e modelos	35
4.2.2.1 Bibliotecas Digitais	36
4.2.2.2 Interpretador.....	37
4.2.2.3 Banco de Dados	38
4.2.3 Algoritmo de ranqueamento	40
4.2.4 Arquitetura e Tecnologias	41
4.2.4.1 Modelo Arquitetural.....	41
4.2.4.2 Front-end.....	43
4.2.4.3 Back-end	43
5 AVALIAÇÃO DA FERRAMENTA	47
5.1 Conceção	47
5.1.1 Tópicos Pesquisados	47
5.1.2 Perguntas	48
5.1.2.1 Questionário SUS:	48
5.1.2.2 Conjunto de perguntas específicas:	49
5.1.2.3 Grau da concordância	49

5.2 Resultados	49
5.2.1 Perfil dos participantes.....	50
5.2.2 Usabilidade.....	50
5.2.3 Feedback das funcionalidades.....	52
5.2.4 Desempenho nos experimentos.....	54
6 CONCLUSÃO	56
6.1 Trabalhos Futuros	57
REFERÊNCIAS	61
APÊNDICE A — MANUAL DA FERRAMENTA	64

LISTA DE FIGURAS

Figura 3.1 SOBEK - Fluxograma	25
Figura 3.2 SpERT - Framework	30
Figura 4.1 Hermeneutics - Diagrama de casos de uso	34
Figura 4.2 Hermeneutics - Fluxograma Principal	35
Figura 4.3 Research - Diagrama de Classes	36
Figura 4.4 DigitalLibrary - Fluxograma simplificado	37
Figura 4.5 Hermeneutics - Diagrama de Classes	38
Figura 4.6 Banco de Dados - Modelo Entidade Relacionamento	39
Figura 4.7 Arquitetura - Modelo MTV	42
Figura 4.8 Hermeneutics - Tela Inicial	43
Figura 4.9 Fluxograma - Fila de Tarefas (Pipeline)	44
Figura 4.10 Diagrama - Assincronismo	45
Figura 4.11 Hermeneutics - Tela de Resultados	46
Figura 5.1 SUS Score - Resultados	51
Figura 5.2 SUS Questionário - Resultados	52
Figura 5.3 NER - Resultados	52
Figura 5.4 SLR - Resultados	53
Figura 5.5 Ordenação - Resultados	53
Figura A.1 Manual - Tela Inicial Protótipo	64
Figura A.2 Manual - Iniciando Busca	65
Figura A.3 Manual - Visualizar Resumo	66
Figura A.4 Manual - Visualizar Relacionamento entre Entidades	66
Figura A.5 Manual - Selecionar Artigo	67
Figura A.6 Manual - Visualizar Bibtex	68
Figura A.7 Manual - Remover Artigo	69
Figura A.8 Manual - Filtrar Artigos	70
Figura A.9 Manual - Baixar Citações	71
Figura A.10 Manual - Paginação	72

LISTA DE TABELAS

Tabela 2.1	Categorias de mineração textual.....	19
Tabela 3.2	Ferramentas de RSL	23
Tabela 3.3	Ferramentas x Técnicas de mineração textual.....	24
Tabela 4.1	Pontuação Heurística	40
Tabela 5.1	RSL - Strings de Buscas	48
Tabela 5.2	Perfil dos participantes	50
Tabela 5.3	SUS Score - Interpretação	51
Tabela 5.4	Desempenho - Experimentos.....	55

LISTA DE ABREVIATURAS E SIGLAS

API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
CSS	Cascading Style Sheets
ER	EntityRelationship
ES	Engenharia de Software
ELMo	Embeddings from Language Model
GloVe	Global Vectors for Word Representation
HTML	HyperText Markup Language
IE	Information Extraction
IR	Information Retrieval
ITS	Intelligent Transportation Systems
IVi	Information Visualization
JS	JavaScript
JSON	JavaScript Object Notation
LSGN	Labeled Span Graph Network
MTV	Model-Template-View
MUC	Message Understanding Conference
MUC-6	Sexta Conferência de Compreensão de Mensagens
MVC	Model-View-Controller
NER	Named-entity recognition
NERC	Named Entity Recognition and Classification
NLP	Natural Language Processing
NoSQL	Not Only SQL
PLN	Processamento de Linguagem Natural

PPGC	Programa de Pós-Graduação em Computação
PGIE	Programa de Pós-Graduação em Informática na Educação
REST	Representational State Transfer
RSL	Revisão Sistemática da Literatura
SciERC	Scientific Entities Relations and Coreference
SciIE	Scientific Information Extractor
SLR	Systematic Literature Review
SOAP	Simple Object Access Protocol
SpERT	Span-based Entity and Relation Transformer
SQL	Structured Query Language
SRL	Semantic Role Labeling
SUS	System Usability Scale
UFRGS	Universidade Federal do Rio Grande do Sul
US	User Story
VM	Virtual Machine

RESUMO

Para realizar pesquisas científicas e acadêmicas consistentes, relevantes, atuais e com caráter inovador, é necessário identificar o que já foi feito na área (trabalhos relacionados), suas contribuições e limitações. Também com essa busca conseguimos identificar como foram feitos e quais tecnologias, métodos e técnicas foram utilizados, o que nos auxilia a criar uma solução e identificar como pode ser feito e como não deve ser feito algo conforme as informações dos trabalhos relacionados. Amplamente utilizada nas áreas da saúde, biologia e medicina, a Revisão Sistemática da Literatura (RSL) tem como objetivo apresentar uma avaliação criteriosa a respeito de um tópico de pesquisa, fazendo uso de uma metodologia de revisão que seja confiável, rigorosa e que permita auditoria. Atualmente, existem algumas ferramentas para auxiliar os pesquisadores na condução da Revisão Sistemática da Literatura. A maioria delas aplica alguma técnica de mineração textual para remover, por exemplo, estudos duplicados. Contudo, quase nenhuma delas apresenta mecanismos de rotulação semântica e filtragem de resultados com base no contexto e conceito do tópico abordado no momento da identificação e seleção. A proposta deste trabalho é desenvolver uma ferramenta web de auxílio à revisão sistemática da literatura, realizando uma classificação dos resultados encontrados em bibliotecas digitais sobre o tópico abordado, utilizando técnicas de mineração textual para filtragem dos dados com rotulação semântica sobre os resumos dos artigos e aplicando hermenêutica sobre os dados, auxiliando na seleção dos trabalhos relacionados para condução da revisão sistemática da literatura.

Palavras-chave: Revisão sistemática da literatura. rotulação semântica.

Hermeneutics: Assistance tool for carrying out systematic literature review

ABSTRACT

In order to carry out consistent, relevant, current, and innovative scientific and academic research, it is necessary to identify what has already been done in the area (related works), their contributions and limitations. Also, with this search, we can identify how they were made, what technologies, methods, and techniques were used, which helps us create a solution and identify how to do and not to do according to the information found in those works. Widely used in the area of health, biology, and medicine, the Systematic Literature Review aims to present a judicious assessment regarding a research topic, using a review methodology that is reliable, rigorous, and that allows auditing. Currently, there are some tools to assist researchers in conducting the Systematic Literature Review. For example, most of them apply textual mining techniques to remove duplicate studies. However, almost none of them have mechanisms for semantic labeling and filtering results based on the context and concept of the topic covered at the time of identification and selection. The purpose of this work is to develop a web tool to aid the systematic review of the literature, applying a classification of the results found in digital libraries on the topic covered and using textual mining techniques for filtering the data with semantic labeling on the abstracts of the articles. Hermeneutics is applied to the data and assists in selecting related works for conducting a systematic literature review.

Keywords: systematic literature review, semantic labeling.

1 INTRODUÇÃO

Para realizar pesquisas científicas e acadêmicas consistentes, relevantes, atuais e com caráter inovador, é necessário identificar o que já foi produzido na área (trabalhos relacionados, estado da arte), bem como suas contribuições e limitações acerca de um assunto. Com essa busca conseguimos, assim, identificar como os trabalhos foram feitos, quais tecnologias foram empregadas em sua construção e quais métodos e técnicas foram utilizados, o que nos auxilia a criar uma solução e identificar como pode e como não deve ser feito algo, conforme as informações dos trabalhos relacionados. Esses trabalhos relacionados nos ajudam a entender como funciona a pesquisa na área, quais são os conceitos principais e como definir os fundamentos teóricos do nosso próprio trabalho, bem como saber o que há em aberto no tópico pesquisado, escopo do trabalho e fronteira do conhecimento sobre o tema.

Amplamente utilizada em diversas áreas, em especial na saúde, biologia e medicina, a Revisão Sistemática da Literatura (RSL) tem como objetivo apresentar uma avaliação criteriosa a respeito de um tópico de pesquisa, fazendo uso de uma metodologia de revisão que seja confiável, rigorosa e que permita auditoria (KITCHENHAM, 2004). A RSL é uma etapa importante para pesquisadores na produção de artigos, monografias e demais trabalhos acadêmicos, que consiste em uma investigação científica sobre a literatura de um determinado tema, utilizando métodos sistemáticos de busca para selecionar e avaliar o conteúdo pertinente ao tema.

Na RSL, durante o processo de planejamento, são identificadas as fontes de busca e é feita a elaboração do protocolo da revisão. Já para o processo de execução, são identificados os estudos e feita a seleção dos mesmos para, enfim, extrair informações sobre o tema.

Na Computação, em Engenharia de Software, a RSL é uma tarefa necessária. Atualmente, existem algumas ferramentas para auxiliar pesquisadores na condução da RSL. Na sua grande maioria, elas aplicam alguma técnica de mineração textual para remoção de estudos duplicados, por exemplo. Entretanto, quase nenhuma das atuais ferramentas de RSL apresenta mecanismos para rotulação semântica e filtragem dos resultados baseados no contexto e conceito do tópico abordado no momento da identificação e seleção.

Além disso, bibliotecas digitais possuem mecanismos de busca baseados em palavras-chaves, e a gama de resultados encontrados é bastante elevada comparada com a quantidade de resultados realmente relevantes para o processo de RSL. Isso torna impra-

ticável que o pesquisador leia todos os documentos para obter uma visão geral e é muito elevado o tempo gasto para separar um conjunto viável de artigos.

Cabe salientar que uma RSL envolve vários pesquisadores e não apenas um, dificultando ainda mais o trabalho em teses ou pesquisas individuais, onde não é possível fazer dessa forma.

A proposta deste trabalho é desenvolver uma ferramenta web de auxílio à revisão sistemática da literatura, aplicando uma classificação dos resultados encontrados em bibliotecas digitais sobre o tópico abordado e utilizando técnicas de mineração textual para filtragem dos dados com rotulação semântica sobre os resumos dos artigos. Também almeja aplicar hermenêutica¹ sobre os dados para auxiliar na análise e seleção dos trabalhos relacionados, facilitando a condução da RSL.

A motivação para escolha do tema e proposta deste trabalho ocorreu durante minha jornada acadêmica da graduação, quando me deparei com a necessidade de buscar conteúdos sobre determinados temas das disciplinas na internet e, principalmente, em bibliotecas digitais e repositórios acadêmicos. Nesses casos, a quantidade de informação retornada não relevante era altíssima quando utilizando apenas palavras-chaves sobre o tópico e ordenação por datas. Como consequência disto, a seleção de textos para leitura, a interpretação e a compreensão mais detalhada do tema se tornavam árduas. Para esse problema recorrente têm sido desenvolvidas inúmeras propostas para facilitar a pesquisa e busca de conteúdo relevantes, tais como ontologias e sistemas baseados em conhecimento.

Combinado a esse problema de filtragem de resultados de busca, ao estudar técnicas de computação para classificação e pesquisa de dados (KNUTH; ADDISON-WESLEY, 1997; AZEREDO, 1996), foi possível observar que nesse grande ramo de pesquisa e ensino existem inúmeros algoritmos para facilitar pesquisa em dados, bem como agrupar dados para eliminação de elementos repetidos e a identificação de itens presentes num conjunto de elementos através de ordenação e categorização, que podem ser utilizados para auxiliar na condução de revisões sistemáticas da literatura.

Além dos algoritmos clássicos de classificação e pesquisa de dados, outra área de pesquisa e estudo que contribui para a pesquisa na atualidade é a Inteligência Artificial. Nas técnicas de mineração textual utilizadas para Processamento de Linguagem Natural

¹Hermenêutica é um ramo da filosofia que estuda a teoria da interpretação, que pode se referir tanto à arte da interpretação quanto à prática e treino de interpretação. Refere-se ao estudo da interpretação de textos escritos, especialmente nas áreas de literatura, religião e direito (ZIMMERMANN, 2015).

(PLN) relacionadas à análise léxica semântica, temos o Reconhecimento de Entidades Mencionadas, que tem sido aplicado para resolver problemas de classificação, ordenação e compreensão de conteúdos em motores de buscas, além de ferramentas de tradução e revisão textual da literatura (Feng; Chiam; Lo, 2017).

Por fim, tem-se como ideal facilitar e melhorar a análise de informação, utilizando métodos computacionais para fazer com que o computador leia e interprete o conteúdo e trazendo resultados mais relevantes relacionados à temática pesquisada. Nesse contexto, assim como o processo de RSL (que teve sua origem na necessidade de estudo da biologia e medicina), na área da linguística existe a Hermenêutica, definida como sendo o conjunto de teorias voltadas para a interpretação de algo, não somente um texto escrito, mas de tudo a que se possa atribuir significado e sentido, proveniente da interpretação textual (REESE, 1996) na Filosofia e Direito². A Hermenêutica foi utilizada como inspiração na pesquisa para a implementação da ferramenta para resolução do problema proposto.

Conforme descrito anteriormente, a proposta deste trabalho de conclusão é apresentar uma ferramenta de auxílio na seleção de artigos em bibliotecas digitais da Ciência da Computação e da Engenharia para a condução de RSL pertinente à área de estudo. O estudo desenvolvido e os resultados obtidos visam à aplicação de técnicas de PLN, como a Rotulação de Função Semântica para a nomeação das entidades contidas no texto e seus relacionamentos, facilitando a hermenêutica do texto e auxiliando na seleção de artigos para condução de uma RSL para contribuir, com os resultados deste estudo, para com a academia.

A estrutura deste trabalho está organizada da seguinte forma: o capítulo 2 apresenta a fundamentação teórica utilizada para desenvolver esse estudo; no capítulo 3 são apresentados os trabalhos relacionados ao tema; o capítulo 4 apresenta o estudo desenvolvido, bem como a proposta de solução e implementação em detalhes; no capítulo 5 são apresentados a metodologia de avaliação do experimento, seguido dos resultados obtidos e, por fim, no capítulo 6 são apresentadas as conclusões.

² A hermenêutica jurídica é o ramo da hermenêutica que se ocupa da interpretação das normas jurídicas, estabelecendo métodos para a compreensão legal (PAREDES, 2012).

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os principais conceitos da fundamentos teóricos relacionados com o problema e proposta de solução.

2.1 Revisão Sistemática da Literatura

A maioria das pesquisas na Engenharia de Software (ES) começa com uma revisão da literatura de algum tipo (MOLLÉRI, 2013). No entanto, a menos que uma revisão da literatura seja completa e justa, é de pouco valor científico (KITCHENHAM; CHARTERS, 2007). A finalidade de uma RSL é avaliar e interpretar todas as evidências de pesquisa disponíveis e relevantes para uma determinada questão. Estudos individuais que contribuem para uma RSL são chamados estudos primários; uma RSL é uma forma de estudo secundário.

Uma RSL é um método bem definido quanto à identificação, avaliação e interpretação de todas as pesquisas relevantes disponíveis em uma questão particular de pesquisa ou tópico ou fenômeno de interesse (KITCHENHAM; CHARTERS, 2007). A condução de uma RSL tem por objetivo apresentar uma avaliação honesta do tópico de pesquisa, à medida que utiliza um método de revisão rigoroso, confiável e passível de auditoria (KITCHENHAM; CHARTERS, 2007).

Portanto, por meio de uma RSL consegue-se construir uma síntese da pesquisa existente que seja não tendenciosa e assegura-se que o procedimento de revisão esteja visível e seja reproduzível por outros pesquisadores. A condução de uma RSL não é uma tarefa simples, visto que utiliza conceitos e termos específicos potencialmente desconhecidos para pesquisadores acostumados a realizar revisões informais (MOLLÉRI, 2013). Mesmo quando realizadas de acordo com as boas práticas, mas de forma não sistemática, carecem da falta de rigor científico na execução das suas diferentes etapas (BIOLCHINI et al., 2005).

Outra dificuldade em relação à condução de uma RSL é a execução manual da maioria de suas fases. Uma RSL é composta por três fases e cada fase possui diversas atividades. Uma fase de planejamento, onde é definido o protocolo de busca, uma de execução, onde é realizada a busca e a seleção de artigos, e outra fase de documentação, onde é estabelecido o mecanismo de divulgação e o formato do relatório (KITCHENHAM; CHARTERS, 2007).

A RSL consiste em um método específico que vai além da revisão informal e tem por objetivo combinar a pesquisa empírica para criar generalizações, que permitem ao pesquisador analisar criticamente os dados coletados, resolver conflitos detectados no material de estudo e identificar questões para pesquisas futuras (BIOLCHINI et al., 2005).

2.1.1 Detalhamento

Nesta seção será apresentado o detalhamento de uma RSL, seguindo o modelo proposto por Kitchenham (2004). Uma RSL envolve várias atividades distintas. Diretrizes existentes para revisões sistemáticas têm sugestões diferentes sobre o número e a ordem das atividades. As três fases principais são: Planejar a revisão, Executar a revisão e Documentar a revisão.

Na fase de Planejamento são identificadas as necessidades de uma revisão, que decorre da exigência dos pesquisadores de resumir todas as informações existentes sobre algum fenômeno de forma completa e de maneira imparcial. Também nesta fase é realizado o desenvolvimento de um protocolo de revisão que especifica os métodos que serão usados para realizar uma RSL específica.

Um protocolo predefinido é necessário para reduzir a possibilidade de viés do pesquisador. Por exemplo, sem um protocolo é possível que a seleção de estudos individuais ou a análise possam ser conduzidos pelas expectativas do pesquisador. No protocolo de revisão, geralmente são validados por mais de um pesquisador através de revisão por pares.

