

**UMA LINGUAGEM VISUAL PARA DEFINIÇÃO
DE STRINGS DE BUSCAS ACADÊMICAS**

PRICILA RESENDE RODRIGUES

UMA LINGUAGEM VISUAL PARA DEFINIÇÃO
DE STRINGS DE BUSCAS ACADÊMICAS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

ORIENTADOR: RAQUEL OLIVEIRA PRATES

Belo Horizonte

Março de 2017

© 2017, Pricila Resende Rodrigues.
Todos os direitos reservados

busca Acadêmica, Linguagem Visual, Interface de Busca.

Ficha catalográfica elaborada pela Biblioteca do ICEx - UFMG

Rodrigues, Pricila Resende.

R696l Uma linguagem visual para definição de strings de busca acadêmica. / Pricila Resende Rodrigues. – Belo Horizonte, 2017.
xxii, 128 f.: il.; 29 cm.

Dissertação (mestrado) - Universidade Federal de Minas Gerais – Departamento de Ciência da Computação.

Orientadora: Raquel Oliveira Prates

1. Computação – Teses. 2. Recuperação da informação. 3. Interfaces de usuário (Sistema de computador). 4. Linguagem Visual. I. Orientadora. II. Título.

CDU 519.6*73(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Uma linguagem visual para definição de strings de busca acadêmica

PRICILA RESENDE RODRIGUES

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROFA. RAQUEL OLIVEIRA PRATES - Orientadora
Departamento de Ciência da Computação - UFMG

PROFA. CLARISSE SIECKENIUS DE SOUZA
Departamento de Informática - PUC - Rio

PROF. MARCOS ANDRÉ GONÇALVES
Departamento de Ciência da Computação - UFMG

PROFA. MARILIA LYRA BERGAMO
Departamento de Desenho - UFMG

Belo Horizonte, 31 de março de 2017.

Aos meus pais.

Agradecimentos

Agradeço a Deus pelos dons que me deu e a Nossa Senhora por sempre me proteger.

Aos meus pais, por me ensinarem a voar e a seguir em frente, não medindo esforços para que meus sonhos se realizassem. À minha irmã Michelly, por sempre estar por perto cuidando de mim. À minha irmã Rafaela, que foi o motivo de eu ter começado a aprender arte digital e ser fonte de inspiração. Ao meu cunhado Ricardo, que pelo exemplo me ensinou muitas coisas em computação. Aos meus sobrinhos, Mariana e Gabriel, por todos os sorrisos, abraços e momentos que me fazem mais feliz. Aos meus avôs e demais familiares, obrigada por sempre acreditarem em mim.

Às minhas amigas, as SI Girls, Iasmini, Kemilly, Mariana e Leidiane, que me acompanharam de perto durante a graduação e de longe durante o mestrado. Mas que mesmo distantes, sempre se fizeram presentes, me dando forças para continuar e torcendo por meu sucesso.

Aos amigos que fiz no mestrado, Tati, Francisco, Fabrício, Lídia, Luiz, Érica, Diego, Bárbara e Júlio que tornaram os dias mais felizes. Em especial, pelos momentos de descontração, com joguinhos, filmes e séries, que deixaram o peso das obrigações mais leve. A todo o pessoal do PENSi, por todos os momentos de compartilhamento de ideias e aprendizado.

À minha orientadora, Raquel, por me receber e me guiar durante o desenvolvimento da pesquisa. Obrigada pelo apoio, incentivo e todos os ensinamentos e dicas valiosas compartilhadas. À professora Marília, pelas tardes de discussão e cheias de ideias durante o desenvolvimento da representação visual da linguagem.

Aos voluntários que participaram dos testes de avaliação. Ao Caíque por me auxiliar durante a realização das avaliações, fazendo observações importantes. Aos membros da banca, professores Clarisse de Souza, Marcos André e Marília, pela avaliação e discussão. Ao Departamento de Ciência da Computação da UFMG, pela oportunidade do mestrado e disposição em ajudar sempre que necessário. Ao CNPq pelo apoio financeiro recebido.

Por fim, agradeço a todos que de alguma forma contribuíram para o meu sucesso.

“Vamos pensar o impensável, vamos fazer o impossível. Vamos nos preparar para lidar com o próprio inefável, e ver se não podemos expressá-lo depois.”

