

# Revisão Sistemática Aliada à Mineração de Textos: Redução do Esforço Humano na Busca por Trabalhos Acadêmicos na Web

Hugo H. L. Diniz<sup>1</sup>, Leonardo O. Moreira<sup>1</sup>

<sup>1</sup> Instituto Universidade Virtual (UFC Virtual)  
Universidade Federal do Ceará (UFC) – Fortaleza, CE – Brasil

hhldiniz@alu.ufc.br, leoomoreira@virtual.ufc.br

**Abstract.** *Scientific research in the electronic medium is considered slow and laborious due to the large amount of search sources and unfriendly interfaces. This work explains the development process of a system to perform research in several sources in an integrated and facilitated way, reducing the researcher's effort in the task. An assessment of the application with users was made that shows their opinion and experience during the use of the system prototype.*

**Resumo.** *Pesquisas científicas no meio eletrônico são consideradas lentas e trabalhosas devido a grande quantidade de fontes de buscas e da interface pouco amigável de muitas delas. Este trabalho explica o processo de desenvolvimento de um sistema para realização de pesquisa em diversas fontes de forma integrada e facilitada reduzindo assim o esforço do pesquisador na tarefa. Foi feita uma avaliação da aplicação com os usuários que mostra a opinião e experiência deles durante o uso do protótipo do sistema.*

## 1. Introdução

A revisão sistemática surge como um guia e viabilizador na exploração por um determinado tema de pesquisa de forma clara e explícita [Sampaio and Mancini 2007]. Para isso, a revisão sistemática possui um conjunto de passos ou etapas que são definidas para guiar esse processo de exploração em um determinado tema de pesquisa [Coutinho et al. 2015]. A busca por trabalhos ou evidências em plataformas especializadas de pesquisa é uma das etapas envolvidas na revisão sistemática que tem como objetivo reunir trabalhos importantes que possam ter algum impacto na pesquisa em questão.

Qualquer pesquisador já precisou obter um trabalho acadêmico em um formato mais formal (e.g., artigos, capítulos, livros, monografias, dissertações, teses etc.). Atualmente existem plataformas especializadas de pesquisa como o Google Acadêmico [Google 2018], IEEE Xplore Digital Library [IEEE 2018], ACM Digital Library [ACM 2018], ScienceDirect [ScienceDirect 2018] e SciELO [SciELO 2018] que auxiliam na busca por trabalhos por meio de critérios de busca. No entanto, elencar trabalhos relevantes é algo quase sempre desafiador, pois exige etapas que demandam tempo, leitura e dedicação.

Os resultados de buscas em plataformas especializadas de pesquisa, normalmente, são disponibilizados na forma de páginas HTML. No entanto, diversos resultados podem

ser obtidos dado um critério utilizado para a busca. Além disso, em uma revisão sistemática pode ser utilizada diversas plataformas especializadas de pesquisa. Para simplificar o processo de busca por trabalhos mais relevantes que outros em diversas plataformas especializadas de pesquisa, surge a necessidade de uma abordagem que auxilie na identificação da relevância entres diversos trabalhos recuperados nas plataformas especializadas de pesquisa.

Como várias plataformas especializadas de pesquisa disponibilizam seus resultados em páginas HTML, surge a necessidade de exploração deste conteúdo para que seja possível algum processamento no intuito de estabelecer relevância entre os trabalhos obtidos nas buscas. Assim, as páginas HTML devem ser representadas em um formato que facilite o processamento dos dados mais relevantes para o problema em questão. Existem diversas formas para representação de documentos em sistemas de recuperação de informação, onde uma forma amplamente utilizada são os modelos baseados em saco de palavras (do inglês, *bag of words*) [Scarpa 2017].

*Bag of words* é uma forma de representação onde um documento é caracterizado como um conjunto de *tokens*, onde a ordem e a ligação entre os *tokens* não são consideradas [Santos et al. 2010]. Carrilho Junior (2007) discute que *bag of words* é uma abordagem puramente estatística e que apresenta bons resultados na literatura. Quando os dados estão disponibilizados em uma forma de representação, técnicas estatísticas e de inteligência artificial podem ser utilizadas para estabelecer relevâncias entres os trabalhos que foram obtidos na etapa da revisão sistemática que busca por trabalhos ou evidências.

Este trabalho tem como objetivo geral desenvolver um sistema para simplificar a busca por trabalhos acadêmicos em diversas plataformas especializadas de pesquisa. Para atingir o objetivo geral do trabalho, alguns objetivos específicos foram elencados: i) estudar o fluxo necessário para elaborar uma revisão sistemática; ii) identificar as etapas da revisão sistemática que podem ser automatizadas de modo a diminuir o esforço humano na busca por trabalhos acadêmicos; iii) estudar técnicas computacionais que podem auxiliar na automatização das etapas identificadas; iv) desenvolver um sistema para automatizar as etapas identificadas por meio das técnicas computacionais estudadas; e v) avaliar, sob a perspectiva do usuário, o ganho alcançado pelo sistema desenvolvido em relação a redução do esforço humano na busca por trabalhos acadêmicos.

Como contribuições, este artigo destaca os seguintes pontos: i) uma aplicação de *bag of words* para o problema de ranqueamento de dados no contexto da revisão sistemática; ii) uma arquitetura de sistema para a diminuição do esforço humano na busca por trabalhos acadêmicos em plataformas especializadas; iii) uma implementação da arquitetura de sistema proposta; e iv) uma avaliação experimental para verificar o ganho alcançado pela implementação desenvolvida.

## **2. Referencial Teórico**

Esta seção tem como objetivo apresentar e discutir, brevemente, o embasamento teórico necessário para uma melhor compreensão da abordagem proposta nesta pesquisa.