Os componentes de um protocolo incluem todos os elementos da revisão e algumas informações de planejamento adicionais (BRERETON et al., 2007):

- I *Background* ou visão geral: Contexto e justificativa para revisão.
- II Questão de pesquisa: A pergunta que a revisão visa responder.
- III Estratégia de busca: A estratégia utilizada para buscar os estudos primários, incluindo os termos de busca e bibliotecas digitais (editoras eletrônicas).
- IV Critérios de seleção dos estudos: Critérios para determinar quais estudos serão incluídos e quais serão excluídos da revisão.
- V Procedimentos para seleção dos estudos: O protocolo deve determinar como os

critérios de seleção serão aplicados, por exemplo, quantos pesquisadores irão fazer a seleção, e como divergências de escolha serão resolvidas.

- VI Procedimentos de avaliação da qualidade dos estudos: Os pesquisadores devem desenvolver um *checklist* de qualidade para avaliar os estudos individuais. O propósito da avaliação da qualidade orientará o desenvolvimento do *checklist*.
- VII Estratégias para extração dos dados: Definem como as informações requeridas de cada estudo primário serão obtidas. Se os dados exigem manipulação ou suposições a serem feitas, o protocolo deve especificar um processo de validação apropriado.
- VIII Síntese dos dados extraídos: Isso define a estratégia de síntese. Deve clarificar se uma metanálise^{1,2} formal é ou não pretendida e, em caso afirmativo, quais técnicas serão utilizadas.
- IX Estratégia de disseminação: Necessário se ainda não foi definida na identificação da revisão.
- X Cronograma do projeto: Define a agenda da revisão.

Os estágios associados à condução da revisão são: identificação da pesquisa, seleção dos estudos primários, avaliação da qualidade dos estudos e extração e sintetização dos dados.

A fase de documentação possui um único estágio, que consiste em especificar os mecanismos de divulgação e formatar o relatório principal para a avaliação do relatório.

Os estágios listados acima podem parecer sequenciais, mas é importante reconhecer que muitos dos estágios envolvem iteração. Em particular, muitas atividades são iniciadas durante o estágio de desenvolvimento do protocolo e refinado quando a revisão propriamente dita começa a ocorrer.

Por exemplo, a seleção de estudos primários é regida por critérios de inclusão e exclusão. Esses critérios são especificados inicialmente, quando o protocolo é definido, mas podem ser refinados após a definição dos critérios de qualidade. Os formulários de extração de dados inicialmente preparados durante a construção do protocolo poderão ser alterados quando forem acordados critérios de qualidade. Os métodos de síntese de dados

¹Historicamente, a metanálise já foi confundida com a metodologia da revisão sistemática, quando Glass, autor do termo metanálise, em 1976, utilizou-o para definir o processo identificado atualmente por revisão sistemática (GLASS, 1976).

²Metanálise é a análise estatística de resultados de diferentes estudos individuais, com o objetivo de integrá-los, combinando e resumindo seus resultados (SACKS et al., 1996).

definidos no protocolo podem ser alterados quando os dados forem coletados (KITCHENHAM, 2004).

2.2 Técnicas de mineração textual

Mineração de textos é o processo de descoberta de conhecimento que utiliza técnicas de análise e extração de dados a partir de textos, frases ou palavras. É o processo de extrair padrões interessantes e não triviais ou conhecimento a partir de documentos em textos não estruturados (SEBASTIANI, 2001). Esta descoberta de conhecimento envolve diversas aplicações, tais como análise de textos, extração de informações, sumarização, classificação, agrupamentos, linguística computacional, dentre outras.

A seguir são apresentadas as principais técnicas de mineração textual empregadas em ferramentas de auxílio à RSL segundo estudo realizado por Feng, Chiam e Lo (2017).

Tabela 2.1: Categorias de mineração textual

Categoria	Descrição
Extração de Informação (IE)	Encontrar um pedaço específico de informações a partir de um documento de texto, usando um método de correspondência de padrões de frases-chave e relações com o texto.
Recuperação de Informação (IR)	Investigação de mecanismos adequados para a busca de informações interessantes a partir de uma coleção de recursos.
Visualização de Informação (IVi)	Visualização de informações usando representações visuais interativas para amplificar reconhecimento humano.
Classificação (Categorização)	Encontrar padrões/características que os tornam parte de um definido agrupamento e atribuição de documentos para categorias conhecidas interessantes.
<i>Clustering</i> (Agrupamento)	Encontrar padrões interessantes associados com dados extraídos e agrupar documentos semelhantes com base em seu conteúdo.
Sumarização	Reduzir o comprimento e detalhes do texto original em uma versão mais curta, preservando o significado implícito de sua informação.

Fonte: (Feng; Chiam; Lo, 2017).

2.3 Natural Language Processing

Natural Language Processing (NLP), em português chamada de Processamento de Linguagem Natural e também conhecida como Linguística Computacional, é uma subárea da Ciência da Computação, da Inteligência Artificial e da Linguística que estuda os problemas da geração e compreensão automática de línguas humanas naturais.

Antigamente, muitos sistemas de processamento de linguagem eram projetados codificando manualmente um conjunto de regras (WINOGRAD, 1971), por exemplo, escrevendo gramáticas ou criando regras heurísticas para o *stemming*³.

Desde a chamada “revolução estatística” (JOHNSON, 2009), no final dos anos 80 e meados da década de 90, muitas pesquisas sobre processamento de linguagem natural dependem muito do aprendizado de máquina. Em vez disso, o paradigma de aprendizado de máquina exige o uso de inferência estatística para aprender automaticamente essas regras através da análise de grandes *corpora*⁴.

Dentro das inúmeras aplicações da NLP nos dias de hoje, temos na divisão semântica o reconhecimento de entidades no texto (comumente chamada de NER) e rotulações semânticas. Nas subseções seguintes serão detalhadas as duas principais técnicas de NLP empregadas neste trabalho.

2.3.1 *Named-Entity Recognition*

O reconhecimento de entidades mencionadas é um ramo do processamento de linguagem natural que procura extrair e classificar as entidades mencionadas em um texto escrito em linguagem natural (GRISHMAN; SUNDHEIM, 1996).

O termo Entidade Nomeada, agora amplamente usado na PLN, foi criado para a Sexta Conferência de Compreensão de Mensagens (MUC-6)⁵. Naquela época, a MUC estava concentrada nas tarefas de Extração de Informações (*Information Extraction – IE*), nas quais informações estruturadas das atividades da empresa e atividades relacionadas à defesa são extraídas de texto não estruturado, como artigos de jornal.

Ao definir a tarefa, as pessoas notaram que é essencial reconhecer unidades de informação como nomes, incluindo pessoas, nomes de organizações e locais e expressões numéricas, incluindo hora, data, dinheiro e expressões percentuais. A identificação de referências a essas entidades no texto foi reconhecida como uma das subtarefas importantes da IE, e foi denominada “*Reconhecimento de Entidades Nomeadas e Classificação*” (*Named Entity Recognition and Classification – NERC*), de acordo com Nadeau e Sekine

³Em morfologia linguística e recuperação de informação, a “stemização” (do inglês, *stemming*) é o processo de reduzir palavras flexionadas (ou às vezes derivadas) ao seu tronco (*stem*), base ou raiz, geralmente uma forma da palavra escrita (LOVINS, 1968).

⁴Plural de *corpus*, é um conjunto de documentos, possivelmente com anotações humanas ou de computador) de exemplos típicos do mundo real.

⁵MUC-6, *the sixth in a series of Message Understanding Conferences*, disponível em <<https://cs.nyu.edu/~grishman/muc6.html>>

(2007).

2.3.2 *Semantic Role Labeling*

No processamento de linguagem natural, a rotulação semântica é o processo que atribui rótulos a palavras ou frases, indicando seu papel semântico na frase, como a de um agente, objetivo ou resultado.

Semantic Role Labeling (às vezes abreviada como SRL) é uma tarefa da NLP que fornece os meios para analisar, do ponto de vista semântico, as informações expressas através de texto ou fala. O objetivo é capturar e representar os participantes e as circunstâncias de eventos ou situações descritas no nível sentencial.

A SRL é a tarefa de encontrar, automaticamente, as funções semânticas de cada argumento, de cada predicado em uma frase. As abordagens atuais para a rotulação de funções semânticas são baseadas no aprendizado de máquina supervisionado (GILDEA; JURAFSKY, 2000). É a relação que um constituinte sintático tem com um predicado.

Os argumentos semânticos típicos incluem agente, paciente, instrumento etc. e também argumentos auxiliares indicando aspectos locativos, temporais, de maneira, de causa etc. Reconhecer e rotular argumentos semânticos é uma tarefa essencial para responder perguntas sobre “Quem”, “Quando”, “O que”, “Onde”, “Por que” etc. nas perguntas de extração de informações, resposta a perguntas, resumo e, em geral, em todas as Tarefas de NLP nas quais é necessário algum tipo de interpretação semântica (CARRE-RAS; MÀRQUEZ, 2005).

2.4 Resumo do Capítulo

Como visto neste capítulo, a condução de uma RSL possui modelo e métodos bem definidos no seu processo e, tendo em vista a complexidade e esforço necessário para realização de uma RSL, existem inúmeras técnicas de mineração textual que podem ser utilizadas em ferramentas para auxiliar os pesquisadores na realização de RSL. Por fim, na NLP temos a NER e a SLR que visam rotular/extrair informações relevantes no texto e podem ser empregadas na seleção e interpretação de artigos que serão incluídos ou excluídos na revisão.

3 TRABALHOS RELACIONADOS

Neste capítulo, é apresentada uma revisão de ferramenta de auxílio na condução de RSL e artigos sobre mineração textual utilizados no presente trabalho.

3.1 Ferramentas de Revisão Sistemática da Literatura

Conforme mencionado anteriormente, há inúmeras ferramentas para auxiliar em todo o processo de revisão sistemática da literatura, algumas delas *desktop* outras *web*. Para fins de análise comparativa entre as ferramentas existentes, para auxiliar na realização de revisão sistemática a seguir, são apresentados os critérios comparativos e a tabela contendo as ferramentas analisadas:

- **Bases de dados:** conjunto de bibliotecas digitais e repositórios on-line nos quais a ferramenta tem acesso aos dados de artigos, revistas e livros para revisão. Se for possível, pesquisar e coletar artigos através da ferramenta nas bases de dados que a ferramenta possui acesso;
- **Plataforma:** relacionada à categoria do ambiente onde a ferramenta é utilizada, podendo ser *desktop* instalada localmente na máquina do pesquisador ou *web*, sendo acessada na internet através do navegador;
- **Palavras-Chave:** relacionadas à *string* de busca da revisão sistemática e a filtros para catalogar os documentos selecionados para revisão;
- **String de busca:** funcionalidades relacionadas à manipulação da *string* de busca
- **Filtros:** funcionalidade de filtragem de dados e resultados;
- **Análise Semântica:** se a ferramenta aplica alguma técnica de mineração textual (NLP/SRL) para análise semântica dos documentos.

Tabela 3.2: Ferramentas de RSL

ID	Ferramenta	Bases de Dados	Plataforma	Palavra Chave	String de Busca	Filtros	Análise Semântica
0	StArt	Não automatiza a busca	Desktop	Não	Não	Não	Não
1	ARS/SESRA	IET; Springer; IEEE xplore	WEB	Sim	Sim	Não	Não
2	SLuRp	Não automatiza a busca	WEB	Não	Não	Não	Não
3	SLRTOOL	Não definido	WEB	-	-	-	-
4	SLR-Tool	Não automatiza a busca	Desktop	Não	Não	Não	Não
5	Metta	PubMed; EM-BASE CINAHL; PsycINFO; Cochrane Central; Register of Controlled Trials;	WEB	Sim	Não	Sim	Não
6	EPPI-Reviewer	PubMed	WEB	Sim	Não	Sim	Sim
7	DistillerSR	PubMed	WEB	Sim	Não	Não	Não
8	Researchr	Não automatiza a busca	WEB	Não	Não	Não	Não
9	RESuLT	ACM; IEEE xplore; Science-Direct; Scopus; Springer	WEB	Sim	Sim	Sim	Não
10	Portal Periódicos CAPES/-MEC	532 bases cadastradas	WEB	Sim	Sim	Sim	Não
11	Parsifal	Não automatiza a busca	WEB	Sim	Sim	Não	Não
12	Hermeneutics	ArXiv; IEEE xplore; Science-Direct; Scopus; Springer	WEB	Sim	Sim	Sim	Sim

Fonte: Adaptado de (CORDOVA, 2017).

Na última posição da Tabela 3.2 encontra-se a ferramenta desenvolvida neste trabalho que, diferente das demais, possui o enfoque apenas na seleção dos trabalhos relacionados e aplicação de análise semântica na classificação dos resultados encontrados para auxiliar o pesquisador no processo de realização da revisão sistemática, processo detalhado no capítulo seguinte.

3.1.1 Técnicas de mineração textual utilizadas

Apesar da importância de realizar RSL para identificar as lacunas da pesquisa em engenharia de software (*Software Engineering* – SE), é um processo complexo e demorado, se realizado manualmente. A condução de uma RSL de acordo com as diretrizes e práticas no domínio SE requer um esforço e conhecimento consideráveis (Feng; Chiam; Lo, 2017).

Dentre as 32 ferramentas analisadas por Feng, Chiam e Lo (2017) cinco delas fazem parte do estudo realizado por Cordova (2017) e das ferramentas listadas apenas uma delas possui mecanismo de análise semântica. Na Tabela 3.3 apresenta-se a relação de técnicas utilizadas pelas ferramentas em comum dos dois estudos.

Tabela 3.3: Ferramentas x Técnicas de mineração textual

ID	Ferramenta	Categoria	Algoritmos / Modelos / Teorias
0	StArt	IVi e IE	Vector Processing Model
2	SLuRp	N / D	N / D
3	SLR-Tool	IR, Clustering	Não especificou quais foram utilizadas
5	Metta	IR, IE	Federated search model
6	EPPI-Reviewer	Clustering	Clustering algorithm: lingo; information extraction algorithms: C-value; ML classification algorithm: SVM classier

Fonte: (Feng; Chiam; Lo, 2017).

É importante observar que as ferramentas de revisão sistemática na sua grande maioria não possuem tratamentos semânticos dos dados e este fica manualmente aos encargos dos pesquisadores envolvidos.

3.1.2 Sobek

Na linha de ferramentas para auxiliar pesquisadores, a ferramenta Sobek foi desenvolvida pelo grupo Gtech de pesquisa da UFRGS e foi utilizada como referencial teórico na elaboração da ferramenta proposta neste trabalho.

Inicialmente, percebeu-se a necessidade de auxiliar o professor no acompanhamento e gerenciamento de grandes volumes de produções textuais à distância (MACEDO et al., 2009). Assim, a ferramenta Sobek foi desenvolvida com o objetivo de fornecer aos professores instrumentos para auxiliá-los na análise de produções textuais dos estudantes (REATEGUI et al., 2016).

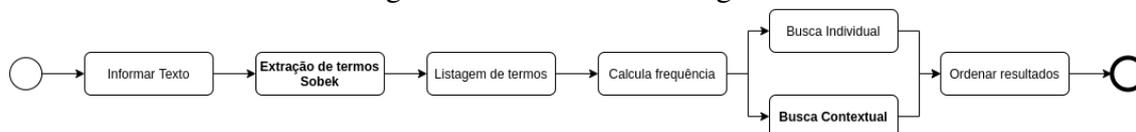
A ferramenta Sobek utiliza um algoritmo definido por Schenker et al. (2005), que

se baseia em análises estatísticas dos textos e representa as informações extraídas em um modelo de grafo de distância n -simples, baseado na ideia de que cada palavra relevante no texto deve estar conectada a N palavras relevantes subsequentes (LORENZATTI, 2009). Assim, o usuário pode utilizar uma interface gráfica simples e amigável, onde ele pode escrever ou colar um texto que será minerado e terá os principais termos extraídos. A seguir, após apertar o botão de extração (Extrair Conceitos), ocorre a representação gráfica desses termos e suas relações, sendo os termos os vértices (os maiores são os de maior frequência) e as relações as arestas, juntamente com a demonstração, ao clicar em cima do termo no grafo, de suas ocorrências no texto.

A partir da representação por grafo, é possível ter-se uma ideia de como a ferramenta pode auxiliar o usuário na compreensão do texto. A sequência de termos apresentada possibilita a identificação de relações entre estes, levando o usuário a refletir sobre o texto por meio da interpretação do grafo (LORENZATTI, 2009).

Em Jacobs 2016 foi utilizada a ferramenta Sobek para identificar informações relevantes em buscas *online*, seguindo o fluxo de execução descrito abaixo conforme a Figura 3.1:

Figura 3.1: SOBEK - Fluxograma



Fonte: Autor.

O estudo desenvolvido por Jacobs (2016) conforme ilustrado na Figura 3.1 possui o seguinte conjunto de etapas descritos abaixo:

1. Usuário Informa o texto;
2. Ferramenta Sobek extrai os principais termos;
3. Com a lista de principais termos é obtido o contexto do texto;
4. O contexto é utilizado para ordenar todos os termos extraídos conforme a frequência no texto;
5. Utilizando o mecanismo de busca na internet (Google)
 - 5.1. Individual = Para cada termo é feita a busca utilizando o termo como palavra-chave;

- 5.2. Contextual = Para cada termo é feito a busca utilizando o termo como palavra-chave *E* os termos principais como o contexto;
6. Ordena a lista de resultado a partir da razão: número de resultados COM Contexto / número de resultados SEM contexto.

3.2 Frameworks de reconhecimento NER/SRL

Para a identificação de entidades e relacionamentos, a ferramenta desenvolvida neste trabalho utilizou como base o estado da arte do reconhecimento de entidades nomeadas, que vem sendo aprimorada anualmente através de técnicas e métodos de *Deep Learning* voltados para NLP.

Nesta seção serão apresentados os principais trabalhos que alicerçaram a ferramenta proposta. Em (LUAN et al., 2018) foi desenvolvido um modelo de aprendizado unificado para extrair entidades científicas, relações e resolução de correferências. Para explorar esse problema, foi criado um conjunto de dados, chamado SciERC, para extração de informações científicas, que inclui anotações de termos científicos, categorias de relações e links de correferência. Os experimentos realizados mostraram que o modelo unificado é melhor na previsão de limites de *span*¹ e supera os sistemas científicos avançados de IE anteriores sobre extração de entidades e relações.