(Douglas Adams)

Resumo

Todos os dias, milhões de pessoas se envolvem na Recuperação de Informação, seja por meio de buscas em sites especializados ou consultas ao seu e-mail. Uma vez que a quantidade de informação disponível é maior do que a capacidade das pessoas de buscá-las e abstraí-las, faz-se necessário o desenvolvimento de linguagens e ferramentas que possibilitem a formulação de *string* (perguntas de busca) mais específicas e com isso, os resultados gerados sejam melhores. No contexto acadêmico, a busca por artigos é uma constante. Contudo, a definição de uma *string* adequada aos resultados esperados, consome tempo e, somado a isso, cada sistema de busca possui uma sintaxe própria, levando o usuário a definir *string* específicas para cada sistema. Entende-se que a definição de *string* apoiada pelo uso de linguagens visuais, permite ao usuário a compreensão da estrutura da *string* e amplia sua capacidade de expressar a necessidade de informação, ao mesmo tempo em que pode otimizar seu tempo, gerando consultas mais completas. Assim, buscando alternativas para tornar mais eficiente a definição de uma *string* de busca no contexto acadêmico, o objetivo deste projeto é a definição de uma linguagem visual para buscas acadêmicas e uma ferramenta web para interação com esta linguagem. A definição da linguagem visual, nomeada VILAS, ocorreu a partir da categorização dos elementos e linguagens presentes nos sistemas de busca inspecionados sob a perspectiva da Engenharia Semiótica. Além disso, foram aplicados conceitos de design da Gestalt e de usabilidade, não somente para a definição da representação visual, como também para o *design* da interface da ferramenta *web* desenvolvida. Assim, com o uso da ferramenta *web* é possibilitado ao usuário, além da definição e manipulação de uma *string* de busca genérica, a tradução desta *string* para a sintaxe de diferentes sistemas de busca. Posteriormente, foram realizados testes qualitativos, de modo a investigar o entendimento dos usuários quanto as representações visuais da linguagem, e avaliar o uso da ferramenta *web* para a definição de *strings* utilizando a linguagem VILAS.

Palavras-chave: Busca Acadêmica, Linguagem Visual, Interface de Busca.

Abstract

Everyday, searches on specialized sites or through queries to ones email, involve millions of people in Information Retrieval. Since the amount of information available is greater than people's ability to seek and abstract them, it is necessary the development of languages and tools that allow the formulation of more specific strings (search questions) and with that, generate better results. In the academic context, the search for articles is constant. However, the definition of an appropriate string for the expected results consumes time; and, added to this, each search system has its own syntax, leading the user to define specific strings for each system. The definition of strings supported by the use of visual languages allows users to understand the string structure and expands their capacity to express the need for information, while it can optimize their time, generating more complete queries. Thus, looking for alternatives to make the definition of a search string in the academic context more efficient, the objective of this project was the definition of a visual language for academic search and a web tool for interaction with this language. The definition of the visual language, called VILAS, occurred from the categorization of elements and languages presents in the search systems inspected from the perspective of Semiotic Engineering. In addition, concepts of Gestalt design and usability were applied, not only for the definition of visual representation, but also for the interface design of the web tool developed. Thus, the web tool allows users to not only define and manipulate a generic search string, but also translate this string to the language expressing search strings of different search systems. Subsequently, qualitative tests were conducted in order to investigate the users' understanding of the visual representations of the language, and to evaluate the communicability of the web tool for the definition of strings using the VILAS language.

Keywords: Academic Search, Visual Language, Search Interface.