### **2.1. Revisão Sistemática**

Revisão Sistemática é definida por [Kitchenham 2004] como a identificação, avaliação e interpretação de toda pesquisa relevante a uma questão de pesquisa popular ou uma área

de pesquisa ou fenômeno de interesse. Segundo Coutinho et al. (2015) uma Revisão Sistemática pode ser dividida em três grandes seções que possuem diversas etapas cada. A primeira etapa, denominada Revisão de Plano, possui as seguintes ações:

- Identificar a necessidade da Revisão: ação onde deve-se constatar o motivo de realizar uma Revisão Sistemática;
- Desenvolver Protocolo de Revisão: ação onde se estabelece um protocolo para a Revisão Sistemática e onde deve-se definir sete pontos.
  1. Perguntas de pesquisa: tem como objetivo guiar o pesquisador através de perguntas-chave;
  2. Linha de busca: definir qual texto será usado como entrada de dados, onde esse texto pode ser utilizado para iniciar a Mineração de Texto;
  3. Fontes de pesquisa: elencar quais bases de dados que serão sujeitas a busca;
  4. Procedimento de pesquisar nas fontes: realização do passo-a-passo para concretizar a busca nas diferentes fontes de pesquisa;
  5. Critérios para inclusão e exclusão dos resultados obtidos: a fim de otimizar o processamento dos dados, deve-se realizar uma limpeza dos dados menos relevantes que foram obtidos no ponto anterior;
  6. Procedimentos de seleção: definir quais informações vão ser extraídas das fontes de pesquisa;
  7. Procedimento de extração de dados: definir o processo de extração, armazenamento e representação dos dados brutos.

Já a segunda etapa, chamada de Condução de Revisão, possui as seguintes ações:

- Identificar estudos primários: devido as particularidades de cada fonte de pesquisa, os resultados obtidos em uma ou mais delas podem não ser satisfatórios, o que pode ocasionar na necessidade de alterar ligeiramente as palavras-chaves ou configurações pertinentes a aquela fonte de pesquisa;
- Selecionar estudos primários: identificam-se quais os resultados que são mais relevantes para o objetivo da pesquisa, mas ainda mantendo uma amostra relevante de dados;
- Analisar qualidade dos estudos: deve-se verificar se as amostras obtidas contém os dados necessários para serem consideradas pertinentes para a pesquisa;
- Extrair Dados: momento da efetivação da pesquisa.

Por fim, a terceira e última etapa, denominada Revisão de Resultado, possui as seguintes ações:

- Apresentar um Resumo dos Resultados: apresenta-se, de forma geral, quais e que tipos de dados foram obtidos de forma a contextualizá-los;
- Apresentar os Resultados das Questões: onde são apresentados os resultados das questões-chave que permeiam a pesquisa.

## **2.2. Mineração de Textos**

Barion e Lago (2008) definem Mineração de Textos como “processo de extração de informação útil (conhecimento) em documentos de textos não-estruturados”, onde pode

ser utilizada na visualização de padrões típicos em quantidades enormes de textos no intuito de encontrar fatos significativos ou relevantes [Carvalho 2017]. Segundo Carrilho Junior (2007) é comum, na literatura, utilizar *bag of words* como forma de representação em mineração de textos. O processo de *bag of words* envolve três várias etapas: categorização de textos, clusterização e sumarização.

Categorização de Textos é a tarefa responsável por identificar e agrupar os textos em categorias predefinidas. O principal objetivo de se categorizar alguma coisa, sejam livros ou textos, é permitir que o acesso a determinado conteúdo seja feito de forma facilitada, sem que haja a necessidade de se despende muito esforço [Carrilho Junior 2007]. Já a etapa de clusterização é o agrupamento dos textos que possuem características similares. Esta etapa é computacionalmente complexa devido à quantidade de variáveis envolvidas, além de não se saber, a priori, quantos grupos podem ser gerados. Por fim, a etapa de sumarização responde pela redução da quantidade de dados a serem processadas, eliminando todos os dados que não são essenciais para a mineração de texto.

Mineração de Textos é uma técnica muito versátil, podendo ser aplicada em praticamente qualquer área que exista registro de informações digitais. Segundo Carrilho Junior (2007), na área de negócios, a mineração pode ser usada para avaliar a satisfação dos clientes de forma que este pode responder de maneira imprecisa e ainda assim prover informações valiosas. Pode ser usada também para analisar dados deixados pelos concorrentes, identificar oportunidades, tendências tecnológicas e outros. É possível também a criação de sistemas automatizados para atendimento ao cliente, evitando a intervenção humana ao mesmo tempo que aumenta-se a velocidade do atendimento. Ainda segundo Carrilho Junior (2007), para grandes empresas é possível a gestão de contratos, agrupando por setor ou assunto evitando redundância ou sobreposição de contratos. É possível também a extração de informações relevantes dos contratos a fim de criar *templates* para usos futuros.

Mineração de Textos também pode ser aplicada no contexto da área jurídica. De acordo com Carrilho Junior (2007) existem um quantitativo de aplicações que se servem da Mineração de Textos como instrumento de obtenção de conhecimento jurisprudencial, objetivando a geração automática de sumários a partir de depoimentos e de relatórios policiais, onde estes documentos e relatórios policiais são comumente escritos em texto livre. Na área da medicina, por sua vez, há um grande acúmulo de informações sobre pacientes ao longo do tempo, pesquisas e registros hospitalares. Assim, médicos tem à disposição muitos dados quem podem ser compilados através da Mineração de Texto para ajudar em diagnósticos futuros.

### 2.3. Formas de Representação de Documentos

Para realização da Mineração de Textos é necessário uma estrutura que represente os dados obtidos. Scarpa (2017) define as seguintes formas de representação de dados:

1. Modelo Vetorial: nesta representação, um documento é visto como um saco de palavras, isto é, as posições das palavras nos documentos não são consideradas, apenas a quantidade de vezes que cada palavra aparece [Scarpa 2017].
2. Representação matricial de um *corpus*: o elemento  $X_{ij}$  dessa matriz  $X$  representa a relação entre o documento  $D_i$  e o termo  $W_j$ . A dimensão da matriz é  $m \times n$ , em que  $m$  é a quantidade de documentos e  $n$  é a quantidade de termos [Scarpa 2017].