De encontro com a proposta da ferramenta desenvolvida neste trabalho, o *framework* desenvolvido chamado SciIE possui a motivação de agregar valor às Revisões Sistemáticas da Literatura:

À medida que as comunidades científicas crescem e evoluem, novas tarefas, métodos e conjuntos de dados são introduzidos e métodos diferentes são comparados entre si. Apesar dos avanços nos mecanismos de busca, ainda é difícil identificar novas tecnologias e seus relacionamentos com o que existia antes. Para ajudar mais os pesquisadores a identificar rapidamente oportunidades para novas combinações de tarefas, métodos e dados, é importante projetar algoritmos inteligentes que podem extrair e organizar informações científicas de uma grande coleção de documentação (LUAN et al., 2018).

Dando continuidade aos trabalhos de modelos de extração de informações em (EBERTS; ULGES, 2019), um novo *framework* foi desenvolvido utilizando as redes *transformers* BERT para a extração informações de artigos científicos. Nas subseções seguintes serão apresentados em detalhes os *frameworks* citados e no capítulo seguinte como eles foram utilizados no trabalho proposto.

¹ Sequência ordenada de *tokens*.

3.2.1 SciERC e SciIE

Scientific Information Extractor (SciIE) é uma configuração multitarefa (*framework*) de identificação de entidades, relações e agrupamentos de centralizações em artigos científicos. É um conjunto de dados que inclui anotações para todas as três etapas e desenvolvemos uma estrutura unificada com representações de span compartilhadas. A configuração de várias tarefas reduz erros de cascata e aproveita as relações entre frases através de links de correferência. Esse *framework* estende modelos anteriores baseados em *span* para resolução de correferência e rotulação de função semântica.

O *SciIE* se beneficia de representações contextuais expressivas de amplitude como recursos do classificador. Ao compartilhar representações de amplitude, as tarefas no nível da frase podem se beneficiar das informações propagadas da resolução da correferência entre as frases, sem aumentar a complexidade da inferência.

SciERC é um conjunto de dados (*dataset/benchmark*) que inclui anotações para entidades científicas, suas relações e *clusters* de núcleo para 500 resumos científicos. Esses resumos foram retirados de 12 conferências/*workshops* da AI em quatro comunidades da AI, do *Semantic Scholar Corpus*. *SciERC* estende conjuntos de dados anteriores em artigos científicos SemEval 2017 Task 10 e SemEval 2018 Task 7 estendendo tipos de entidades, tipos de relação, cobertura de relações e adição de relações entre frases usando links de correferência. Esse *dataset* foi anotado por um especialista na área e cerca de 12% dos dados foram anotados duplamente por outros 4 especialistas. O *SciERC* foi desenvolvido com foco na anotação de relações entre sentenças e tem mais cobertura de relações que os *datasets* antecessores.

Para esse *dataset* e *framework* de extração de informação, foi um esquema de anotação dividido em dois conjuntos de categorias (tipos), um para as entidades e outro para os relacionamentos, detalhados no apêndice² de (LUAN et al., 2018).

3.2.1.1 Categoria Entidade

1. **Task:** Aplicações, problemas a resolver ou sistemas a construir.

Por exemplo: extração de informações, sistema de leitura de máquina, segmentação de imagens etc.

2. **Method:** Métodos, modelos, sistemas a serem utilizados, ou ferramentas, compo-

²Annotation Guideline <http://nlp.cs.washington.edu/sciIE/annotation_guideline.pdf>.

nentes de um sistema, estruturas.

Por exemplo, modelo de linguagem, CORENLP, POS parser, kernel, etc.

3. **Evaluation Metric:** Métricas, medidas ou entidades que podem expressar a qualidade de um sistema/método.
Por exemplo, F1, BLEU, Precisão, Recall, Curva ROC, classificação recíproca média, erro ao quadrado médio, robustez, complexidade de tempo, etc.
4. **Material:** Dados, conjuntos de dados, recursos, Corpus, base de conhecimento.
Por exemplo, dados de imagem, dados de fala, imagens estéreo, dicionário bilíngue, perguntas parafraseadas, CoNLL, Panntreebank, WordNet, Wikipedia, etc.
5. **Other Scientific Terms:** Frases que são termos científicos, mas não se enquadram em nenhuma das classes anteriores. Por exemplo, restrições físicas ou geométricas, conhecimento prévio qualitativo, estrutura do discurso, regra sintática, estrutura do discurso, árvore, nó, núcleo da árvore, características, ruído, critérios.
6. **Generic:** Termos ou pronomes gerais que podem se referir a uma entidade, mas não são eles mesmos informativos, frequentemente usado como palavras de conexão.
Por exemplo, modelo, abordagem, conhecimento prévio.

3.2.1.2 Categoria Relacionamento

Para cada entidade nomeada no texto, foram definidas as categorias que relacionam a entidade A com a entidade B . A relação entre as entidades está limitada a uma sentença. Foram definidos 4 tipos de relação assimétrica (*Used-for*, *Feature-of*, *Hyponym-of* e *Part-of*), juntamente com 2 tipos de relações simétricas (*Compare*, *Conjunction*). B sempre aponta para A , em relações assimétricas.

1. **Used-for:** B é usado para A , B modela A , A é treinado em B , B explora A , A é baseado em B ;
2. **Feature-of:** B pertence a A , B é uma característica de A , B está no domínio de A ;
3. **Hyponym-of:** B é um hipônimo de A , B é um tipo de A ;
4. **Part-of:** B é uma parte de A ;
5. **Compare:** compara dois modelos / métodos ou listando duas entidades opostas;

6. **Conjunction:** Conjunção, Função como regra semelhante ou uso / incorporação com.

O trabalho foi desenvolvido em Python 2.7 usando TensorFlow³ estendendo LSGN (Labeled Span Graph Network) e o e2e-coref (Higher-order Coreference Resolution with Coarse-to-fine Inference) e também ELMo⁴ e GloVe embeddings⁵.

Além do Framework e Dataset adicionalmente foi construído um grafo do conhecimento com os resultados extraídos. Como resultado o SciIE obteve F1 de 64.2% na predição do dataset SciERC.

3.2.2 SpERT

O trabalho "*Span-based Joint Entity and Relation Extraction with Transformer Pre-training*" (SpERT) investiga o uso de redes Transformer para extração de dados: Dado um conjunto predefinido de relações-alvo e uma frase (EBERTS; ULGES, 2019).

O SpERT, um modelo baseado em span para extração conjunta de entidades e relações que depende da rede pré-treinada da Transformer BERT⁶ como seu núcleo. Neste trabalho, foi mostrado que, com forte amostragem negativa, filtragem de amplitude e uma representação de contexto localizada, uma pesquisa em todos os períodos de uma sentença de entrada se torna possível. Os resultados obtidos sugerem que as abordagens baseadas em extensão são competitivas em relação aos modelos baseados em BILOU (B - beginning , I - inside, L - last, O - outside e U - unit) e podem ser a abordagem mais promissora para pesquisas futuras devido à sua capacidade de identificar entidades sobrepostas.

Alguns diferenciais do *framework* é que os modelos foram treinados com todo o texto e não apenas o *abstract*, também possui um dicionário de palavras maior que utilizado pelo SciIE. Com contextualização local por frase, o treinamento dos modelos é supervisionado. Dadas sentenças com entidades anotadas (incluindo seus tipos de entidade) e relações, foi uma função de perda conjunta para classificação e relação de entidade e amostras negativas da mesma frase produzem que produziram treinamento que seja eficiente e eficaz.

A seguir, na Figura 3.2 é exibida a estrutura do modelo, conforme definido por

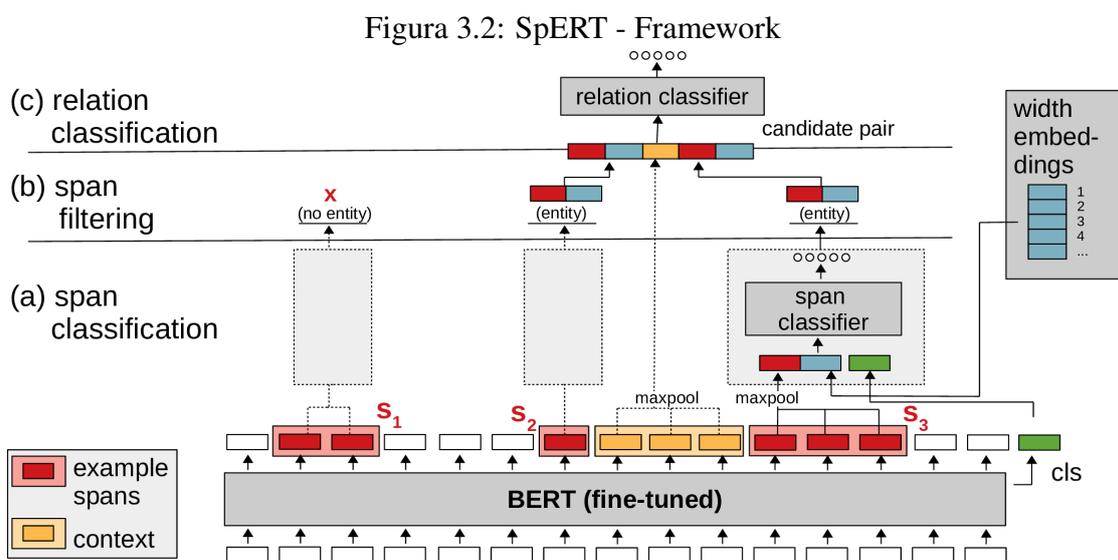
³TensorFlow <<https://www.tensorflow.org/?hl=pt-br>>.

⁴Embeddings from Language Model <<https://allennlp.org/elmo>>.

⁵Global Vectors for Word Representation <<https://nlp.stanford.edu/projects/glove/>>.

⁶BERT <<https://nlp.stanford.edu/seminar/details/jdevlin.pdf>>.

Eberts e Ulges (2019). Na abordagem de extração de entidades e relacionamento do framework SpERT, primeiramente o texto passa uma sequência de tokenização através do BERT. Então, em (a) todos os *spans* dentro da frase são classificados em tipos de entidade, conforme ilustrado para três exemplos de *spans* s_1 , s_2 , s_3 (em vermelho). Em (b), *spans* classificados como "não entidades"(aqui, s_1) são filtrados. Em (c), todos os pares de entidades restantes (aqui, (s_2 , s_3)) são combinados com seu contexto (o intervalo entre as entidades, em amarelo) e classificados em relações.



Fonte: (EBERTS; ULGES, 2019).

O trabalho foi desenvolvido em Python 3 usando PyTorch⁷, através de *transformers* (VASWANI et al., 2017), que são modelos de aprendizado profundo que adotam o mecanismo de atenção, pesando diferencialmente a importância de cada parte dos dados de entrada; combinado com o *sentencepiece*⁸, que consistem em um tokenizador e um destokenizador de textos, não supervisionados, principalmente para sistemas de geração de textos baseados em Redes Neurais, onde o tamanho do vocabulário é predeterminado antes do treinamento do modelo neural, em redes BERT pré-treinadas, conforme modelo da Figura 3.2. Como resultado, o SpERT obteve F1 de 70.33% na predição do *dataset* SciERC⁹.

⁷PyTorch <<https://pytorch.org/>>.

⁸SentencePiece <<https://github.com/google/sentencepiece>>.

⁹F1 Score SciERC <<https://paperswithcode.com/sota/named-entity-recognition-ner-on-scierc>>.

4 ESTUDO DESENVOLVIDO

Neste capítulo é apresentada o detalhamento da implementação da ferramenta desenvolvida no presente trabalho.

4.1 Solução Proposta

Nesta seção, é apresentada a especificação da ferramenta desenvolvida no presente trabalho, intitulada *Hermeneutics*. Como parte do estudo desenvolvido, foram analisados o conjunto de ferramentas de revisão sistemática da literatura e as técnicas de mineração que cada uma delas utiliza. Durante esse levantamento, o escopo deste trabalho foi limitado apenas para a etapa de seleção de artigos para realização do processo de revisão sistemática da literatura.

Conforme identificado, as ferramentas existentes não possuem mecanismos e técnicas de mineração textual relacionadas à filtragem e análise semântica e neste trabalho foi implementada uma ferramenta web para busca e seleção de artigos. A ferramenta coleta artigos de cinco bibliotecas digitais relacionadas à área de Computação, a saber: arXiv¹, IEEE², Scopus³, Science Direct⁴ e Springer⁵. As bibliotecas digitais foram escolhidas conforme a disponibilidade de acesso aos artigos através de serviço web com documentação de uso e optou-se por utilizar REST API para fazer as consultas, pois todas bibliotecas digitais selecionadas ofereciam documentação necessária para implementação da proposta e para o uso do serviço. Dos artigos são coletados o título, informações sobre autores, DOI (ou identificador único) e resumo.

Para fazer a coleta dos artigos, a ferramenta utiliza a *string* de busca definida pelo pesquisador, conforme a temática da revisão sistemática. Em seguida, essa *string* é submetida às respectivas APIs das bibliotecas mencionadas e os dados dos artigos resultantes são coletados. Após a coleta, os dados são processados através do *framework* SpERT para a obtenção de Entidades e Relacionamentos presentes no texto do resumo dos artigos. Com as devidas Entidades e Relacionamentos dos artigos, a ferramenta disponibiliza a filtragem e visualização das entidades e relacionamentos para auxiliar o pesquisador na

¹<https://arxiv.org/help/api/>.

²<https://developer.ieee.org/>.

³https://dev.elsevier.com/sc_apis.html.

⁴https://dev.elsevier.com/sd_apis.html.

⁵<https://dev.springernature.com/>.

seleção dos artigos.

4.1.1 Requisitos

Logo abaixo são listados os requisitos funcionais e não funcionais da ferramenta:

4.1.1.1 Funcionais

- Os usuários devem poder informar *strings* de buscas genéricas usando ANDs, ORs, parênteses e palavras chaves entre aspas;
- Os usuários devem poder extrair a citação/referência bibliográfica dos artigos selecionados.

4.1.1.2 Não-Funcionais

- O sistema deve ser implementado em arquitetura WEB para suportar a mobilidade do usuário e acesso em múltiplas plataformas;
- O sistema deve permitir o acesso simultâneo de usuários;
- O tempo de resposta do sistema não deve ultrapassar 10 minutos.

4.1.2 User Stories

A seguir é apresentado o conjunto de *Users Stories* idealizado para a ferramenta proposta.

- **US01 - Pesquisar Artigos**

Como pesquisador, eu quero procurar por artigos para condução da RSL informando a String de Busca do Tópico da pesquisa para que seja possível selecionar artigos.

- **US02 - Visualizar Resumos**

Como pesquisador, eu quero visualizar o resumo dos artigos pesquisados exibindo as anotações de entidades e relacionamentos no texto para que seja possível escolher quais artigos irão fazer parte da RSL.

- **US03 - Selecionar Artigo**

Como pesquisador, eu quero escolher os artigos que farão parte da RSL e exibí-los separadamente dos demais para que seja possível baixar as citações dos mesmos.

- **US04 - Visualizar Citação**

Como pesquisador, eu quero visualizar a citação dos artigos no formato Bibtex para que possa selecionar os artigos com base na informação da citação.

- **US05 - Filtrar Artigos**

Como pesquisador, eu quero poder filtrar a listagem de artigos conforme as entidades e relacionamentos existentes no texto dos resumos para auxiliar na seleção dos artigos para a condução da RSL.

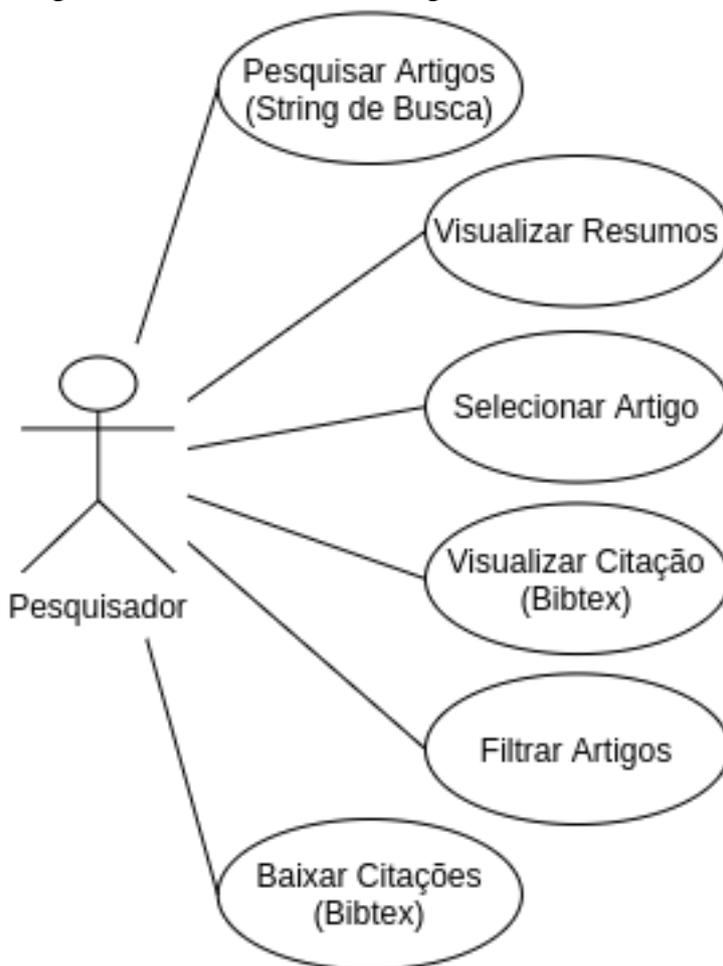
- **US06 - Baixar Citações**

Como pesquisador, ao final do processo eu quero que seja possível baixar em um único arquivo todas as citações dos artigos selecionados, para que seja utilizado no restante da RSL.

4.1.2.1 Casos de Uso

Para o conjunto de User Stories detalhado anteriormente foi construído o digrama de casos de uso representado pela Figura 4.1.

Figura 4.1: Hermeneutics - Diagrama de casos de uso



Fonte: Autor.

4.2 Implementação

Nesta seção, serão apresentados os principais componentes da implementação da ferramenta, a saber: seu fluxo de funcionamento, estruturas e modelos, algoritmo de ranqueamento aplicado nos resultados e, por fim, a arquitetura da ferramenta.

4.2.1 Fluxo principal da ferramenta

Nesta seção, é apresentado o detalhamento de cada etapa do processo e, logo a seguir, o fluxo principal da ferramenta implementada, conforme Figura 4.2.