Lista de Figuras

1.1	Ciclo de refinamento de uma <i>string</i>	2
1.2	Passos da Metodologia	6
2.1	Fragmento da interface de busca básica do <i>EBSCOHost</i>	14
2.2	Fragmento da interface de busca visual do <i>EBSCOHost</i>	15
2.3	Exemplo da proposta de <i>redesign</i> da Wikipédia	18
2.4	Exemplo de consulta em <i>QueryVOWL</i>	20
2.5	Exemplo do sistema que implementa <i>VXQ</i>	22
2.6	Fragmento da interface de busca do navegador <i>RB</i>	23
2.7	Fragmento da interface de busca do navegador <i>RB++</i>	24
2.8	Construção da representação <i>InfoCrystal</i>	25
2.9	Representação de uma <i>string</i> em <i>InfoCrystal</i>	26
2.10	Representação de uma consulta em <i>VISUAL</i>	27
3.1	Passos de execução do MIS	33
3.2	Fragmento 1 da interface da ACM DL	38
3.3	Fragmento 2 da interface da ACM DL	39
3.4	Fragmento 3 da interface da ACM DL	39
3.5	Fragmento 4 da interface da ACM DL	40
3.6	Fragmento 1 da interface da IEEE	44
3.7	Fragmento 2 da interface da IEEE	45
3.8	Fragmento 3 da interface da IEEE	45
3.9	Fragmento 4 da interface da IEEE	46
3.10	Fragmento 5 da interface da IEEE	46
3.11	Fragmento 1 da interface do Google Scholar	48
3.12	Fragmento 2 da interface do Google Scholar	49
3.13	Fragmento 3 da interface do Google Scholar	49
3.14	Exemplo de agrupamento na ACM	52

3.15	Exemplo de signos de busca simples	56
3.16	Exemplos de signos de busca avançada na interface	58
3.17	Exemplos de signos sintáticos da busca	59
3.18	Exemplos de signos de refinamento dos resultados da busca	60
4.1	Protótipos de VILAS	65
4.2	VILAS - Exemplo da representação de uma <i>string</i>	66
4.3	VILAS - Exemplo da representação do agrupamento AND	66
4.4	VILAS - Exemplo da representação do agrupamento OR	67
4.5	VILAS - Exemplo da representação de uma expressão	67
4.6	VILAS - Exemplo da representação NOT	67
4.7	VILAS - Exemplo genérico de uma <i>string</i> complexa	68
4.8	VILAS - Exemplo de uma <i>string</i> complexa equivalente à representação textual	69
5.1	Mapa da interface da ferramenta VILAS-Web	78
5.2	Fragmento das páginas de ajuda em VILAS-Web	79
5.3	Inserção de palavras em VILAS-Web	79
5.4	Inserção de expressões em VILAS-Web	80
5.5	Área de edição visual da <i>string</i> de busca em VILAS-Web	80
5.6	Área de exibição da <i>string</i> de busca genérica em VILAS-Web	81
5.7	Conjunto de botões para refinamento da busca em VILAS-Web	82
5.8	Inserção de termos na interface	83
5.9	Agrupamento de dois termos com o operador lógico OR	83
5.10	Representação da <i>string</i> de busca depois da ação de agrupamento OR	84
5.11	Agrupamento de dois termos com o operador lógico AND	84
5.12	Representação da <i>string</i> de busca depois do ação de agrupamento AND	85
5.13	Exclusão de um termo da <i>string</i> de busca	85
5.14	Representação da <i>string</i> de busca depois da ação de exclusão de um termo	86
5.15	Adição do filtro de ano inicial na <i>string</i> de busca	86
5.16	Representação final da <i>string</i> de busca	87
5.17	<i>String</i> de busca traduzida para a biblioteca da ACM	88
5.18	<i>String</i> de busca traduzida para a biblioteca da IEEE	88
5.19	Listagem das <i>strings</i> de busca traduzidas para o Google Scholar	89
5.20	Representação de uma das <i>strings</i> de busca traduzida para o Google Scholar	89
6.1	Representações do agrupamento OR	94
6.2	Representações do agrupamento AND	94
6.3	<i>Strings</i> de busca definidas pelos participantes da avaliação	97

Lista de Tabelas

3.1	Sistemas de busca acadêmica	34
5.1	Restrições para definição da <i>string</i>	90

Sumário

Agradecimentos	ix
Resumo	xiii
Abstract	xv
Lista de Figuras	xvii
Lista de Tabelas	xix
1 Introdução	1
1.1 Objetivos	4
1.1.1 Objetivos Específicos	4
1.2 Justificativa	4
1.3 Metodologia	6
2 Fundamentação Teórica	9
2.1 Linguagem Visual	9
2.1.1 Especificação	10
2.1.2 Linguagens Visuais em Computação	11
2.2 Trabalhos Relacionados	12
2.2.1 Busca de Artigos	13
2.2.2 Busca em Banco de Dados	19
2.2.3 Representações Visuais	24
2.3 Contextualização	28
3 Inspeção Semiótica de Interfaces de Busca Acadêmica	31
3.1 Método de Inspeção Semiótica (MIS)	32
3.2 Aplicação do MIS	33
3.2.1 Metamensagem da Interface da Biblioteca Digital da ACM	35