Para este trabalho será usada uma representação de Modelo Vetorial, sendo este o modelo *bag of words* já que este modelo representa, assim como descrito anteriormente, um saco de palavras como o nome sugere. Ao realizar a Mineração de Textos, os documentos são lidos e suas palavras relevantes são distribuídas em um formato de dicionário, com a chave sendo a palavra selecionada e o valor sendo o número de vezes que tal palavra aparece no texto.

### 3. Abordagem Proposta

Esta seção apresenta detalhes da solução proposta para simplificar a busca por trabalhos acadêmicos em plataformas especializadas de pesquisa. Assim, é mostrado e discutido a arquitetura de sistema proposta para conduzir a implementação da solução proposta. Além disso, são comentados os aspectos de implementação e destacadas as tecnologias utilizadas. Por fim, os aspectos de funcionalidades da solução proposta são ilustradas e comentadas.

#### 3.1. Arquitetura do Sistema

A abordagem apresentada neste artigo possui a especificação de uma arquitetura de sistema para que possa ser instanciada em qualquer linguagem de programação. A arquitetura especificada foi baseada no modelo cliente/servidor e projetada para que seja possível o funcionamento e uso com múltiplos clientes *web* e *mobile*, visualizando as mesmas regras de negócio. Uma visão geral da arquitetura de sistema especificada pode ser obtida na Figura 1.

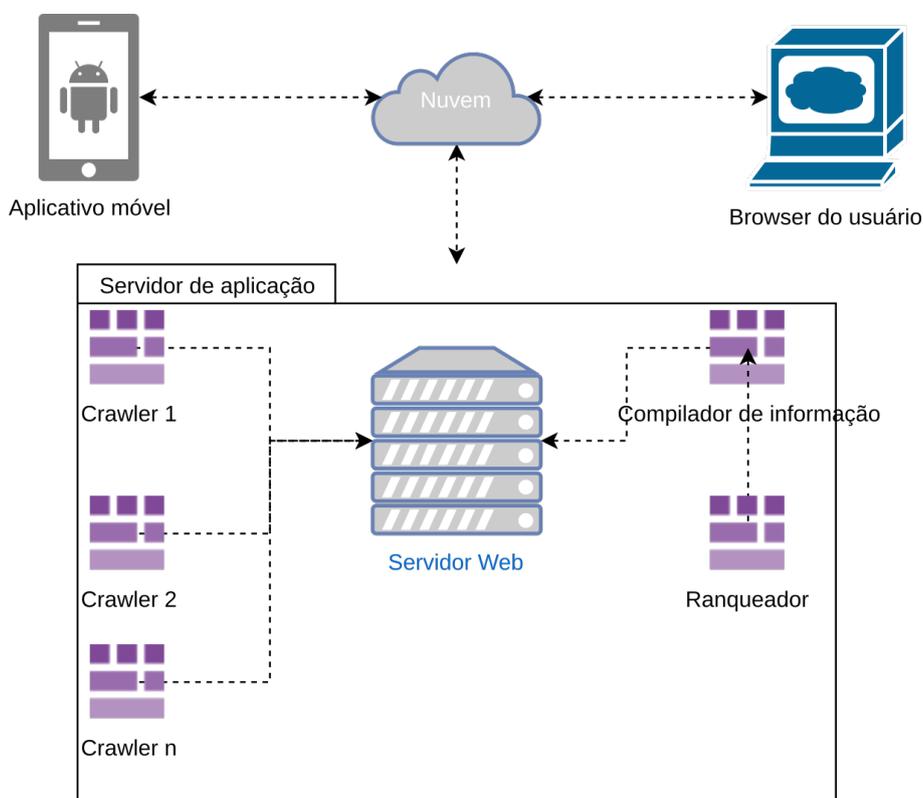


Figura 1. Arquitetura do Sistema

Do ponto de vista dos clientes, a arquitetura de sistema proposta possui interfaces com navegadores *web* e *smartphones*. Para abstrair a comunicação com dois tipos de clientes e para padronizar as regras de negócio do sistema para os tipos de clientes, adotou-se a tecnologia de *web services* como camada de integração entre cliente/servidor. A tecnologia de *web service* oferece funcionalidade de integração, por meio da Internet, em aplicações de diferentes tecnologias e linguagens de programação [Alonso et al. 2004]. Assim, as requisições oriundas dos clientes chegam ao servidor por meio dos *web services*. O servidor de aplicação, ao receber uma requisição, realiza uma comunicação com as plataformas especializadas de pesquisa por meio dos componentes *Crawler*.

O *Crawler* é um software que interpreta o HTML produzido como resposta a uma requisição HTTP e faz a extração de dados relevantes do HTML produzido. Assim, para cada plataforma especializada de pesquisa, existe um *Crawler* que conhece quais argumentos são passados para fazer busca na plataforma e faz a captura dos resultados da busca. Os resultados obtidos pelos componentes *Crawler* são enviados ao componente *Compilador de Informação*. O *Compilador de Informação*, por sua vez, recebe os dados obtidos pelo *Crawler* e faz a estruturação destes dados para facilitar a interpretação e o processamentos destes dados. Após a estruturação dos dados, esse resultado é enviado ao componente *Ranqueador* que vai tratar da ordenação dos resultados de forma a priorizar itens de pesquisa mais relevantes que outros. Por fim, o resultado da ordenação, por meio do *Ranqueador*, é enviado aos clientes como uma resposta do servidor de aplicação dada a requisição inicial.

### 3.2. Aspectos de Implementação

O servidor foi implementado usando a linguagem Python e com o *microframework* Flask<sup>1</sup> já que *microframeworks* são mais simples e menos rígidos quanto ao processo de implementação, possibilitando ao desenvolvedor utilizar metodologias próprias. A diferença do Flask para o principal concorrente em *frameworks* Python, o Django<sup>2</sup>, é a sua simplicidade e flexibilidade, algo que também traz leveza, já que ele possui bem menos ferramentas prontas em suas bibliotecas. O código do servidor foi modularizado de maneira que as responsabilidades de cada classe ficassem o mais clara possível. Para isso o código foi separado em pacotes com nomes sugestivos para os tipos de classes encontradas dentro dele e todo código segue o padrão *Model-View-Controller* (MVC).