1. **Informar *string* de busca:** o usuário/pesquisador informa sua *string* de busca usando os operadores lógicos AND, OR, usando Aspas Duplas para os termos com mais de uma palavra e, opcionalmente, parênteses para comutação e associação booleana dos termos;
2. **Consultar Bibliotecas Digitais:** para cada uma das bibliotecas, a *string* de busca é tratada e são feitas requisições REST para as API. Resultados encontrados são salvos em arquivo JSON contendo título, autor, *abstract*, DOI e identificador da biblioteca de onde foi obtido;
3. **Prever Entidades e Relacionamentos:** o JSON da etapa anterior é utilizado como entrada do *framework* SpERT que processa os *abstracts* na rede neural para a obtenção das entidades e relacionamentos do texto;
4. **Filtragem e Análise dos Artigos:** após a obtenção de entidades e relacionamentos, o pesquisador (usuário) pode aplicar filtros, detalhados na subseção seguinte, para interpretação dos artigos, conforme as anotações semânticas das entidades e relacionamentos contidos no resumo;
5. **Seleção dos artigos:** depois de analisados e interpretados os artigos utilizando os filtros da ferramenta, o usuário pode baixar as referências dos artigos individualmente e/ou exportar a lista de artigos incluídos e excluídos para a condução da RSL.

Figura 4.2: Hermeneutics - Fluxograma Principal



Fonte: Autor.

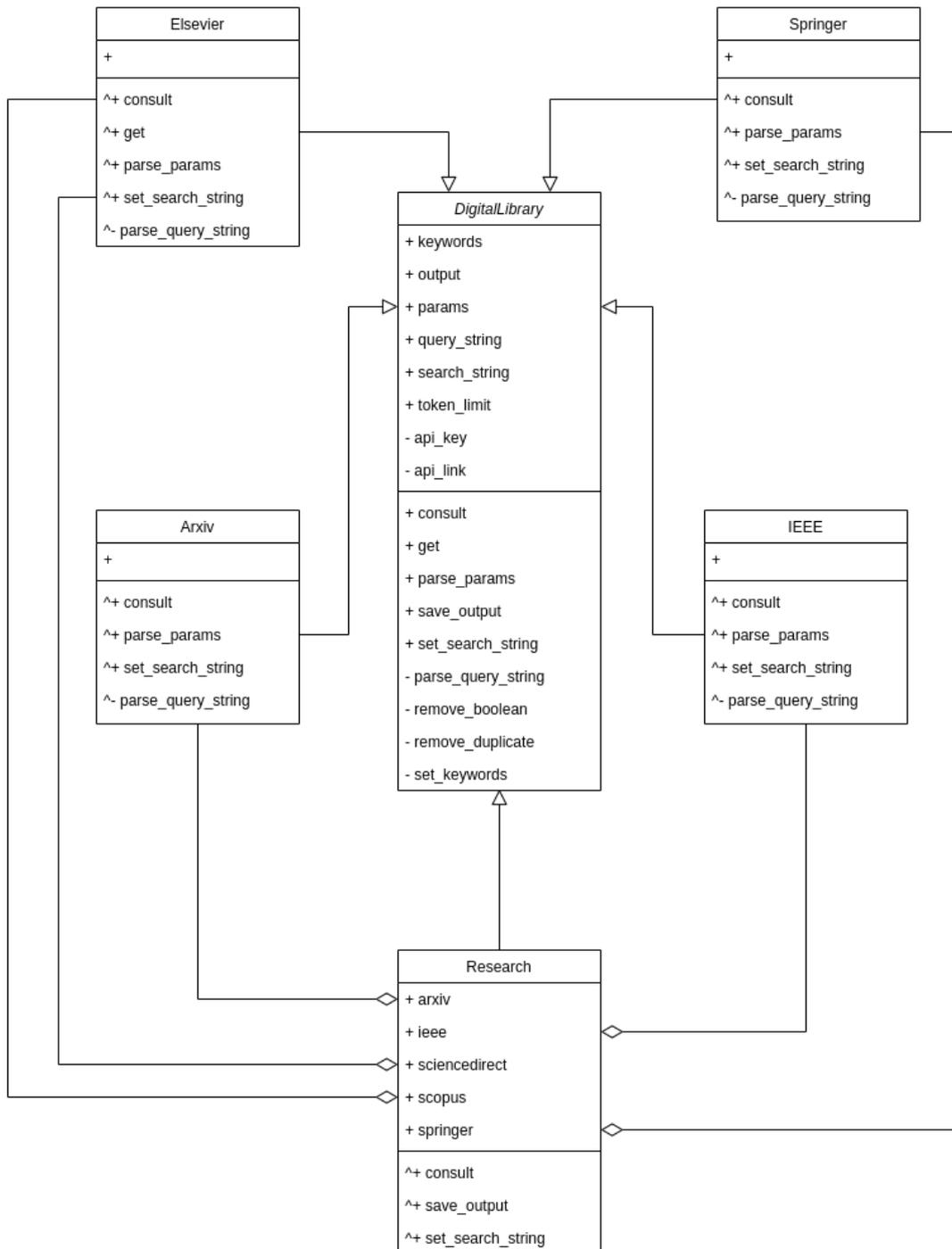
4.2.2 Estruturas e modelos

Nesta seção são apresentados os principais diagramas de classes da implementação, bem como o Diagrama ER do banco de dados da ferramenta.

4.2.2.1 Bibliotecas Digitais

Para fazer a gerência de consultas e dados provenientes das bibliotecas digitais, foi implementado um conjunto de classes conforme a Figura 4.3.

Figura 4.3: Research - Diagrama de Classes



Fonte: Autor.

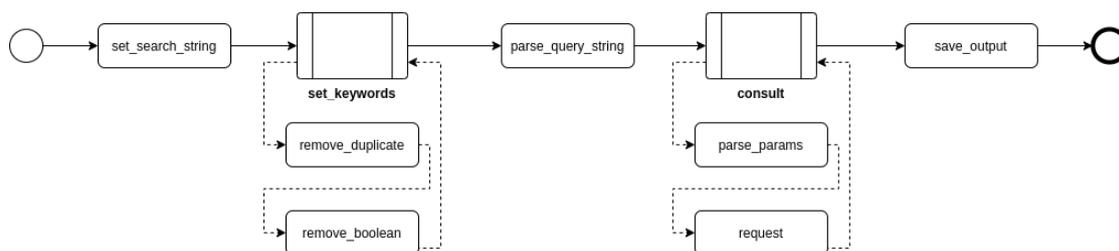
Na classe DigitalLibrary são generalizados todos os atributos e métodos que uma biblioteca digital precisa ter para fazer consultas via REST API das respectivas class-

ses especializadas *Arxiv*, *Elsevier*, *IEEE* e *Springer*. Nas classes herdeiras, são feitas as sobrescritas dos métodos genéricos conforme a especificação do webservice REST de cada biblioteca digital. Por fim, na classe *Research*, que herda os métodos e atributos da classe genérica *DigitalLibrary*, do mesmo modo que as classes especializadas, porém centralizando as chamadas de métodos das demais classes herdeiras através de atributos de agregação, conforme exemplificado na Figura 4.3.

De maneira geral, um objeto que estende a classe *DigitalLibrary* trabalha sobre o modelo *Cliente-Servidor*. Antes no cliente, no caso o objeto da classe *Research*, enviar a requisição para o servidor uma série de tratamentos na *string* de busca são feitos conforme listagem seguinte:

1. **set_search_string**: armazena a *string* de busca utilizada na consulta.
2. **set_keywords**: a partir da *string* de busca informada são extraídas as palavras chaves.
3. **parse_query_string**: converte a *string* de busca genérica nas *strings* de consulta específica de cada biblioteca digital, conforme documentação da respectiva REST API.
4. **consult**: onde é feita a requisição GET ao serviço web da biblioteca digital.
5. **save_output**: extrai os resultados em JSON para utilização na ferramenta.

Figura 4.4: DigitalLibrary - Fluxograma simplificado



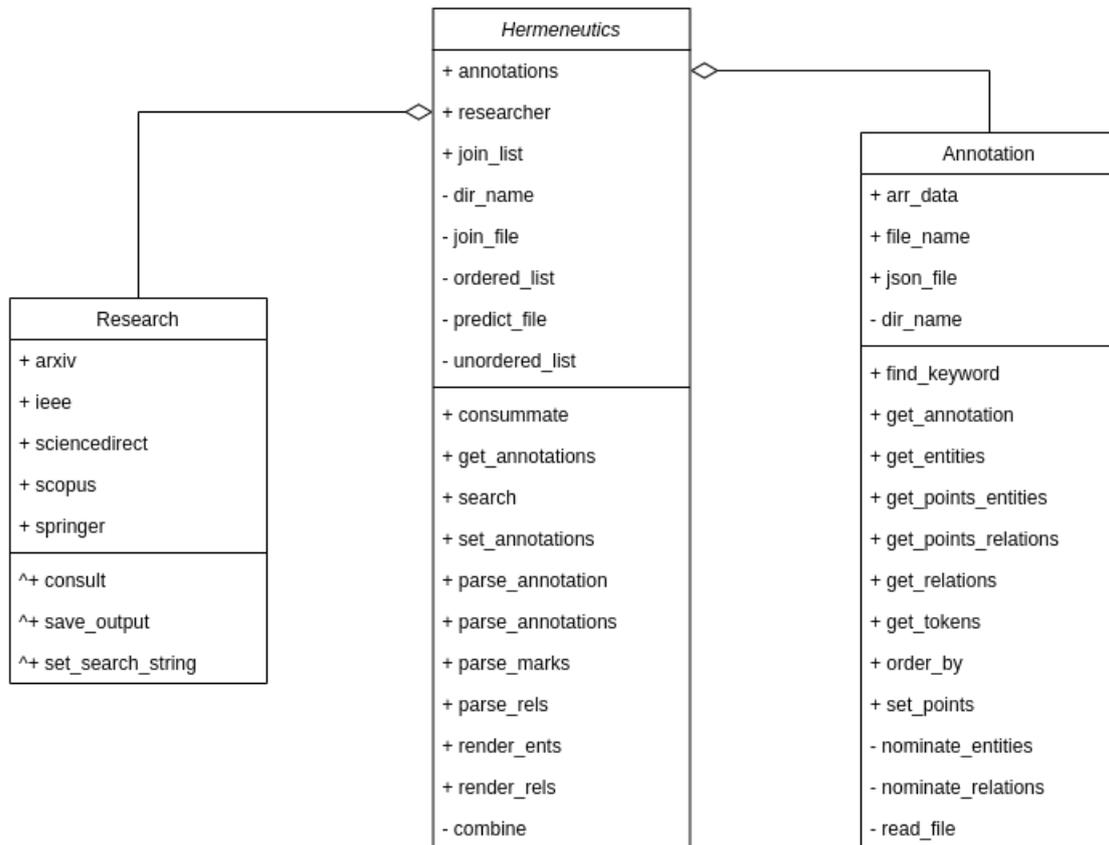
Fonte: Autor.

4.2.2.2 Interpretador

Para orquestrar a ferramenta como um todo, foi criada uma classe chamada *Hermentics*, que é responsável por efetuar as consultas nas bibliotecas digitais utilizando a classe *Research* e manipular os artigos e as anotações semânticas de entidades e relacionamentos obtidas pelo processamento do *framework* SpERT através da classe auxiliar

Annotation, onde também é aplicado o algoritmo de ranqueamento, detalhado na seção 4.2.3. Logo a seguir, é apresentado o diagrama de classes contendo as três principais classes da ferramenta.

Figura 4.5: Hermeneutics - Diagrama de Classes



Fonte: Autor.

Conforme mencionado anteriormente, a classe *Hermeneutics* é responsável pela gerência e execução do processo principal da ferramenta, descrito na Figura 4.2. Além de aplicar os algoritmos de ranqueamento e filtragem dos resultados nessa classe é feita a renderização dos artigos e suas anotações, estruturados na forma de dicionários JSON para o formato *tags* HTML, utilizando a biblioteca spaCy⁶, onde o resumo é transposto em texto e as anotações de entidades são representadas por *tags* `<mark>`, combinados os atributos *popover* para exibição dos relacionamentos.

4.2.2.3 Banco de Dados

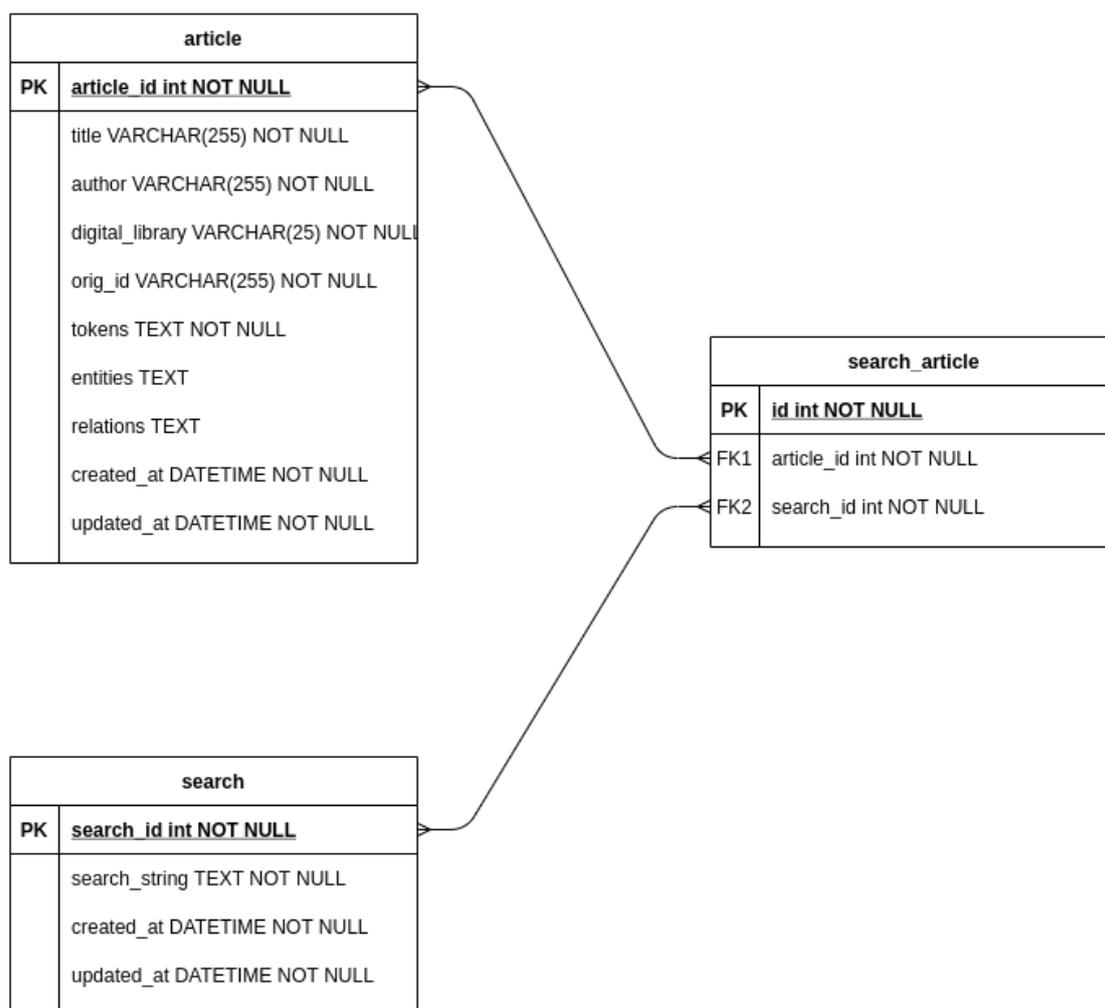
Para a persistência dos dados da aplicação, foi construído um modelo de entidade de relacionamento, representado pela Figura 4.6. Na listagem a seguir, são descritas

⁶spaCy <<https://spacy.io/>>

resumidamente as tabelas do modelo e seu papel dentro da ferramenta.

- **Article:** tabela que contém os registros dos artigos obtidos após consultas nas bibliotecas digitais.
- **Search:** tabela que contém as *strings* de buscas utilizadas pelo usuário.
- **search_article:** tabela $N : N$ contendo a relação das *string* de busca com os artigos obtidos

Figura 4.6: Banco de Dados - Modelo Entidade Relacionamento



Fonte: Autor.

O principal uso do banco de dados foi para minimizar reprocessamentos na forma de *cache* dos resultados, uma vez que, se uma *string* de busca já foi pesquisada na ferramenta, então os resultados obtidos das bibliotecas digitais são armazenados e atualizados com o processamento de NER/SLR feito pelo *framework* SpERT para obtenção das entidades e relacionamentos contidas no resumo dos artigos.

4.2.3 Algoritmo de ranqueamento

Como parte do algoritmo de classificação e ranqueamento da ferramenta, foi utilizada uma heurística de pontuação similar às utilizadas em árvores A^* . A ideia de implementação veio através de um estudo paralelo da graduação com a leitura de artigos de ITS (*Intelligent Transportation Systems*), onde informações sobre o trânsito eram retiradas de *microblogging* na internet para encontrar rotas que evitem congestionamentos e acidentes através da pontuação dos termos relacionados com a rota do usuário (Hasby; Khodra, 2013).

Com base nisso, foi construída a heurística de valores conforme as categorias de entidades e relacionamentos do modelo de predição do SciERC apresentada na tabela seguinte (4.1).

Tabela 4.1: Pontuação Heurística

Entidade	Pontos	Relacionamento	Pontos
Task	4	Used-for	5
Method	5	Feature-of	4
Evaluation Metric	3	Hyponym-of	4
Material	2	Part-of	3
Other Scientific Term	2	Compare	2
Generic	1	Conjunction	2

Fonte: Autor.

O algoritmo de ranqueamento consiste em, inicialmente, somar a quantidade de pontos obtidos com base na ocorrência das palavras chaves da *string* de busca que foram prevista como entidades e relacionamentos extraídos do resumo do artigo e, posteriormente, ordenar a listagem de artigos da maior pontuação para a menor. A seguir, é apresentado o algoritmo de ranqueamento utilizado para ordenação dos resultados em pseudocódigo.