3.2.2	Metamensagem da Interface da Biblioteca Digital da IEEE . . .	38
3.2.3	Metamensagem da Interface do Site de Busca Google Scholar . .	44
3.3	Resultados da Triangulação	48
3.3.1	Apreciação da Inspeção em Interfaces de Busca Acadêmica . . .	50
3.3.2	Linguagem de Busca Acadêmica	53
3.3.3	Estratégias de Interação	55
4	A Linguagem Visual VILAS	63
4.1	Processo de Definição de VILAS	63
4.2	Representação Visual	65
4.3	Discussão	69
5	A Ferramenta VILAS-Web	77
5.1	Interface	78
5.2	Cenário de Uso	81
5.3	Limitações da Ferramenta	87
6	Avaliação da Linguagem VILAS	91
6.1	Avaliação da Representação Visual	91
6.1.1	Discussão	93
6.2	Avaliação com uso do Protótipo	95
6.2.1	Discussão	100
6.3	Avaliação das Dimensões Cognitivas	101
6.3.1	Discussão	103
7	Conclusão	111
	Referências Bibliográficas	115
	Apêndice A Avaliação da Representação Visual	119
	Apêndice B Avaliação com Uso do Protótipo	125

Capítulo 1

Introdução

As possibilidades de criação e armazenamento de informação em dispositivos computacionais têm contribuído para o crescente número de documentos, sejam eles impressos ou digitais. Existe, não somente a necessidade de armazenamento de informação, mas também da busca por informação. Antes da era dos computadores, as buscas eram realizadas em numerosos volumes de livros, as chamadas enciclopédias. Com o advento da Internet, as enciclopédias se tornaram virtuais. Surgiram então sites com o objetivo de fornecer uma maneira das pessoas perguntarem e obterem suas respostas, sem para isso procurar por páginas e páginas de impressos. Os esforços para a obtenção de melhores sistemas de busca deram origem ao campo de pesquisa de Recuperação de Informação (RI) [28; 1]. Os diversos aspectos da recuperação de informação, desde a indexação dos dados na internet à exibição dos resultados da busca, são assunto recorrente na literatura. Uma definição formal de RI dentro do meio acadêmico é apresentada por Manning et al., onde “RI é encontrar material (geralmente documentos) de natureza não estruturada (geralmente textos) que satisfaça uma necessidade de informação a partir de grandes coleções (geralmente armazenados em computadores)”.

RI abrange suporte a usuários em navegação ou filtragem de coleções de documentos, processamento de conjuntos de dados recuperados e classificação de documentos. Para Manning et al., sistemas de RI são distinguidos em três principais escalas:

- Busca *web*: busca ao longo de bilhões de documentos armazenados em milhões de computadores;
- Recuperação de informação pessoal: busca em *e-mails* ou em computadores pessoais;
- Busca de informação empresarial, institucional ou de domínio específico: busca

em sistemas de arquivo centralizados ou em um pequeno número de máquinas dedicadas.

A interação do usuário com um sistema de RI se inicia a partir da necessidade de informação com a seleção do sistema de busca e definição da *string*, obtenção, avaliação e interpretação dos resultados e, por fim, a decisão de obter mais informações ou de parar a busca [1]. Para formular uma *string*, o usuário deve selecionar coleções, metadados ou conjuntos de informações, onde possivelmente encontrará as informações desejadas e, então, especificar palavras, frases, descritores ou outros tipos de informação para serem comparados com as informações nas coleções [1]. O sistema de busca gera como resultado à *string* do usuário conjuntos de dados, metadados ou outros tipos de informação, exibidos na tela em listas ou outro modelo. A maioria dos sistemas de RI para textos completos e bibliográficos suportam apenas o uso de *strings booleanas*, nas quais a pergunta do usuário é expressa por palavras ligadas por expressões lógicas booleanas (OR, AND, NOT) [1].