Com relação aos clientes que se servem do servidor implementado, existem dois tipos de clientes: navegador *web* e uma aplicação Android<sup>3</sup>. Para o cliente navegador *web* e após as configurações iniciais do Flask, uma interface *web* foi desenvolvida, já que esta é mais simples de construir e testar, provendo posteriormente uma interface de teste para cada nova funcionalidade que fosse implementada na aplicação. Essa interface foi implementada usando HTML, CSS e JavaScript com o uso da biblioteca jQuery<sup>4</sup> para agilizar e facilitar o desenvolvimento da camada lógica dessa interface. A interface também foi desenvolvida usando o padrão MVC.

O funcionamento do servidor dar-se pela integração entre diversos módulos que são demonstrados na Figura 1. Ao receber uma requisição do cliente, o servidor faz uma

<sup>1</sup>Flask. Disponível em: <http://flask.pocoo.org/>. Acesso em: 11 de novembro de 2018.

<sup>2</sup>Django. Disponível em: <https://www.djangoproject.com/>. Acesso em: 11 de novembro de 2018.

<sup>3</sup>Android. Disponível em: <https://www.android.com/>. Acesso em: 11 de dezembro de 2018.

<sup>4</sup>jQuery. Disponível em: <https://jquery.com/>. Acesso em: 11 de dezembro de 2018.

chamada aos módulos *Crawler*, estes possuem uma especialização de uma classe genérica chamada *Crawler* que possui métodos padrão para todas as classes que farão o trabalho de obtenção dos dados usados na Mineração de Texto. Existem objetos que descrevem onde dado *Crawler* irá buscar a informação dentro da página-alvo HTML e quais informações deverão ser usadas durante a busca a depender do que a página tem disponível em seu formulário. Tais fontes de dados possuem seus próprios motores de busca, então em determinados momentos são retornados resultados que não são tão relevantes ou estão em uma língua diferente do português, porém isso se deve a interpretação que um motor específico aplicou a dada palavra-chave. Atualmente a aplicação está usando apenas a palavra-chave como vetor de busca para manter a interface genérica visto que algumas fontes suportam mais informações do que outras.

Após essa fase inicial, a informação é passada para o módulo *Compilador de Informação* que implementa o algoritmo de *bag of words* e realiza a contagem das palavras disponíveis no texto (atualmente apenas no título). Tal informação é passada para o *Ranqueador*, que também possui funcionalidades de limpeza dos dados. Todo conteúdo é retornado em forma de vetor e unido em um vetor maior no controlador da camada de visão correspondente e retornado em formato JSON para o cliente. Para realização do processo de Mineração de Texto apenas a biblioteca Beautiful Soup<sup>5</sup> foi usada como código de terceiros além das bibliotecas-padrão disponíveis no Python 3.6 que foi a versão escolhida para o desenvolvimento da aplicação servidora.

Já para o cliente Android, foi desenvolvido na forma de uma aplicação nativa utilizando a linguagem Kotlin<sup>6</sup>. Mesmo sendo relativamente nova, essa linguagem já possui suporte oficial da Google e possui recursos adicionais sobre a biblioteca padrão Java. O código foi construído usando o padrão *Model-View-Presenter* (MVP) [Stawart 2011] para facilitar o entendimento e manutenibilidade do código. O aplicativo consiste em uma interface simples com duas formas de interação, teclado e voz, cabendo ao usuário decidir usar qual achar melhor.

O servidor encapsula a implementação da técnica de Mineração de Dados, em particular a Mineração de Textos, utilizada neste trabalho. Adotou-se as fases comentadas por Carvalho (2017) onde a Mineração de Dados é mostrada como as seguintes fases distintas: Pré-Processamento de Dados, Mineração de Dados e Pós-Processamento de Dados. A fase de Pré-Processamento de Dados é responsável pela preparação da informação para que ela seja minerada e analisada de maneira eficiente. Na fase de Mineração de Dados ocorre a extração da informação propriamente dita e armazenada em estruturas de dados unificadas, pois a mineração pode ser realizada usando fontes diversas e heterogêneas. Aqui os dados brutos são submetidos a um ou mais algoritmos de mineração. Por fim, na fase de Pós-Processamento de Dados, a informação colhida é transformada e reduzida de forma a se tornar mais compreensível pelo analista. Esse processo pode ser feito através de alterações nos limiares de suporte e confiança ou na remoção de nós da árvore de decisão. A Tabela 1 apresenta um relacionamento entre essas fases e os componentes desenvolvidos para a solução abordada neste projeto.

---

<sup>5</sup>Beautiful Soap. Disponível em: <https://www.crummy.com/software/BeautifulSoup/>. Acesso em: 11 de dezembro de 2018.

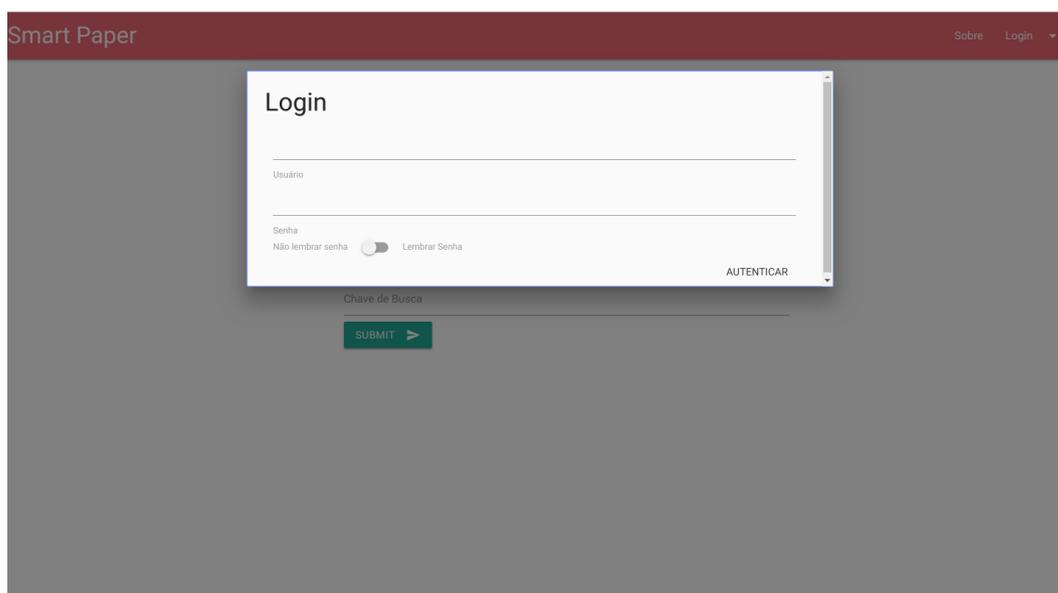
<sup>6</sup>Kotlin. Disponível em: <https://kotlinlang.org/>. Acesso em: 11 de dezembro de 2018.