Algoritmo 1: Algoritmo de ranqueamento

Entrada: String de Busca
Resultado: Lista de artigos ordenados
 inicio;
 busca artigos;
para todo *Artigo encontrado* com a String de busca **faça**
 para cada *Palavra Chave contida* na String de busca **faça**
 para cada *Entidade contida* no abstract do artigo **faça**
 se *Palavra Chave é uma entidade* **então**
 soma pontos conforme a *categoria da entidade*;
 fim
 fim
 para cada *Relacionamento contido* no abstract do artigo **faça**
 se *Palavra Chave faz parte de um relacionamento* **então**
 soma pontos conforme a *categoria do relacionamento*;
 fim
 fim
 fim
ordena artigos conforme a pontuação;
retorna lista de artigos;

4.2.4 Arquitetura e Tecnologias

Nesta seção será detalhado o conjunto de linguagens, bibliotecas e *frameworks* que foi utilizado para confeccionar o protótipo da ferramenta e também como está estruturada sua arquitetura.

4.2.4.1 Modelo Arquitetural

A ferramenta foi desenvolvida para web conforme os requisitos apresentados na seção 4.1.1, por se tratar de ferramenta web foi construída seguindo a arquitetura cliente-servidor, através do *framework* Django⁷ (versão 2.2), escrito na linguagem de programação Python (versão 3.6).

Um projeto Django possui como padrão de arquitetura o MTV (Model-Template-View), no qual cada um dos componentes da servem para:

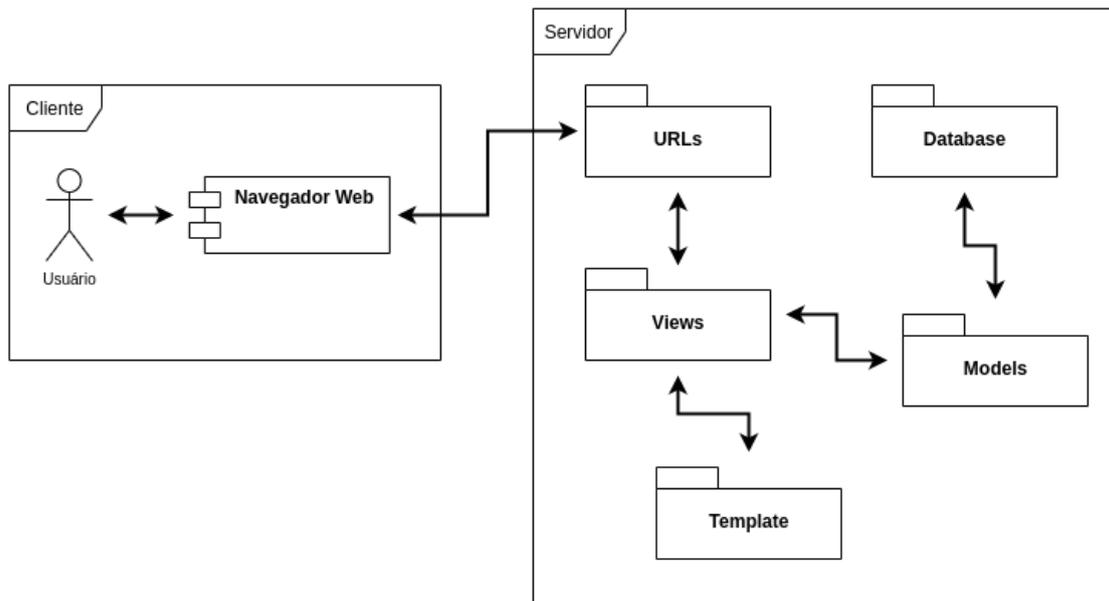
- **Model:** Mapeamento do banco de dados para o projeto.
- **Template:** Define a forma da apresentação dos dados ao usuário.

⁷Django Project <<https://www.djangoproject.com/>>.

- **View:** Recebe os comandos dos usuários e efetua as regras do negócio.

Na figura seguinte é exemplificado o modelo arquitetural do Django utilizado na ferramenta.

Figura 4.7: Arquitetura - Modelo MTV



Fonte: Autor.

Toda essa arquitetura se assemelha bastante com o padrão MVC (Model-View-Controller) (SOMMERVILLE, 2010), amplamente utilizado em diversos outros *frameworks*. Comparado ao MVC, as camadas do padrão MTV podem ser consideradas conforme listagem abaixo:

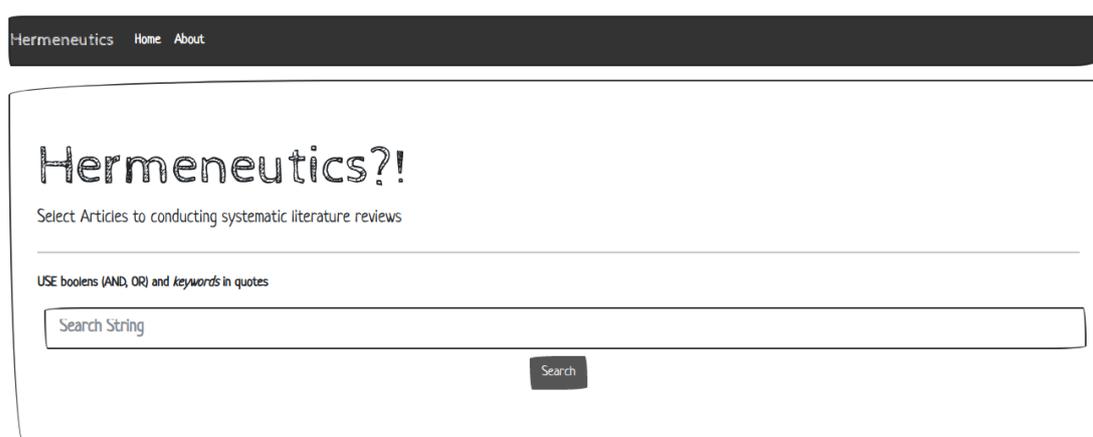
- **M (MTV) = M (MVC):** estas duas camadas possuem a mesma responsabilidade, mapeamento do banco de dados para o projeto;
- **T (MTV) = V (MVC):** estas camadas possuem a mesma responsabilidade, exibir informações para o usuário da aplicação, normalmente utilizando páginas HTML;
- **V (MTV) = C (MVC):** estas duas camadas, apesar de possuírem responsabilidades parecidas, conceitualmente, apresentam algumas diferenças⁸.

⁸Django FAQs #django-appears-to-be-a-mvc-framework-but-you-call-the-controller-the-view-and-the-view-the-template-how-come-you-don-t-use-the-standard-names. <<https://docs.djangoproject.com/en/2.2/faq/general/>>.

4.2.4.2 Front-end

Esta camada corresponde ao componente *Template* do modelo MTV, sendo responsável pela interface com o usuário. Em uma arquitetura Web, esta camada corresponde ao conteúdo recebido e executado no cliente, ou seja, código HTML⁹, CSS¹⁰ e JavaScript¹¹. Para a construção do *front-end* foi utilizado o *framework Bootstrap*¹², para funcionalidades relacionadas a interatividade/comportamento da interface com usuário foi utilizada a biblioteca *JQuery*¹³

Figura 4.8: Hermeneutics - Tela Inicial



Fonte: Autor.

A Figura 4.8 mostra a tela inicial da ferramenta. Através dela o pesquisador informa a *string* de busca que será utilizada na seleção de artigos da revisão sistemática.

4.2.4.3 Back-end

Esta camada corresponde aos componentes *Model* e *View* do padrão MTV, sendo responsável pelo acesso e manipulação dos dados e tratamento dos comandos recebidos do usuário.

Conforme já descrito anteriormente, a ferramenta foi desenvolvida utilizando Python através do *framework* Django, encarregado do processamento das requisições do usuário conforme o modelo de arquitetura. A ferramenta faz utilização do *framework* SpERT, que utiliza abundante recurso de hardware para obtenção das entidades e relacionamentos dos resumos de artigos obtidos pelas REST API das bibliotecas digitais citadas

⁹HTML <<https://developer.mozilla.org/pt-BR/docs/Web/HTML>>.

¹⁰CSS <<https://developer.mozilla.org/pt-BR/docs/Web/CSS>>.

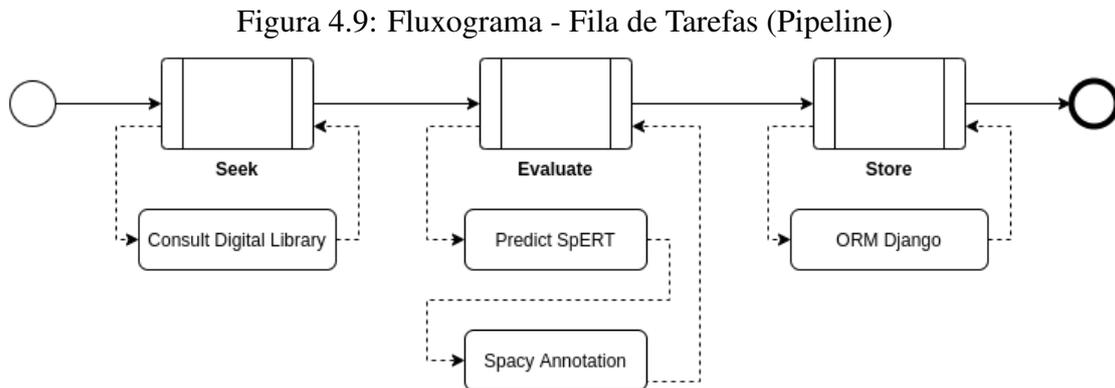
¹¹JS <<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>>.

¹²Bootstrap Framework 5.0.1 <<https://getbootstrap.com/>>.

¹³JQuery 3.5.1 <<https://jquery.com/>>.

anteriormente.

Visando o requisito não funcional de tempo de espera foi necessário implantar no *back-end* um subsistema de filas assíncronas para lidar com as requisições do usuários e consultar nas bibliotecas digitais de maneira encadeada conforme o fluxograma de tarefas abaixo:



Fonte: Autor.

4.2.4.3.1 Fila de Tarefas Assíncrona

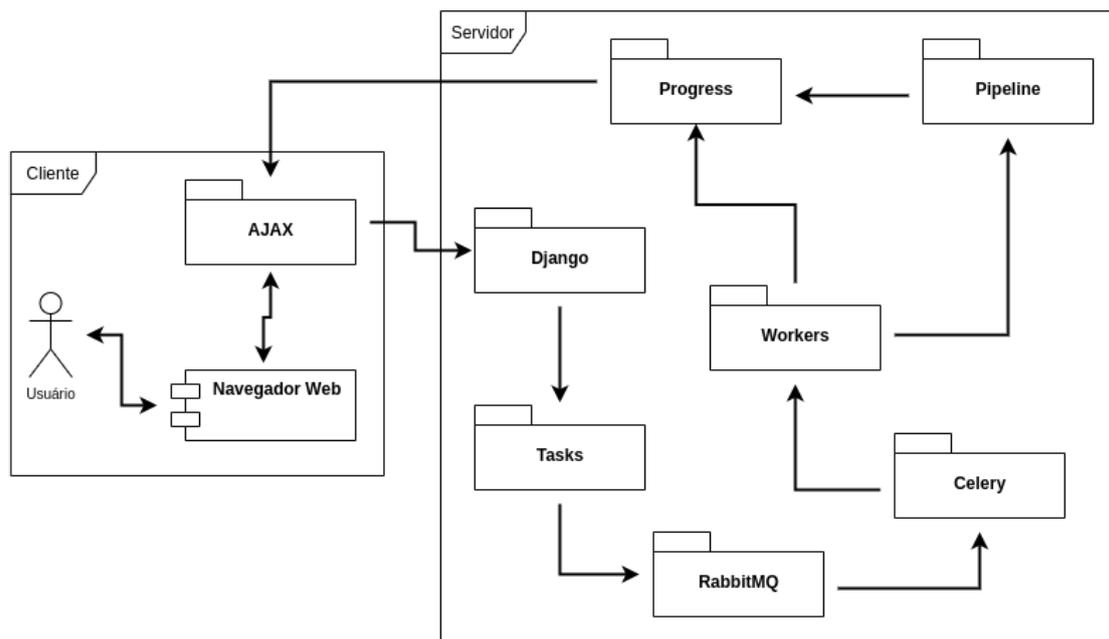
Para lidar com o assincronismo paralelo do fluxo de execução da ferramenta, foi utilizada a biblioteca Celery¹⁴ do Python que utiliza RabbitMQ¹⁵ através de extensões do *framework* Django para a distribuição e execução de tarefas paralelas/assíncronas.

A seguir é exemplificado o fluxo de execução da comunicação cliente/servidor para a utilização da fila de tarefas assíncrona.

¹⁴Celery 5.1.2 <<https://docs.celeryproject.org/en/stable/>>.

¹⁵RabbitMQ 3.6.10 <<https://www.rabbitmq.com/>>.

Figura 4.10: Diagrama - Assincronismo



Fonte: Autor.

Na Figura 4.10 é possível observar o seguinte passo a passo detalhado:

1. Usuário faz requisições via o Navegador Web;
2. O Navegador Web através de AJAX^{16,17} envia as requisições ao Servidor;
3. A requisição chega no Servidor Django e, seguindo modelo arquitetural, a URL da requisição é processada pelo *middleware* que encaminha para *View* responsável pela ação solicitada na requisição;
4. A *View*, por sua vez, chama a *task* responsável pela ação solicitada;
5. A *Task* através de mensagens do broker RabbitMQ agenda as tarefas no Celery;
6. O Celery, responsável por orquestrar as tarefas, cria dois *workers* para cada requisição do usuário:
 - 6.1. **Progress**: responsável por contabilizar o progresso de execução e responder para a solicitação AJAX periodicamente conforme o andamento das etapas do Pipeline;
 - 6.2. **Pipeline**: sequência de tarefas (*tasks*) que implementam o a execução do processo definido na Figura 4.9.

¹⁶AJAX definição <<https://developer.mozilla.org/pt-BR/docs/Web/Guide/AJAX>>.

¹⁷JQuery AJAX <<https://api.jquery.com/jquery.ajax/>>.

7. Conforme as respostas do Progress são recebidas pelo usuário, a barra de progresso da ferramenta é atualizada com o percentual de conclusão da obtenção de artigos para seleção. Quando finalizado o pipeline o usuário é redirecionado para a tela (Figura 4.11) contendo a listagem de todos os artigos obtidos.

Figura 4.11: Hermeneutics - Tela de Resultados

Hermeneutics Home About

Results - ("systematic literature review" OR "systematic review" OR "evidence-based") AND (clustering OR classification OR "information extraction" OR "information retrieval" OR "information visualization" OR "summarization" OR "text mining")

Have Entities: Method, Task, Metric, Material, OtherEntityTypes

Boolean: AND OR

With Relations: USED-FOR, HYPONYM-OF, FEATURE-OF, EVALUATE-FOR, PART-OF

Ordered

Title	Author	DOI	From	Points	Abstract
How to Improve Text Summarization and Classification by Mutual Cooperation on an Integrated Framework	Hyoungil Jeong	10.1016/j.eswa.2016.05.001	sciencedirect	175	+
Text-Mining Techniques and Tools for Systematic Literature Reviews: A Systematic Literature Review	Luyi Feng	10.1109/APSEC.2017.10	IEEE	162	+
Architecture and optimization of data mining modeling for visualization of knowledge extraction: Patient safety care	Gebeyehu Belay Gebremeskel	10.1016/j.jksuci.2018.12.001	sciencedirect	154	+
Clustering and visualization of a high-dimensional diabetes dataset	Piotr Lasek	10.1016/j.procs.2019.09.392	sciencedirect	144	+

Fonte: Autor.

5 AVALIAÇÃO DA FERRAMENTA

Este capítulo aborda a avaliação da ferramenta projetada e implementada conforme relatado no Capítulo 4. Foi utilizado como critério de avaliação o *System Usability Scale* (SUS), conforme definido por Brooke et al. (1996). A seção a seguir descreve o processo de concepção da avaliação, enquanto a seção posterior aborda a aplicação e a interpretação dos resultados da avaliação, por fim são apresentados alguns resultados do desempenho da ferramenta durante os experimentos.

5.1 Concepção

Para guiar os testes da ferramenta, foram disponibilizados o link para acessar o protótipo da ferramenta e o manual da ferramenta (Anexo A). Ao final do uso, o usuário é instruído a responder o questionário contendo o conjunto de 10 questões para a avaliação da usabilidade da ferramenta utilizando o SUS e o conjunto de perguntas sobre as funcionalidades da ferramenta.

Foram convidados alunos de doutorado do PGIE e do PPGC. Pediu-se que eles utilizassem a ferramenta para realizar a seleção de artigos, simulando a fase inicial de uma revisão sistemática da literatura.

5.1.1 Tópicos Pesquisados

Como atividades de avaliação da ferramenta, foram sugeridas as strings de buscas listadas na Tabela 5.1. Além disso, os participantes eram livres para utilizar qualquer *string* de busca a fim de explorar a ferramenta. A única restrição é que fossem termos de Computação.

Tabela 5.1: RSL - Strings de Buscas

Tópico/Tema	String de Busca	Fonte
<i>Text-mining Techniques and Tools for SLR</i>	("systematic literature review"OR "systematic review"OR "evidence-based") AND (clustering OR classification OR "information extraction"OR "information retrieval"OR "information visualization"OR "summarization"OR "text mining")	(Feng; Chiam; Lo, 2017)
<i>Ferramentas para auxiliar a execução de uma RSL</i>	("systematic-literature-review"OR "systematic-review"OR "mapping-study") AND ("tool"OR "web-application"OR "software-system") AND ("software-engineering") AND ("data-extraction"OR "conducting")	(CORDOVA, 2017)
<i>Jogos para Ensinar Computação</i>	("validation"OR "evaluation") AND ("jogo-educacional"OR "educational-game"OR "game-based-learning") AND ("computacao"OR "computing"OR "computer-science"OR "software")	(CORDOVA, 2017)

Fonte: Autor.

5.1.2 Perguntas

Essa seção descreve as questões de usabilidade da ferramenta utilizadas na avaliação, bem como o conjunto de perguntas específicas das funcionalidades da ferramenta utilizadas para obtenção do *feedback* do usuário nos experimentos de avaliação.

5.1.2.1 Questionário SUS:

- Q1 Eu acho que gostaria de usar esse sistema com frequência.
- Q2 Eu acho o sistema desnecessariamente complexo.
- Q3 Eu achei o sistema fácil de usar.
- Q4 Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema.
- Q5 Eu acho que as várias funções do sistema estão muito bem integradas.
- Q6 Eu acho que o sistema apresenta muita inconsistência.
- Q7 Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente.
- Q8 Eu achei o sistema atrapalhado de usar.