A definição de uma *string* de busca é um processo que na maioria das vezes percorre um ciclo, no qual o usuário define uma *string* curta e depois de observar os resultados modifica os termos inseridos, em um processo de refinamento [23; 38; 22]. Considerando este processo, é exibido na Figura 1.1 uma representação do ciclo de refinamento da *string* proposta por Jones. Neste ciclo, após formular uma *string* inicial, o usuário passa pelas etapas de submissão da *string*, espera, recebimento e avaliação de resultados e, por fim, refinamento da *string*, seguindo um ciclo a partir daí.

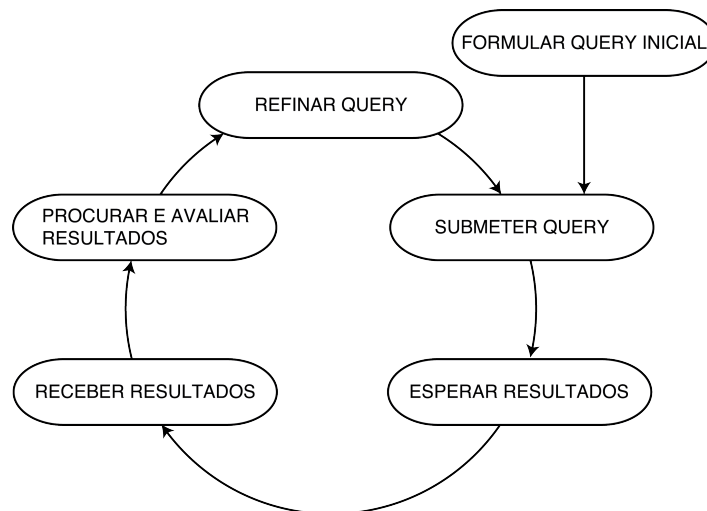


Figura 1.1. Ciclo de refinamento de uma *string*

Figura adaptada de [23]

Todos os dias, milhões de pessoas se envolvem na recuperação de informação, seja

por meio de buscas em sites especializados ou consultas ao seu *e-mail* [28]. Uma vez que a quantidade de informação disponível é maior do que a capacidade das pessoas de buscá-las e abstraí-las, faz-se necessário o desenvolvimento de linguagens e ferramentas que possibilitem a formulação de *strings* (perguntas de busca) mais específicas e com isso, os resultados gerados sejam melhores. No contexto acadêmico, a busca por artigos é uma constante. Contudo, a definição de uma *string* adequada aos resultados esperados, consome tempo e, somado a isso, muitas vezes, definem-se diferentes *strings* para os diferentes sistemas de busca, para obtenção de resultados no mesmo contexto de pesquisa.

Estratégias para o desenvolvimento de interfaces de usuário para recuperação de informação, apresentam embasamento teórico em princípios de *design* e usabilidade. É através da interface que o usuário expressa a necessidade de informação, traduzida em *strings* de busca. A definição da *string* de busca pelo usuário é um ciclo constante, no ele busca, avalia e refina seus objetivos de busca até se sentir satisfeito com os resultados obtidos, ou desistir da busca. Dado que a natureza da visão humana é ser analítica e sintética, de modo a receber e conservar um número infinito de unidades de informação numa fração de segundos [14], considera-se a definição de *strings* visuais uma alternativa à apresentação de *strings* textuais. Assim, a definição de *strings* apoiada pelo uso de linguagens visuais, visa permitir ao usuário a compreensão da estrutura da *string* e ampliar sua capacidade de expressar sua necessidade de informação.

Neste contexto, com o objetivo de investigar formas de tornar o processo de busca mais simples para o usuário, este trabalho trata da definição de uma linguagem visual para buscas acadêmicas. Para tanto, no Capítulo 2, são discutidos aspectos relacionados a linguagens visuais em geral, relacionando outros trabalhos existentes na literatura que também se dedicam a linguagens visuais e interfaces interativas para busca. Uma vez que a linguagem proposta neste trabalho é apresentada em um domínio já existente, sendo uma alternativa à interfaces de buscas textuais, interfaces de busca acadêmica consolidadas foram investigadas e são assunto no Capítulo 3, a fim de entender quais os elementos e ações são essenciais dentro deste domínio. No Capítulo 4 é apresentada a linguagem visual para busca acadêmica proposta neste trabalho, bem como seus elementos e ações possíveis, além de uma breve contextualização do processo de definição da linguagem e os aspectos de *design* levados em consideração no momento da definição. Assim, tendo definido a linguagem visual, no Capítulo 5 é apresentada a ferramenta *web* desenvolvida para implementação dessa linguagem e utilizada como objeto de avaliação da mesma na aplicação de testes com usuários, assunto no Capítulo 6. Foram realizados duas avaliações distintas com usuários, a partir das quais foram identificados pontos fortes e fracos da linguagem e da ferramenta que a implementa. Por fim,