**Tabela 1. Relação entre fases da Mineração de Textos e os componentes da solução proposta**

Fase	Módulo
Pré-Processamento de Dados	Componentes <i>Crawler</i> (1..n)
Mineração de Texto	Compilador de Informações
Pós Processamento de Dados	Ranqueador

### 3.3. Aspectos Funcionais

Como mencionado, existem dois clientes que foram implementados na solução proposta. Ao iniciar a solução proposta pelo cliente *web*, o usuário visualiza um menu na parte superior direita da tela, conforme Figura 2. Passando o mouse sobre o menu “Login”, o usuário visualiza as opções “Login” e “Cadastro”. Clicando em “Login”, um *pop-up* abre com um pequeno formulário onde o usuário pode inserir suas informações de conta e, quando concluído, pressionar o botão “Login” para efetuar sua autenticação no sistema. No estado atual “Cadastro” e “Login” ainda não são necessários, porém será implementado no futuro a visualização de histórico do usuário.



**Figura 2. Tela para efetuar o login no sistema**

No menu da página inicial do cliente *web* e ao clicar em “Cadastrar”, o usuário é direcionado para a tela responsável pela inserção das informações de conta como mostra a Figura 3. Ao inserir todas as informações obrigatórias e clicar em “Cadastrar”, o usuário é redirecionado para a página inicial do *site*, onde ele pode voltar ao menu citado, clicar em “Entrar” ou realizar o processo de autenticação.

Para realizar uma busca por trabalhos acadêmicos, o usuário, estando na página inicial do cliente *web*, clica sobre a única caixa de texto presente, digita os termos de busca que deseja e clica em “Submit”. A Figura 4 ilustra a tela que permite a consulta por trabalhos acadêmicos. O resultado, então, será exibido em formato de lista ordenada do título do artigo de maior para menor relevância. A Figura 5 demonstra a tela de visualização dos resultados da busca por trabalhos acadêmicos.

A screenshot of a web browser displaying a registration page. The address bar shows 'localhost:5000/signup'. The form contains the following elements: 'Nome' and 'Email' input fields; a 'Usuário' label above an 'Escolher arquivo' button and a 'Foto' label; a 'Senha' input field; and a 'Confirme a Senha' input field. A green 'CADASTRAR' button with a right-pointing arrow is positioned between the password and confirm password fields.

**Figura 3. Tela para registro de um novo usuário**

A screenshot of a web browser showing a search interface. The top navigation bar is red and contains the text 'Smart Paper' on the left and 'Sobre' and 'Login' on the right. Below the navigation bar is a search bar with the label 'Chave de Busca' and a 'SUBMIT' button with a right-pointing arrow.

**Figura 4. Tela que mostra o campo de busca na interface web**

Quanto ao cliente Android, assim que o aplicativo inicia, o usuário pode visualizar uma caixa de texto semelhante a presente na interface *web* e funciona da mesma maneira. Ao inserir um texto e clicar em “Buscar” (Figura 6(a)), o processo de busca ocorre e o usuário visualiza os resultados em forma de lista em uma nova janela porém, no dispositivo móvel, há também um botão de ação que abre uma janela de diálogo para reconhecimento de voz, possibilitando a entrada de dados através da fala (Figura 6(b)). A partir de então os resultados serão mostrados da mesma maneira.