Q9 Eu me senti confiante ao usar o sistema.

Q10 Eu precisei aprender várias coisas novas antes de conseguir usar o sistema.

5.1.2.2 Conjunto de perguntas específicas:

1. A exibição de entidades nomeadas no texto do resumo dos artigos ajudou na seleção de artigos para realização de uma RSL.
2. A exibição dos relacionamentos das entidades nomeadas no texto do resumo dos artigos ajudou na seleção de artigos para realização de uma RSL.
3. A ordenação dos artigos apresentados ajudou na seleção de artigos para realização de uma RSL.

5.1.2.3 Grau da concordância

Para cada uma das perguntas, o usuário informou seu grau de concordância, dentre os cinco níveis disponíveis mapeados nos valores de 1 até 5:

- I Discordo totalmente;
- II Discordo parcialmente;
- III Neutro;
- IV Concordo parcialmente;
- V Concordo totalmente.

Com base nos valores das respostas, foi calculado o SUS Score detalhado na próxima seção.

5.2 Resultados

Foram enviados convites para a participação da avaliação, através de e-mail para alunos do grupo de pesquisa do professor orientador deste trabalho. No total, 5 indivíduos participaram do teste, atingindo o mínimo necessário para avaliação da usabilidade, segundo Nielsen (1994). Nas sessões seguintes, são detalhados o perfil dos participantes, as respostas obtidas do questionário SUS e o *feedback* das funcionalidades da ferramenta.

5.2.1 Perfil dos participantes

Os participantes foram 80% alunos de doutorado e 20% pesquisadores/professores, dos cursos de Computação e Informática na Educação, respectivamente, mais da metade deles havia realizado 6 ou mais RSL, enquanto os demais haviam realizado em torno de dois RLS. Além disso, informações sobre o ambiente de testes foram coletadas e 40% deles utilizavam sistema operacional Windows, 40% Mac e 20% Linux. Também para os testes foram utilizados variados navegadores web, conforme observado na Tabela 5.2 abaixo.

Tabela 5.2: Perfil dos participantes

Grau de Formação	%
Aluno de Graduação	0
Aluno de Especialização	0
Aluno de Mestrado	0
Aluno de Doutorado	80
Professor/Pesquisador	20

Curso	%
Computação	40
Informática na Educação	60

Quantas RSL realizou	%
Nenhuma	0
De 1 a 2	40
De 3 a 4	0
De 5 a 6	0
Mais de 6	60

Navegador Web	%
Chrome	40
Edge	20
Firefox	20
Opera	20
Safari	0

Fonte: Autor.

5.2.2 Usabilidade

Conforme mencionado anteriormente, foi utilizado o SUS como questionário para avaliação da usabilidade da ferramenta. Com base nas respostas dos usuários, foi calculado o SUS Score, que possui valores no intervalo de 0 a 100, onde a média são 68 pontos calculados conforme a definição de Brooke et al. (1996).

Utilizando a Tabela 5.3 para interpretação do SUS Score, na Figura 5.1 são apresentados as pontuações individuais de cada participante e a média aritmética de pontuação da ferramenta.

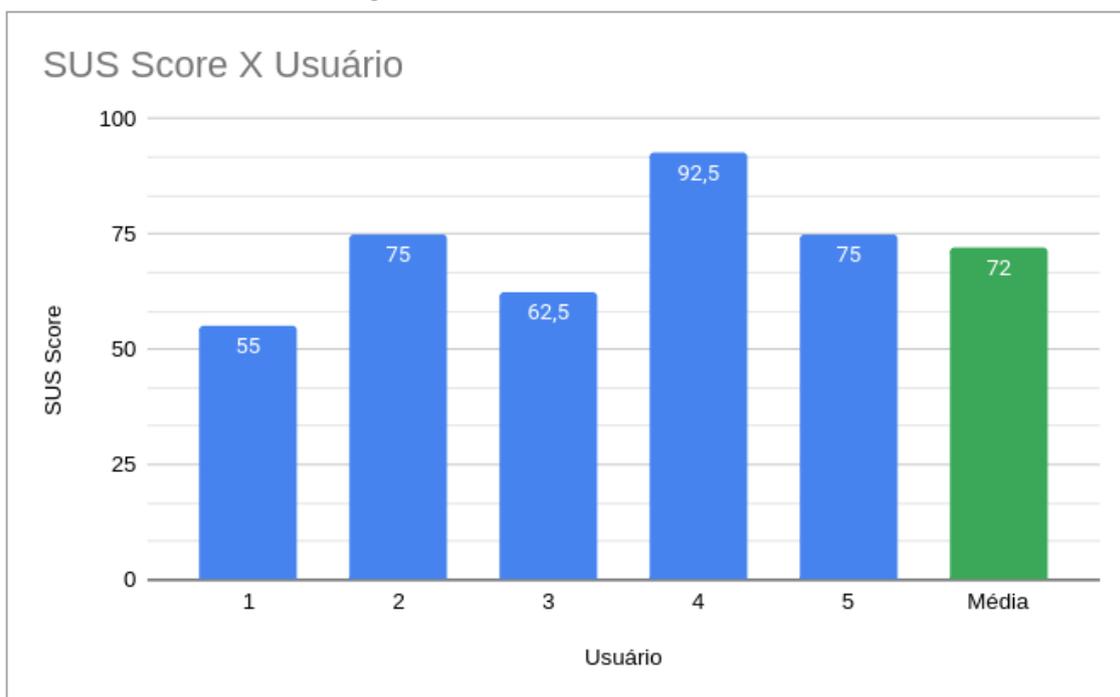
Conforme observado acima, a ferramenta obteve o valor 72, o que equivale ao conceito B, sendo ela classificada como tendo Boa Usabilidade.

Tabela 5.3: SUS Score - Interpretação

SUS Score	Conceito	Classificação
0-50	F	Horrível
51-62	D	Pobre
63-70	C	Ok
71-80	B	Boa
80-100	A	Excelente

Fonte: (BRAUM, 2019).

Figura 5.1: SUS Score - Resultados

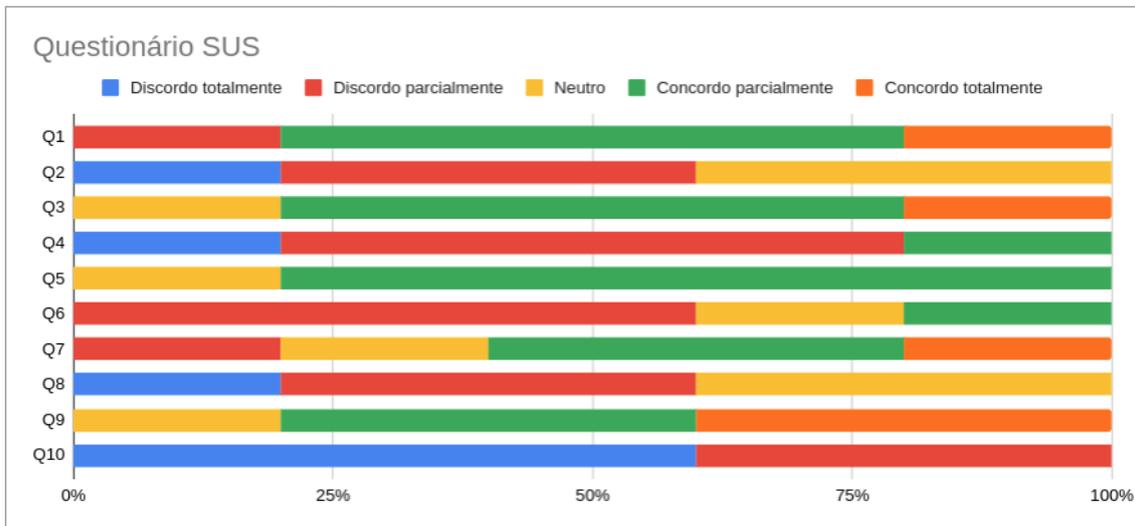


Fonte: Autor.

Na Figura 5.2 são apresentadas as respostas das 10 perguntas do questionário SUS (seção 5.1.2.1) na forma de barras empilhadas, visando mostrar o percentual acumulado de resposta para cada questão.

Para as questões ímpares (pontos positivos) não houveram respostas *Discordo totalmente* e para questões pares (pontos negativos), por sua vez, não houveram respostas *Concordo plenamente* que conforme indica o SUS Score obtido a ferramenta possui uma boa usabilidade.

Figura 5.2: SUS Questionário - Resultados

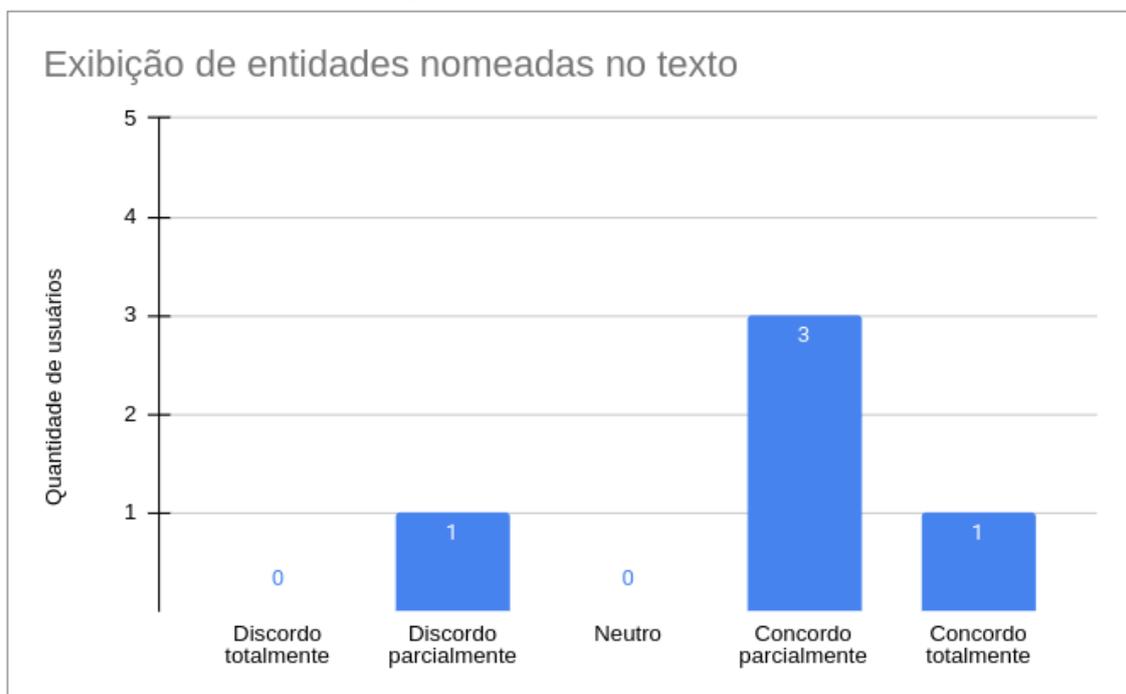


Fonte: Autor.

5.2.3 Feedback das funcionalidades

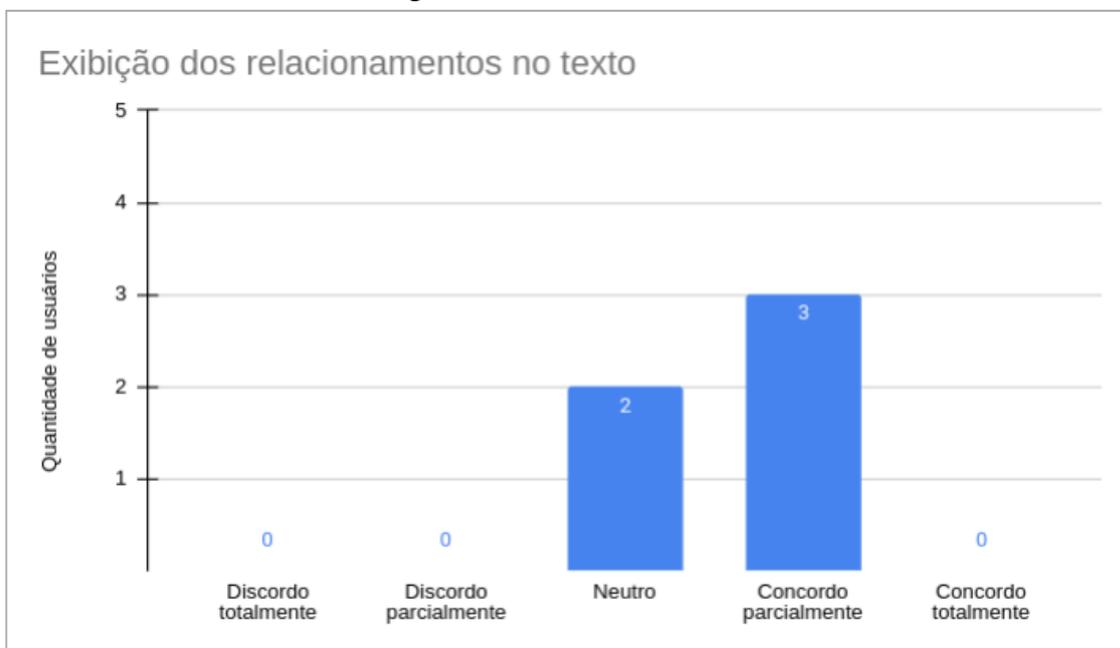
Nesta seção são apresentados os resultados do *feedback* sobre as funcionalidades obtidas através do conjunto de perguntas específicos da seção 5.1.2.2

Figura 5.3: NER - Resultados



Fonte: Autor.

Figura 5.4: SLR - Resultados



Fonte: Autor.

Figura 5.5: Ordenação - Resultados



Fonte: Autor.

Conforme pode ser observado nas figuras anteriores, a maioria dos usuários considerou relevante a exibição das entidades nomeadas nos textos dos resumos dos artigos, bem como os relacionamentos entre entidades, pois a possibilidade de ver itens destacados como os métodos e tarefas são utilizados, fazem parte ou são comparados acaba auxiliando na análise semântica que o pesquisador precisa fazer para selecionar os artigos

para condução de uma RSL.

Além disso, a ordenação da listagem que foi desenvolvida utilizando o algoritmo de ranqueamento demonstrou, através do *feedback* dos avaliadores, que o algoritmo simples de acúmulo de pontos por correspondência das entidades e relacionamentos com as palavras da *string* de busca o torna significativo, pois ajuda bastante na estruturação da busca e no atingimento dos resultados esperados.

Dentre as dificuldades reportadas, vale frisar a combinação dos filtros em tela que, ao selecionar um dos artigos, acabava redirecionando o pesquisador para listagem inicial resultando em uma experiência ruim para alguns avaliadores conforme observado nas respostas Q5, Q6 e Q8 da Figura 5.2.

Por fim, era esperado que, através da ferramenta, os pesquisadores tivessem funcionalidades que proporcionassem auxílio na realização de uma RSL. Através da combinação de mecanismo de busca unificando bibliotecas digitais com marcação de entidades científicas e seus relacionamentos nos resumos dos artigos ranqueados pela correspondência com a *string* de busca se mostrou útil e de boa usabilidade, facilitando assim a seleção de artigos e para condução de RSL e possibilidade de aprimoramentos e trabalhos futuros.

5.2.4 Desempenho nos experimentos

Nesta seção são apresentadas as medições de tempo obtidas durante a utilização da ferramenta pelos participantes. A ferramenta foi implantada em uma máquina virtual¹ (VM) no servidor cedido pelo grupo de pesquisa do orientador. Para os testes com os participantes o ambiente foi parametrizado para processar apenas os primeiros 200 resultados obtidos das consultas nas bibliotecas digitais, trazendo assim cerca de 40 por biblioteca digital.

Os dados apresentados na Tabela 5.4 foram obtidos através do log de execução das tarefas gerenciadas pelo Celery, de tal modo que cada nova string de busca pesquisada na ferramenta gera um registro de tarefa iniciada com o respectivo *timestamp* do início da execução e ao final quando o worker *progress* retorna para o cliente com o 100% do processamento é atualizada a informação de data e hora da conclusão da tarefa. Sendo assim na coluna *tempo de processamento* é apresentada a diferença de tempo em minutos da criação e finalização de cada string de busca pesquisa durante a semana dos experimentos.

¹Virtual Machine – <<https://www.vmware.com/topics/glossary/content/virtual-machine>>.

Tabela 5.4: Desempenho - Experimentos

search_string_id	tempo de processamento	quantidade de artigos
10	0:10:26	116
11	0:08:11	84
12	0:06:45	197
21	0:10:40	159
22	0:08:19	160
23	0:08:08	160
24	0:18:04	225
25	0:07:51	366
26	0:01:48	356
Média	<i>0:08:55</i>	<i>202,56</i>

Fonte: Autor.

A máquina virtual utilizada possuía a seguinte configuração:

- **Processador** Intel Xeon E5620 2.40GHz x4;
- **Memória RAM** 16 GB 32 bits;
- **Disco** 120 GB;
- **Sistema Operacional** Ubuntu 18.04.6 LTS.

Como a VM utilizada para os experimentos não possui recursos de GPU para o processamento do modelo deep learning do SpERT usando CUDA através do framework Pytorch foi necessário reduzir a quantidade de artigos processados para que não ultrapassasse mais de 10 min de execução. Tendo em vista a condução real de uma RSL o protótipo da ferramenta foi afetado durante os experimentos por causa dos recursos de hardware disponíveis, limitando o escopo de avaliação. No capítulo seguinte são detalhados esses itens de limitações da ferramenta e possíveis trabalhos futuros relacionados ao desempenho.

6 CONCLUSÃO

Este trabalho apresentou brevemente o processo de revisão sistemática da literatura, algumas técnicas de NLP empregadas para facilitar a seleção de artigos, em particular NER e SRL.

Em seguida, foram mostrados os trabalhos relacionados, listando algumas ferramentas existentes na atualidade e comparando as técnicas de mineração textual empregadas na seleção de artigos de cada uma delas. Também foram apresentados dois modelos de *deep learning* para extração de entidades nomeadas e seus relacionamentos. De fato esse processamento foi feito através dos *frameworks* SciIE e SpERT, utilizando as categorias de entidades e relacionamentos do *benchmark* SciERC, base do modelo preditivo da implementação. Também foram descritas ferramentas, como o Sobek, que inspiraram o processo de obtenção dos dados deste estudo.