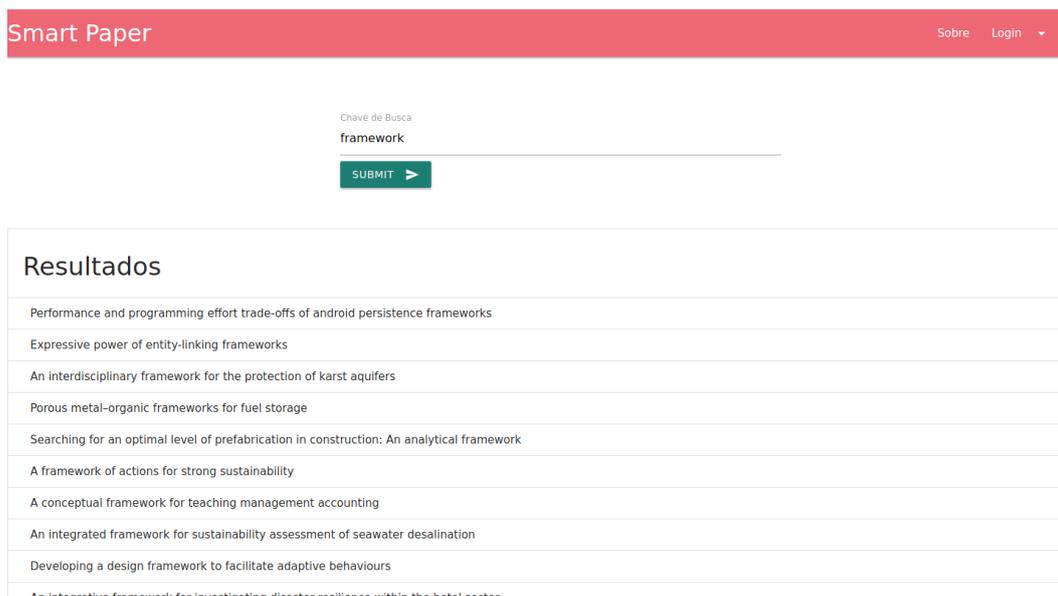


Figura 5. Resultados da ultima busca sendo mostrados

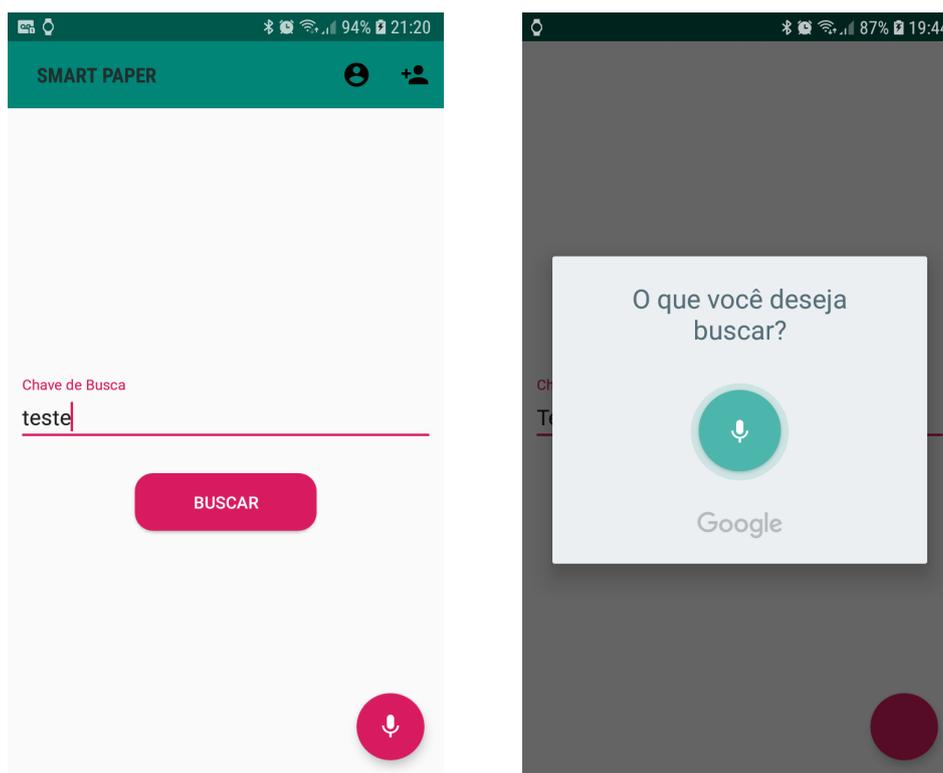


Figura 6. Processos de busca no aplicativo Android

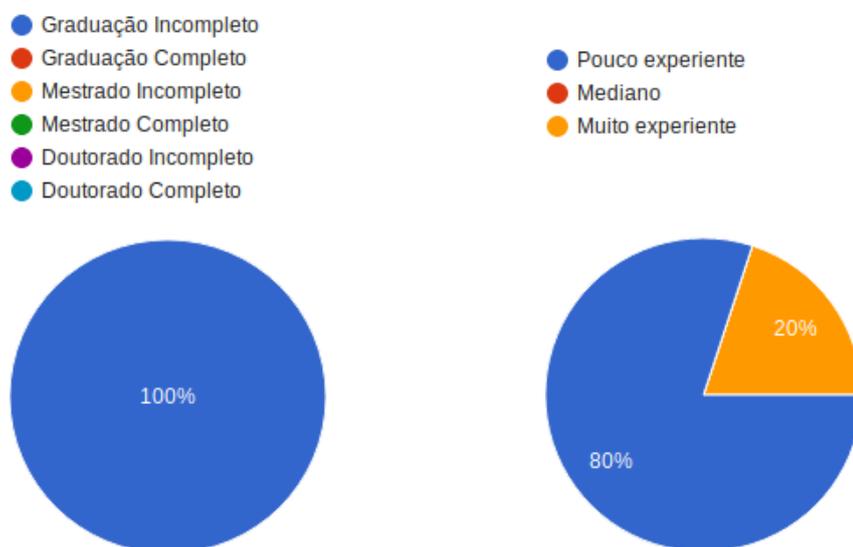
#### 4. Avaliação

A avaliação foi conduzida em duas etapas, provendo ao usuário alguns minutos para utilizar ambas as interfaces da aplicação e então responder um questionário sobre sua experiência com a solução proposta neste artigo. O público-alvo da pesquisa são pesquisadores que, pelo menos, iniciaram uma graduação. Assim, o questionário teve como

objetivo obter dados de pesquisadores que podem ter os seguintes níveis de escolaridade: graduação incompleta, graduação completa, mestrado incompleto, mestrado completo, doutorado incompleto e doutorado completo. Além disso, os pesquisadores devem ter utilizado ferramentas eletrônicas de busca nas suas pesquisas. O questionário consistiu nas seguintes perguntas:

1. (Q1) Qual sua escolaridade?
2. (Q2) Como você se considera como pesquisador?
3. (Q3) Por qual meio você costuma fazer a maior parte das suas pesquisas?
4. (Q4) Como foi a facilidade de uso na *web*?
5. (Q5) Como foi a facilidade de uso no aplicativo *mobile*?
6. (Q6) Alguma sugestão de melhoria com relação as funcionalidades da interface *web*?
7. (Q7) Alguma sugestão de melhoria com relação as funcionalidades do aplicativo móvel?
8. (Q8) Para você como foi a qualidade dos resultados apresentados pela aplicação?
9. (Q9) Você preferiu usar a versão *web* ou *mobile*?

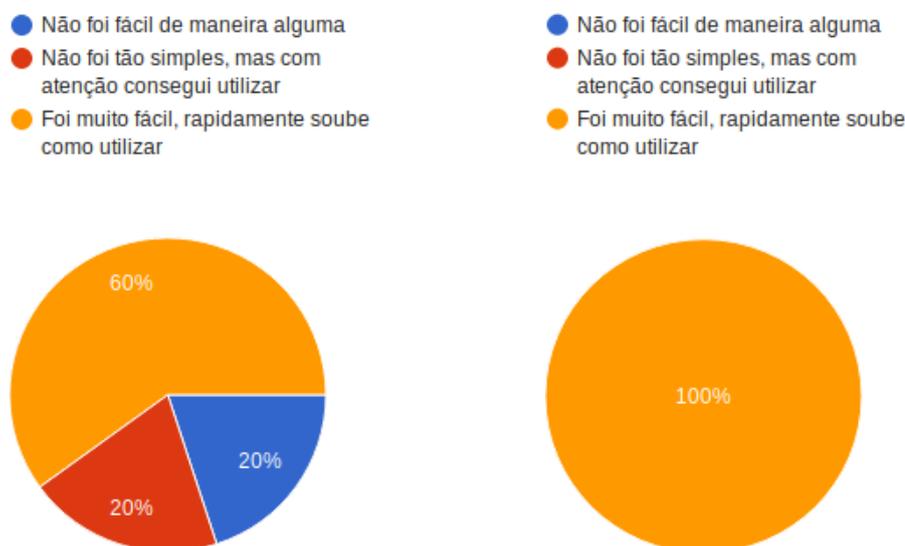
No total, cinco pesquisadores responderam o questionário. A Figura 7(a) mostra que todos os pesquisadores que responderam possuem graduação incompleta. Já a Figura 7(b) mostra que a maioria (80%) do público que respondeu o questionário se considera pouco experiente e apenas uma pessoa se caracterizou como muito experiente.



**Figura 7. a) (Q1) Qual sua escolaridade? e b) (Q2) Como você se considera como pesquisador?**

Com relação a questão Q3, todos as pessoas que responderam afirmaram que costumam fazer maior parte das suas pesquisas por meios eletrônicos (sites, artigos *online* etc.). Nesta questão, as pessoas teriam duas opções de respostas: a) meios físicos (livros, artigos impressos etc.) e meios eletrônicos (sites, artigos *online* etc.). Os resultados demonstram que, para a amostra que foi consultada, há uma tendência da maioria das pesquisas por trabalhos ocorrerem por meios eletrônicos, o que motiva a adoção de ferramentas ou aplicativos do contexto abordado neste presente trabalho. As Figuras 8(a) e

8(b) ilustram os resultados obtidos em relação a facilidade de uso de ambas as versões da aplicação.



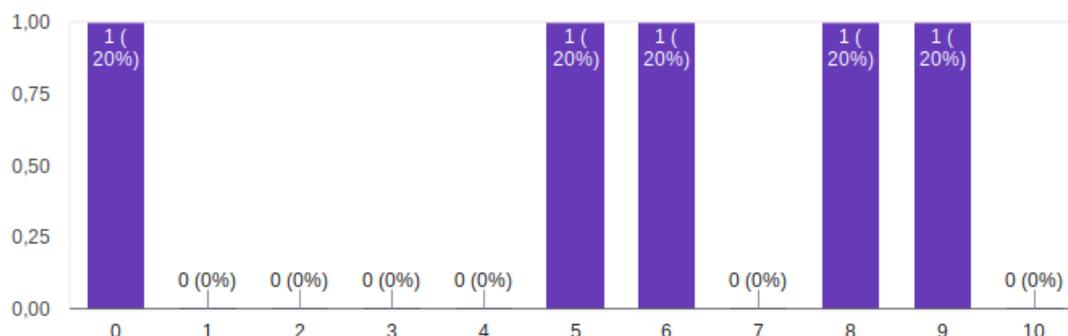
**Figura 8. a) (Q4) Como foi a facilidade de uso na *web*? e b) (Q5) Como foi a facilidade de uso no aplicativo *mobile*?**

As questões Q4 e Q5 têm como objetivo verificar o nível de facilidade, na opinião dos usuários, de ambas as versões da aplicação. Todas as pessoas que utilizaram a aplicação versão *mobile* afirmaram que foi muito fácil de utilizar e que rapidamente souberam como manusear a aplicação. Já para a aplicação, versão *web*, a maioria das pessoas (60%) afirmaram que a aplicação foi fácil de utilizar. No entanto, uma pessoa relatou que não foi tão simples de utilizar e uma outra pessoa afirmou que não foi fácil de maneira alguma. Assim, a versão *web* merece uma atenção maior, em futuras versões, nos aspectos de usabilidade para tornar, mais agradável, a experiência do usuário.

As questões Q6 e Q7 têm como objetivo obter respostas subjetivas para identificar possíveis problemas e coletar sugestões de melhorias para futuras versões da aplicação em ambas as versões. Com relação a versão *web*, houveram quatro respostas no total. Duas respostas sugeriram a exibição de uma barra de progresso entre a ação de buscar trabalhos e a exibição dos resultados. Segundo um usuário, existe a sensação que a ação falhou, pois não existe um indicativo que está sendo processada a busca. Um outro usuário sugeriu a adição de uma lista de termos frequentes e relacionados. Houve também uma demanda por filtro por áreas, para facilitar a categorização de trabalhos. Por fim, um usuário relatou que a experiência do usuário não foi agradável, a busca que foi feita não trouxe resultados e que faltava responsividade no site. Em relação a versão *mobile* houveram três respostas, um usuário recomendou uma melhor estruturação para deixar o visual mais agradável por meio de cores, negrito etc. Já um outro usuário sugeriu adotar componentes para ilustrar o carregamento dos usuários e filtros por áreas. Por fim, um usuário relatou na versão *mobile* deve-se considerar as mesmas sugestões da versão *web*.

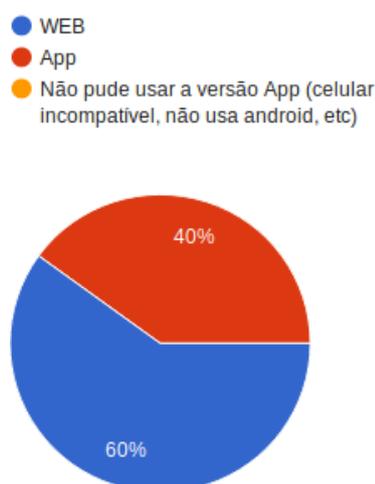
A questão Q8 tem como objetivo obter a opinião do pesquisador sobre a qualidade dos resultados apresentados pela solução desenvolvida. Os resultados obtidos na questão Q8 podem ser visualizados na Figura 9. O eixo y da Figura 9 exibe o número de usuários

que responderam, já o eixo x retrata o nível de qualidade assinalado pelos usuários, onde 0 indica a mais baixa qualidade e 10 a mais alta qualidade. De acordo como os resultados obtidos, pode-se perceber que a ferramenta apresentou um nível bom de qualidade dos resultados apresentados, apenas um usuário indicou uma mais baixa qualidade nos resultados.