Logo após foi detalhada a proposta de implementação deste trabalho, sendo objetivo da ferramenta criada auxiliar na seleção de artigos para condução de uma revisão sistemática da literatura, com foco na execução da busca e seleção de trabalhos relevantes. Foram utilizados filtros sobre as entidades e relacionamentos dos resumos dos artigos obtidos através de *strings* de busca e bibliotecas digitais da Computação.

De forma geral, os resultados alcançados neste trabalho se mostraram satisfatórios. Com a obtenção do SUS Score 72 e os *feedbacks* dos participantes, podemos concluir que a ferramenta tem potencial.

O destaque das entidades nomeadas no texto e seus relacionamentos agrega na semântica da análise feita pelos pesquisadores, facilitando a seleção dos artigos. O agrupamento das bibliotecas digitais em uma única ferramenta de busca e a ordenação dos artigos conforme o algoritmo de ranqueamento se mostraram significativos para seleção dos artigos.

Como limitações, temos o modelo treinado apenas para linguagem em inglês, com artigos na temática de inteligência artificial, bem como rotinas para atualização e inclusão de novos artigos na listagem. Uma vez pesquisada a *string* de busca, a ferramenta não traz mais artigos além das quantidades parametrizadas inicialmente. Também há o problema da similaridade das *strings* de busca utilizadas: qualquer diferença entre *strings* de busca utilizadas torna necessário efetuar todo o processo de busca. Além disso, observou-se algumas incompatibilidades do *framework* de predição com resumos muito longos ou os artigos provenientes da biblioteca Springer, que, na maioria das vezes, gera erros de

tokenização.

Por fim, o cerne da limitação da ferramenta foi a utilização de recursos limitados de hardware, com múltiplos usuários, visto que o modelo de predição requer uso intensivo de CPU e GPU, e a arquitetura cliente-servidor não pode ficar esperando indefinidamente por resposta, impossibilitando que dois ou mais usuários utilizem a ferramenta simultaneamente.

6.1 Trabalhos Futuros

Como trabalhos futuros, foram pensados alguns itens que não foram possíveis de incluir no protótipo inicial da ferramenta utilizada nos experimentos de avaliação de usabilidade, sendo eles outras abordagens teóricas, melhorias e refatorações na implementação, além de itens de ajustes identificados durante a avaliação da ferramenta e melhorias sugeridas pelos usuários. Nesta seção são listados esses itens e eles são comentados brevemente.

- **Modelo de Predição**

Criação de um modelo de predição de entidades e relacionamentos próprio para a ferramenta, tomando como base o *framework* SpERT, que possui melhor F1 para o *dataset* do SciERC. Foi pensada a elaboração de um modelo de predição implementado diretamente para rodar via spaCy, ao invés de Pytorch, por conta de que os modelos do spaCy terem uma performance melhor em ambiente de produção, o que melhoraria a experiência do usuário ao buscar por novos artigos. Nessa linha de raciocínio, foi pensado o treinamento do modelo utilizando conjuntos de artigos de revisões sistemáticas divididos em grupos de exemplos positivos e negativos, visando a especializar o modelo para melhor relevância dos artigos selecionados. Também, como descrito anteriormente nas limitações do modelo utilizado no protótipo, o mesmo foi treinado apenas com artigos de Inteligência Artificial em inglês e uma outra abordagem de melhoria do modelo seria estender para outros idiomas, principalmente para o português brasileiro. Outra ideia que surgiu durante o projeto de implementação da ferramenta foi criar *features* baseadas em regras de hermenêutica utilizando o *Attention Mechanism* das redes *transformers* BERT e analisar se haveria algum ganho significativo na corretude das predições do modelo.

- **Adição de Bibliotecas Digitais**

Conforme detalhado na seção 4.2.2.1, a ferramenta foi desenvolvida utilizando alguns conceitos do paradigma de orientação a objetos e, visando estender o uso da ferramenta, existe a possibilidade de incluir novas bibliotecas digitais. Para isso é necessário criar uma nova classe respectivamente para cada nova biblioteca digital, que herde da classe abstrata *Digital Library* e faça a sobreposição (*override*) dos métodos conforme a especificação de entradas e saídas esperadas, seguindo a documentação REST / SOAP da nova biblioteca digital para o *parsing* dos dados pela classe *Research* e no ambiente onde a ferramenta estiver rodando incluir no arquivo de *enviroment* o nome da nova classe para que o pipeline descrito na seção 4.2.4.3.1 passe a utilizar a nova biblioteca na etapa de consulta (*seek*). Dessa forma tornando possível pesquisar artigos de outras áreas do conhecimento ou novas fontes que estejam relacionados com o tema e tópico da RSL.

- **Algoritmo de ranqueamento**

Visando a uma melhoria na heurística implementada para o algoritmo de ranqueamento da ferramenta, foi pensando utilizar conjunto de artigos selecionados de RSL como *ground truth* para estipular os pesos das categorias de entidades e relacionamento na atribuição da pontuação do algoritmo.

- **Microserviços**

Visando a uma aplicação web mais robusta e escalável, foi pensado refatorar a implementação da ferramenta para a arquitetura de microserviços fazendo a separação das etapas do processo de busca de artigos, predição, armazenamento, *frontend* e *backend* em contêineres separados usando REST APIs para comunicação entre os serviços da ferramenta.

- **Múltiplas filas**

Nesta versão inicial da ferramenta, foi utilizada apenas uma única fila para escalonar as etapas de busca de artigos, predição e anotação das entidades e relacionamentos. Pensando em paralelismo da aplicação, a utilização de múltiplas filas para as tarefas assíncronas agrupadas por biblioteca digital traria um ganho de tempo de resposta da ferramenta.

- **Filtragem de artigos**

Nesta versão inicial da ferramenta, o armazenamento das entidades e relacionamento dos artigos obtidos foi feito utilizando banco de dados relacionais com o uso

de colunas específicas para as estruturas de dados das entidades e relacionamentos de cada artigo. Conforme detalhado na seção 4.2.2.3, foi estudado a possibilidade de utilizar banco de dados não relacionais NoSQL¹ para o armazenamento e indexação das entidades e relacionamentos obtidos dos artigos para utilizar na elaboração de *queries* complexas usando operadores booleanos sobre combinações dos dados de maneira similar a buscas em estruturas JSON.

- **Reordenação da listagem**

Visando uma melhor utilização da ferramenta, conforme observados nos comentários da avaliação, a inclusão da funcionalidade de reordenação da tabela de resultados, dando a possibilidade do usuário ordenar alfabeticamente as colunas e agrupar os resultados por biblioteca digital, bem como filtrar por ano de publicação e exibir esse dado na listagem.

- **Visualização de Entidades e Relacionamentos**

Visando uma melhor experiência do usuário, incluir opções no mecanismo de filtragem que permitam ao usuário escolher quais entidades estejam visíveis no texto do resumo, bem como incluir o acesso ao link para o artigo completo e a utilização de uma fonte que facilite a leitura dos resumos diferente da tipografia utilizada na identidade visual da ferramenta.

- **Tolerância a falhas**

Por se tratar de um protótipo, não foi investido em tratamentos adicionais da ferramenta como tratamento de *strings* de buscas fora da formatação correta e de indisponibilidade das bibliotecas digitais durante as buscas de artigos, além dos *bugs* reportados pelos usuários que realizaram o experimento de avaliação da ferramenta.

- **Melhorias sugeridas pelos avaliadores**

Alguns itens sugeridos nos comentários do experimento:

- Apresentar o total de artigos encontrados;
- Incluir a quantidade de citações dos artigos listados;
- Dar destaque (cor) na listagem para os artigos selecionados.

O código-fonte da ferramenta *Hemeneutics* foi hospedado em um repositório no GitHub (<<https://github.com/dlazarosps/hermeneutics>>). Espera-se que sua liberação na

¹AWS NoSQL <<https://aws.amazon.com/pt/nosql/>>.

forma *open-source* permita formar uma comunidade de desenvolvedores que siga aprimorando a ferramenta, ou que os trabalhos futuros listados aqui sirvam para continuidade do projeto e disponibilização da ferramenta para uso em grupos de pesquisa no instituto de informática.

REFERÊNCIAS

AZEREDO, P. **Metodos De Classificação De Dados: E ANALISE DE SUAS COMPLEXIDADES**. ELSEVIER EDITORA, 1996. ISBN 9788535200041. Available from Internet: <<https://books.google.com.br/books?id=oc1RzAEACAAJ>>.

BIOLCHINI, P. et al. A systematic review process to software engineering. v. 32, 01 2005.

BRAUM, M. **Guia: Como medir a usabilidade de produtos com System Usability Scale (SUS)**. UX Collective, 2019. Available from Internet: <<https://brasil.uxdesign.cc/guia-como-medir-a-usabilidade-de-produtos-com-system-usability-scale-sus-e08f4361d9db>>.

BRERETON, P. et al. Lessons from applying the systematic literature review process within the software engineering domain. **Journal of Systems and Software**, v. 80, n. 4, p. 571 – 583, 2007. ISSN 0164-1212. Software Performance. Available from Internet: <<http://www.sciencedirect.com/science/article/pii/S016412120600197X>>.

BROOKE, J. et al. Sus-a quick and dirty usability scale. **Usability evaluation in industry**, London-, v. 189, n. 194, p. 4–7, 1996.

CARRERAS, X.; MÀRQUEZ, L. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In: **Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)**. Ann Arbor, Michigan: Association for Computational Linguistics, 2005. p. 152–164. Available from Internet: <<https://www.aclweb.org/anthology/W05-0620>>.

CORDOVA, E. N. **Evolução da Ferramenta RESuLT para Auxiliar a Execução de Revisões Sistemáticas da Literatura**. Monografia (TCCgrad) — UNIVERSIDADE FEDERAL DE SANTA CATARINA, 06 2017. Available from Internet: <<https://repositorio.ufsc.br/xmlui/handle/123456789/177713>>.

EBERTS, M.; ULGES, A. **Span-based Joint Entity and Relation Extraction with Transformer Pre-training**. 2019.

Feng, L.; Chiam, Y. K.; Lo, S. K. Text-mining techniques and tools for systematic literature reviews: A systematic literature review. In: **2017 24th Asia-Pacific Software Engineering Conference (APSEC)**. [S.l.: s.n.], 2017. p. 41–50.

GILDEA, D.; JURAFSKY, D. Automatic labeling of semantic roles. In: **Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics**. Hong Kong: Association for Computational Linguistics, 2000. p. 512–520. Available from Internet: <<https://www.aclweb.org/anthology/P00-1065>>.

GLASS, G. V. Primary, secondary, and meta-analysis of research. **Educational Researcher**, [American Educational Research Association, Sage Publications, Inc.], v. 5, n. 10, p. 3–8, 1976. ISSN 0013189X, 1935102X. Available from Internet: <<http://www.jstor.org/stable/1174772>>.

GRISHMAN, R.; SUNDHEIM, B. Message understanding conference-6: A brief history. In: **Proceedings of the 16th Conference on Computational Linguistics - Volume 1**. USA: Association for Computational Linguistics, 1996. (COLING 96), p. 466471. Available from Internet: <<https://doi.org/10.3115/992628.992709>>.

Hasby, M.; Khodra, M. L. Optimal path finding based on traffic information extraction from twitter. In: **International Conference on ICT for Smart Society**. [S.l.: s.n.], 2013. p. 1–5.

JACOBS, L. S. **Utilizando buscas online para identificar informações relevantes em mineração de textos**. Monografia (TCCgrad) — UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, 06 2016. Available from Internet: <<https://lume.ufrgs.br/handle/10183/1510133>>.

JOHNSON, M. How the statistical revolution changes (computational) linguistics. In: **Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?** Athens, Greece: Association for Computational Linguistics, 2009. p. 3–11. Available from Internet: <<https://www.aclweb.org/anthology/W09-0103>>.

KITCHENHAM, B. Procedures for performing systematic reviews. **Keele, UK, Keele Univ.**, v. 33, 08 2004.

KITCHENHAM, K.; CHARTERS, S. Guidelines for performing systematic literature reviews in software engineering. v. 2, 01 2007.

KNUTH, D.; ADDISON-WESLEY. **The Art of Computer Programming: Fundamental algorithms**. Addison-Wesley, 1997. (Addison-Wesley series in computer science and information processing, v. 1). ISBN 9780201896831. Available from Internet: <<https://books.google.com.br/books?id=B31GAAAAYAAJ>>.

LORENZATTI, M. K. e Eliseo Reategui e A. O emprego da ferramenta de mineração de textos sobek como apoio à produção textual. **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)**, v. 1, n. 1, 2009. ISSN 2316-6533. Available from Internet: <<https://www.br-ie.org/pub/index.php/sbie/article/view/1154>>.

LOVINS, J. B. Development of a stemming algorithm. **Mech. Transl. Comput. Linguistics**, v. 11, p. 22–31, 1968.

LUAN, Y. et al. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In: **Proc. Conf. Empirical Methods Natural Language Process. (EMNLP)**. [S.l.: s.n.], 2018.

MACEDO, A. L. et al. Using text-mining to support the evaluation of texts produced collaboratively. In: TATNALL, A.; JONES, A. (Ed.). **Education and Technology for a Better World**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 368–377.

MOLLÉRI, J. S. **Automatização do processo de condução de revisões sistemáticas da literatura em engenharia de software**. Dissertation (Master) — Universidade do Vale do Itajaí, 08 2013. Available from Internet: <<https://siaiap39.univali.br/repositorio/handle/repositorio/1018>>.

NADEAU, D.; SEKINE, S. A survey of named entity recognition and classification. **Lingvisticae Investigationes**, v. 30, 01 2007.

NIELSEN, J. **Usability engineering**. [S.l.]: Morgan Kaufmann, 1994.

PAREDES, V. E. A. Métodos de interpretación jurídica. In: . [S.l.: s.n.], 2012.

REATEGUI, E. et al. Mineração textual e letramento: aplicações iniciais da ferramenta sobek com alunos do ensino fundamental. In: . [S.l.: s.n.], 2016. p. 111.

REESE, W. L. **Dictionary of Philosophy and Religion: Eastern and Western Thought**. [S.l.]: Humanity Books, 1996.

SACKS, H. et al. Meta-analysis: an update. **The Mount Sinai journal of medicine, New York**, v. 63, n. 3-4, p. 216224, 1996. ISSN 0027-2507. Available from Internet: <<http://europepmc.org/abstract/MED/8692168>>.

SCHENKER, A. et al. **Graph-Theoretic Techniques for Web Content Mining**. USA: World Scientific Publishing Co., Inc., 2005. ISBN 9789812563392.

SEBASTIANI, F. Machine learning in automated text categorization. **ACM Computing Surveys**, v. 34, p. 1–47, 04 2001.

SOMMERVILLE, I. **Software Engineering**. 9th. ed. USA: Addison-Wesley Publishing Company, 2010. ISBN 0137035152.

VASWANI, A. et al. **Attention Is All You Need**. 2017.

WINOGRAD, T. **Procedures as a Representation for Data in a Computer Program for Understanding Natural Language**. M.I.T. Project MAC, 1971. (AI-TR). Available from Internet: <<https://books.google.com.br/books?id=afh4oAEACAAJ>>.

ZIMMERMANN, J. **Hermeneutics: A Very Short Introduction**. [S.l.]: Oxford University Press UK, 2015.

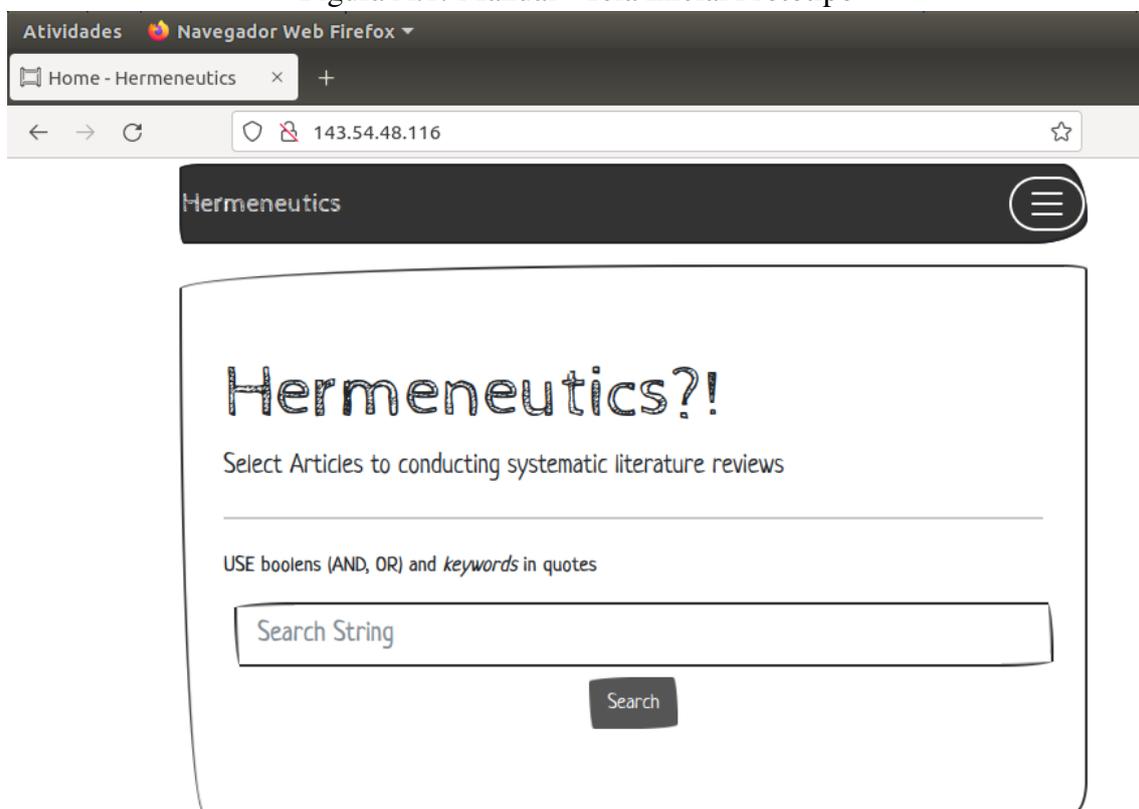
APÊNDICE A — MANUAL DA FERRAMENTA

Este manual explica resumidamente as funções principais do sistema.