**Figura 9. (Q8) Para você como foi a qualidade dos resultados apresentados pela aplicação?**

Por fim, a questão Q9 tem como objetivo verificar qual foi a versão do aplicação que mais agradou. A Figura 10 ilustra os resultados obtidos na questão Q9. De acordo com os resultados, a maioria (60%) dos pesquisadores preferiram utilizar a versão *web*. Já os demais pesquisadores apontaram que preferiram utilizar a versão *mobile*. Apesar da versão *web* ter sido a mais criticada pelos usuários, foi a versão que obteve mais preferência. Acredita-se que a justificativa seja pelo fato de que uma aplicação *web* fornece um amplo acesso maior que as aplicações *mobile*, pois necessita apenas de conectividade e um navegador, diferentemente de algumas aplicações *mobile* que exigem sistema operacional específico, conectividade e instalação do aplicativo.



**Figura 10. (Q9) Você preferiu usar a versão *web* ou *mobile*?**

## 5. Conclusão

Este trabalho apresentou um sistema que tem como objetivo simplificar a busca por trabalhos acadêmicos em plataformas especializadas de pesquisa. Assim, foi feito um estudo sobre técnicas que pudessem auxiliar no desenvolvimento da solução apresentada e

também como ocorre uma revisão sistemática. O sistema proposto foi projetado e implementado em duas versões, *web* e *mobile*. Os aspectos funcionais e de implementação foram discutidos e, por fim, as versões desenvolvidas foram avaliadas por um grupo de usuários. De acordo com os resultados, a aplicação conseguiu auxiliar e agradar de alguma forma na busca por trabalhos acadêmicos.

Durante a avaliação foram levantadas sugestões de melhorias e, pretende-se, como trabalhos futuros, atender os aspectos para agregar facilidade de uso e melhorar a experiência do usuário. Outro ponto importante, que deve ser considerado em futuras versões, é a adoção de outras técnicas ou algoritmos no âmbito da Inteligência Artificial que podem apresentar resultados melhores na busca por trabalhos acadêmicos. Além disso, pretende-se, também, avaliar ambas as versões da solução proposta com um número maior de usuário e, assim, obter mais dados sobre o que pode ser melhorado. O estado atual do trabalho contempla apenas a comunicação com duas plataformas especializadas de pesquisa, em versões futuras almeja-se agregar mais plataformas especializadas de pesquisa.

## Referências

- ACM (2018). *ACM Digital Library*. Disponível em: <https://dl.acm.org/>. Acessado em: 16 de agosto de 2018.
- Alonso, G., Casati, F., Kuno, H., and Machiraju, V. (2004). *Web Services: Concepts, Architectures and Applications*. Data-Centric Systems and Applications. Springer, Berlin.
- Barion, E. C. N. and Lago, D. (2008). Mineração de textos. *Revista de Ciências Exatas e Tecnologia*, 3:123–140.
- Carrilho Junior, J. R. (2007). Desenvolvimento de uma Metodologia para Mineração de Textos. Dissertação de Mestrado (Programa de Pós-Graduação em Engenharia Elétrica), Pontifícia Universidade Católica do Rio de Janeiro (PUC-RJ), Rio de Janeiro, RJ, Brasil.
- Carvalho, R. C. (2017). Aplicação de Técnicas de Mineração de Texto na Recuperação de Informação Clínica em Prontuário Eletrônico do Paciente. Dissertação de Mestrado (Programa de Pós-Graduação em Ciência da Informação da Faculdade de Filosofia e Ciências), Universidade Estadual Paulista “Júlio de Mesquita Filho” (UNESP), Marília, SP, Brasil.
- Coutinho, E. F., de Carvalho Sousa, F. R., Rego, P. A. L., Gomes, D. G., and de Souza, J. N. (2015). Elasticity in cloud computing: a survey. *annals of telecommunications - annales des télécommunications*, 70(7):289–309.
- Google (2018). *Google Acadêmico*. Disponível em: <https://scholar.google.com.br/>. Acessado em: 16 de agosto de 2018.
- IEEE (2018). *IEEE Xplore Digital Library*. Disponível em: <https://ieeexplore.ieee.org/>. Acessado em: 16 de agosto de 2018.
- Kitchenham, B. (2004). Procedures for performing systematic reviews.
- Sampaio, R. and Mancini, M. (2007). Estudos de revisão sistemática: um guia para síntese criteriosa da evidência científica. *Brazilian Journal of Physical Therapy*, 11:83 – 89.

- Santos, L. M., Esmin, A. A. A., Zambalde, A. L., and Nobre, F. M. (2010). Twitter, análise de sentimento e desenvolvimento de produtos: Quanto os usuários estão expressando suas opiniões? *Revista de Ciências e Tecnologias de Informação e Comunicação*, 1(13):159–170.
- Scarpa, A. D. (2017). Técnicas de Processamento de Linguagem Natural Aplicadas às Ciências Sociais. Dissertação de Mestrado (Programa de Pós-Graduação em Matemática Aplicada), Fundação Getúlio Vargas (FGV), Rio de Janeiro, RJ, Brasil.
- SciELO (2018). *SciELO - Scientific Electronic Library Online*. Disponível em: <http://www.scielo.org/>. Acessado em: 16 de agosto de 2018.
- ScienceDirect (2018). *ScienceDirect.com*. Disponível em: <https://www.sciencedirect.com/>. Acessado em: 16 de agosto de 2018.
- Stawart, D. (2011). *Model-View-Presenter*. Dicho.