I Acesso

- 1.1. Abrir navegador web (chrome / firefox)
- 1.2. Acessar o link <<http://hermeneutics.inf.ufrgs.br>>

Figura A.1: Manual - Tela Inicial Protótipo



Fonte: Autor.

II Pesquisar Artigos

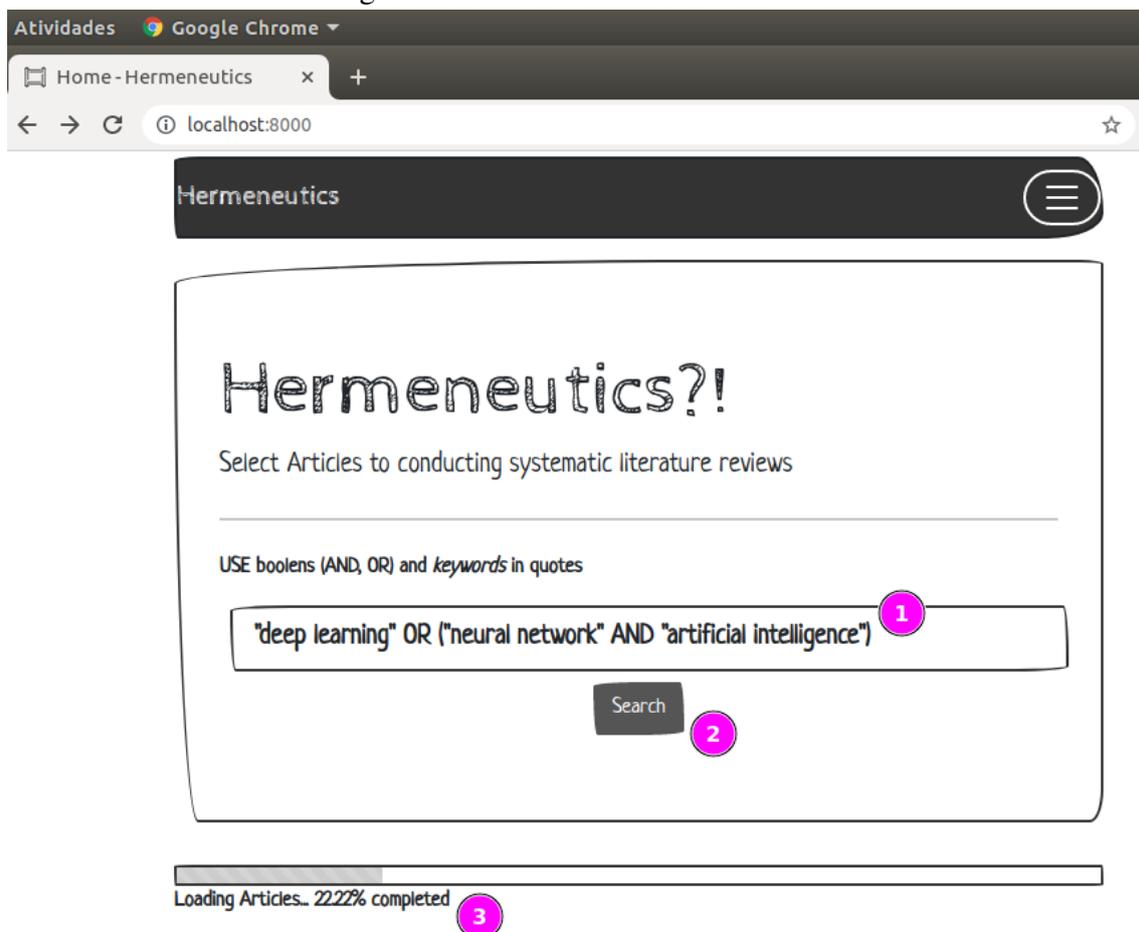
2.1. Informar a String de Busca

- *Palavras Chaves* - entre aspas (Ex: "Deep Learning")
- *Operadores Booleanos* - AND/OR (Ex: "Artificial intelligence"AND "Deep Learning")
- *Precedência* - Usar parênteses para modificar a precedência ou agrupar termos da string de busca

2.2. Clicar no botão *Search*

2.3. Aguardar o carregamento dos artigos conforme a barra de progresso

Figura A.2: Manual - Iniciando Busca



Fonte: Autor.

III Visualizar Resumos

Clicar no ícone + de cada artigo listado

Figura A.3: Manual - Visualizar Resumo

Hermeneutics Home About

Results - "deep learning" OR ("neural network" AND "artificial intelligence")

Have Entities: Method, Task, Metric, Material

Boolean: AND, OR

With Relations: USED-FOR, HYPONYM-OF, FEATURE-OF, EVALUATE-FOR

Title	Author	DOI	From	Points	Abstract
Forecasting wind waves in the US Atlantic Coast using an artificial neural network model. Towards an AI-based storm forecast system	Zhangping Wei	10.1016/j.oceaneng.2021.109646	sciencedirect	275	+
Evaluation tool for the thermal performance of retrofitted buildings using an integrated approach of deep learning artificial neural networks and infrared thermography	Amin Al-Hababeh	10.1016/j.enbenv.2020.06.004	sciencedirect	262	-

Evaluation tool for the thermal performance of retrofitted buildings using an integrated approach of deep learning artificial neural networks and infrared thermography

In most countries, buildings are responsible for significant energy consumption TASK where space heating TASK and air conditioning TASK is responsible for the majority of this energy use. To reduce this massive consumption METRIC and decrease carbon emission, thermal insulation of buildings OTHERSCIENTIFICTERM can play an important role. The estimation of energy savings TASK energy savings TASK following the improvement of a building's insulation remains a key area of research in order to calculate the cost savings METRIC and the payback period. In this paper, a case study has been presented where deep retrofitting METHOD has been

Fonte: Autor.

IV Visualizar Relacionamento entre entidades do resumo

Passar o mouse por cima das entidades marcadas no texto

Figura A.4: Manual - Visualizar Relacionamento entre Entidades

How to Improve Text Summarization and Classification by Mutual Cooperation on an Integrated Framework

Text summarization TASK and classification TASK are core techniques to analyze a huge amount of text data MATERIAL in the big data environment. Moreover, as the need to read texts on smart phones, tablets MATERIAL and television MATERIAL as well as personal computers MATERIAL continues to grow, text summarization and classification techniques METHOD become more important and both of them GENERIC do essential processes for text analysis TASK in many applications. Traditional text summarization and classification techniques METHOD have individually been considered as different research fields in this literature. However, we find out that they METRIC can help each other as text summarization TASK makes use of category information OTHERSCIENTIFICTERM digital_library text classification USED-FOR text classification TASK does summary information OTHERSCIENTIFICTERM digital_library text summarization. Therefore, we propose an effective integrated learning framework METHOD using both of summary and category information OTHERSCIENTIFICTERM in this paper. In this framework, the feature-weighting method METHOD for text summarization TASK utilizes a language model METHOD to combine feature distributions OTHERSCIENTIFICTERM in each category and text, and one for text classification TASK does the sentence importance scores METRIC estimated digital_library the text summarization. In the experiments, the performances of the Integrated framework OTHERSCIENTIFICTERM are better than ones of individual text summarization and classification. In addition, the framework GENERIC has some advantages of easy implementation and language independence OTHERSCIENTIFICTERM because it GENERIC is based on only simple statistical approaches METHOD and POS tagger.

Fonte: Autor.

V Selecionar Artigo

Clicar no botão *Select this* logo abaixo do resumo do artigo

Figura A.5: Manual - Selecionar Artigo

The screenshot shows a Google Chrome browser window with the URL `localhost:8000/list/`. The page title is "Results - Hermeneutics". The search query is "Results - 'deep learning' OR ('neural network' AND 'artificial intelligence')".

Filters are set to "Have Entities" (Method, Task, Metric, Material, Peer-Reviewed), "Boolean" (AND), and "With Relations" (EVALUATE-FOR, PART-OF, COMPARE, CONJUNCTION). The results are ordered.

Title	Author	DOI	From	Points	Abstract
Forecasting wind waves in the US Atlantic Coast using an artificial neural network model: Towards an AI-based storm forecast system	Zhangping Wei	10.1016/j.oceaneng.2021.109646	sciencedirect	275	
<p>Forecasting wind waves in the US Atlantic Coast using an artificial neural network model: Towards an AI-based storm forecast system</p> <p>This study presents an artificial intelligence (AI) model METHOD to forecast time-series wind waves OTHERSCIENTIFICTERM in the US Atlantic Coast. The fundamental technique GENERIC in the proposed model GENERIC is the Long Short-term Memory model, which is able to learn patterns OTHERSCIENTIFICTERM from data sequence. The model GENERIC is trained using historical wind data, wave data, temperature data, and atmospheric pressure data. Then the model GENERIC is used to forecast significant wave height, average wave period, and mean wave direction. A 2-year meteorological data MATERIAL measured at four NOAA buoy stations MATERIAL in the US Atlantic Coast MATERIAL are used to train and evaluate the model's forecast skills. The results show the artificial neural network model METHOD converges fast and does not have over-fitting nor under-fitting issues. Furthermore, a short-term forecast METHOD (e.g. 1 to 6 h) achieves higher accuracy METRIC than a long-term forecast METHOD (e.g. 24 to 48 h). This study shows that sufficient forecast accuracy METRIC can be obtained by using the input hours equal to the forecast lead time. Comparison using rose plots OTHERSCIENTIFICTERM confirms that the artificial neural network model METHOD reproduces wave height and wave period statistics OTHERSCIENTIFICTERM very well. The artificial neural network model METHOD is used to forecast storm waves OTHERSCIENTIFICTERM induced by Hurricanes Isalaz and Eta in the US South Atlantic region, and two winter storms OTHERSCIENTIFICTERM in the US North Atlantic region. The model forecast METHOD is compared with NOAA measurement. The results show that the artificial neural network model METHOD is able to accurately forecast both significant wave height OTHERSCIENTIFICTERM and average wave period OTHERSCIENTIFICTERM associated with storm events OTHERSCIENTIFICTERM when the forecast lead time OTHERSCIENTIFICTERM is relatively short (e.g. 1 to 6 h). The proposed artificial neural network model METHOD can serve as an alternative tool GENERIC to traditional coastal models METHOD for wave prediction TASK and storm forecast.</p> <p>View Bibtext Select This </p>					
Evaluation tool for the thermal performance of retrofitted buildings using an integrated	Amin Al-Hababeh	10.1016/j.enbenv.2020.06.004	sciencedirect	262	

Fonte: Autor.

Após executada a ação o artigo selecionado fica disponível na segunda aba da lista-chamada *Selected*

VI Visualizar Citação

Clicar no botão *View Bibtex* logo abaixo do resumo do artigo

Figura A.6: Manual - Visualizar Bibtex

The screenshot shows a Google Chrome browser window with the URL `localhost:8000/filter/`. The page displays search results for the query "deep learning". The first result is for the article "End-to-end predictive intelligence diagnosis in brain tumor using lightweight neural network" by Ma, Linjuan and Zhang, Fuquan, published in Applied Soft Computing in 2021. A modal window titled "Bibtex" is open, showing the following citation:

```
@article{2021,  
title={End-to-end predictive intelligence diagnosis in brain tumor using  
lightweight neural network},  
volume={111},  
ISSN={1568-4946},  
uri={http://dx.doi.org/10.1016/j.asoc.2021.107666},  
DOI={10.1016/j.asoc.2021.107666},  
journal={Applied Soft Computing},  
publisher={Elsevier BV},  
author={Ma,  
Linjuan and Zhang,  
Fuquan},  
year={2021},  
month={Nov},  
pages={107666}}
```

Below the article title, there is a summary of the paper. At the bottom of this summary, there are two buttons: "View Bibtex" and "Unselect This". A pink arrow points to the "View Bibtex" button.

Fonte: Autor.

VII Remover artigo selecionado

Para remover um artigo é necessário:

- 7.1. Ir na aba selected
- 7.2. Na listagem de artigos selecionados na visualização do resumo
- 7.3. Clicar no botão *Unselect this*

Figura A.7: Manual - Remover Artigo

The screenshot shows a web browser window with the URL `localhost:8000/filter/`. The page title is "Results - Hermeneutics". The search criteria are "deep learning" OR ("neural network" AND "artificial intelligence").

The interface includes the following elements:

- Have Entities:** A dropdown menu with options: Method, Task, Metric, Material.
- Boolean:** Radio buttons for AND and OR, with a "Filter" button below.
- With Relations:** A dropdown menu with options: USED-FOR, HYPONYM-OF, FEATURE-OF, EVALUATE-FOR, DATE-OF.
- Ordered/Selected:** Two tabs, with "Selected" being the active tab (circled in pink with a '1').
- Table:** A table with columns: Title, Author, DOI, From, Points, Abstract. The first row is selected (circled in pink with a '2').
- Abstract:** The abstract of the selected article is displayed below the table.
- Buttons:** "View Bibtex" and "Unselect This" (circled in red with a '3') are located below the abstract.
- Footer:** A table with columns: Title, Author, DOI, From, Points, and an "Export Selected Articles" button.

Fonte: Autor.

VIII Filtrar Artigos

Para filtragem de resultados é feita através de:

- 8.1. Selecionar uma ou mais categorias de entidades
- 8.2. Escolher o operador booleano (AND / OR)
- 8.3. Selecionar uma ou mais categorias de relacionamentos
- 8.4. Clicar no botão *Filter*

Figura A.8: Manual - Filtrar Artigos

The screenshot shows a web browser window with the URL `localhost:8000/filter/`. The page title is "Results - Hermeneutics". The main heading is "Results - 'deep learning' OR ('neural network' AND 'artificial intelligence')".

The filter configuration section includes:

- Have Entities:** A dropdown menu with options: Method (1), Task, Metric, Material.
- Boolean:** Radio buttons for AND (2) and OR.
- With Relations:** A dropdown menu with options: USED-FOR, HYPONYM-OF (3), FEATURE-OF, EVALUATE-FOR.
- Filter:** A button (4).

Below the filter configuration, there are two tabs: "Ordered" and "Selected". The "Selected" tab is active, showing a table of results:

Title	Author	DOI	From	Points	Abstract
Forecasting wind waves in the US Atlantic Coast using an artificial neural network model: Towards an AI-based storm forecast system	Zhangping Wei	10.1016/j.oceaneng.2021.109646	sciencedirect	275	+
Evaluation tool for the thermal performance of retrofitted buildings using an integrated approach of deep learning artificial neural networks and infrared thermography	Amin Al-Habaibeh	10.1016/j.enbenv.2020.06.004	sciencedirect	262	+

Fonte: Autor.

IX Baixar Citações

Para Baixar as citações dos artigos artigo é necessário:

9.1. Ir na aba “selected”

9.2. Ao final da listagem de artigos selecionados

Clicar no botão *Export Selected Articles*

Figura A.9: Manual - Baixar Citações

Hermeneutics Home About

Results - "deep learning" OR ("neural network" AND "artificial intelligence")

Have Entities

Method
Task
Metric
Material

Boolean

AND
 OR

Filter

With Relations

USED-FOR
HYPONYM-OF
FEATURE-OF
EVALUATE-FOR

Ordered Selected **1**

Title	Author	DOI	From	Points	Abstract
End-to-end predictive intelligence diagnosis in brain tumor using lightweight neural network	Linjuan Ma	10.1016/j.jasoc.2021.107666	sciencedirect	141	-
Forecasting wind waves in the US Atlantic Coast using an artificial neural network model: Towards an AI-based storm forecast system	Zhangping Wei	10.1016/j.oceaneng.2021.109646	sciencedirect	275	-

Export Selected Articles **2**

Fonte: Autor.

Os bibtext dos artigos selecionados serão baixados em um único arquivo chamado *selected_articles.bib*

X Paginação de Resultados

São exibidos por página 25 artigos por vez, através dos botões de paginação é possível visualizar demais artigos:

10.1. Primeira Página

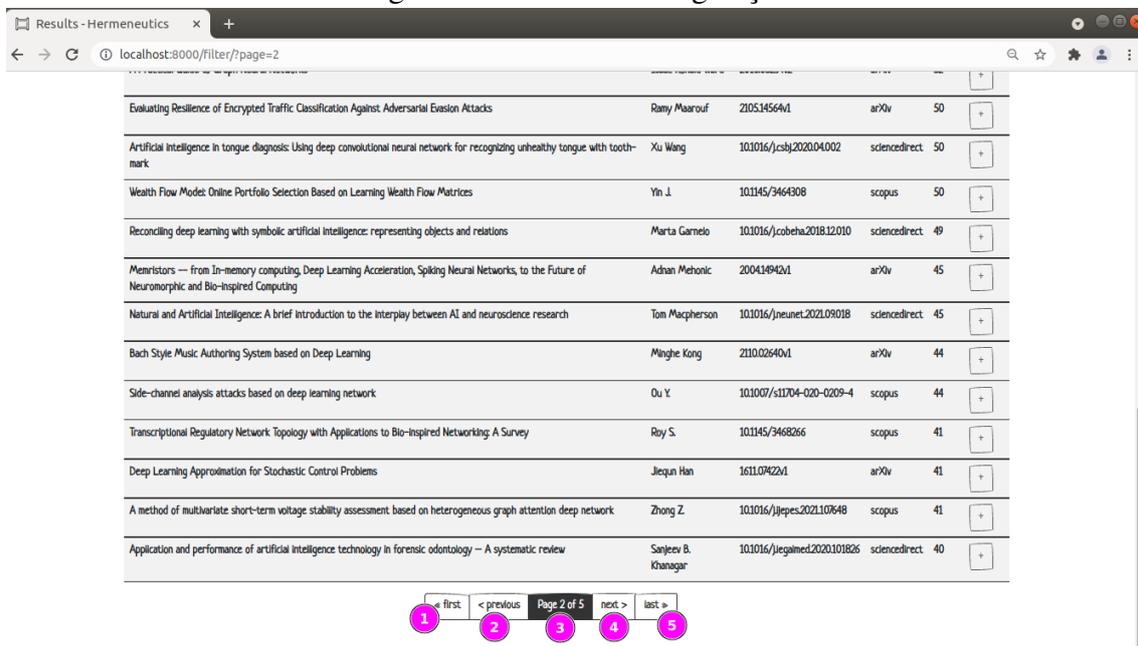
10.2. Página Anterior

10.3. Páginas de X até N

10.4. Próxima Página

10.5. Última Página

Figura A.10: Manual - Paginação



Fonte: Autor.