as contribuições deste trabalho para a Ciência da Computação e pesquisas centradas no desenvolvimento de interfaces para Recuperação de Informação, são discutidas no Capítulo 7.

1.1 Objetivos

O objetivo deste projeto foi o desenvolvimento de uma linguagem visual para definição de *strings* no âmbito acadêmico, de modo a permitir ao usuário a visualização, manipulação e alteração da *string* de busca. Além disso, disponibilizar um ambiente para definição da *string* de busca e tradução desta para os diversos sistemas de recuperação de informação acadêmica.

1.1.1 Objetivos Específicos

Os objetivos específicos foram:

- Análise de interfaces de recuperação de informação acadêmica;
- Investigação do uso de uma linguagem visual para criação de *strings*;
- Definição de uma linguagem visual de busca para definição de *strings*;
- Desenvolvimento de uma ferramenta para aplicação da linguagem visual definida;
- Avaliação da linguagem visual através de testes com potenciais usuários.

1.2 Justificativa

A Internet é presença constante na vida de cada vez mais pessoas, que a utilizam por meio de computadores *desktop*, *laptops* ou outros dispositivos portáteis como *smartphones*. Não obstante, o tempo das pessoas se torna mais escasso à medida que aumenta o volume de informações, destacando-se a necessidade de ferramentas e sistemas com a capacidade de otimizar o tempo do usuário. No meio acadêmico, uma atividade relevante a que pesquisadores se dedicam é a busca por trabalhos pertinentes ao seu tópico de pesquisa. Contudo, muitas das vezes, o pesquisador não consegue encontrar resultados tão bons quanto desejado ou simplesmente não consegue formular uma busca com a qualidade necessária para a obtenção desses resultados [1]. Esses problemas, em geral, podem ser decorrentes da falta de padronização dos sistemas de busca e também do desconhecimento dos critérios de definição da *string* de busca utilizado pelo sistema.

Por exemplo, enquanto alguns sistemas tratam o espaço entre as palavras inseridas na busca como uma combinação OR, outros sistemas tratam como uma combinação AND. Além disso, as nomenclaturas utilizadas nestes sistemas varia, como é o caso de quando o usuário deseja inserir o nome da publicação, que é apresentado como “Publisher” ou “Publish in”. Considerando-se, portanto, a definição de uma linguagem padrão para definição de buscas um desafio a ser tratado neste trabalho.

Sistemas de Recuperação de Informação são utilizados por universidades, bibliotecas e entidades a fim de fornecer aos usuários acesso à livros, artigos, revistas e outros documentos. Porém, cada um dos sistemas apresentados ao usuário possui diferentes formas e critérios de uso, o que muitas vezes gera frustrações e buscas incompletas. Ao resolver problemas, os seres humanos tendem a usar representações internas e externas, sendo aquelas armazenadas em seus cérebros ou gravadas em um papel ou outro, respectivamente [26]. Através de sistemas e representações gráficas, tem-se o poder de expressar abstrações em um nível diferente de outras linguagens, como por exemplo as puramente lógicas. Desse modo, observa-se que abstrações em linguagens visuais se aproximam mais do discurso em linguagem natural, do que linguagens lógicas totalmente abstratas [37]. Assim, encontra-se no uso de linguagens visuais no contexto computacional a capacidade de construir abstrações por meio de representações visuais. Onde a natureza do cérebro humano quanto a associação de representações visuais e resolução de problemas, é extrapolada para o entendimento do significado dessas representações.

Menções à princípios de Semiótica ou teorias em Interação Humano Computador (IHC) concentram-se, em sua maioria, em aspectos de interfaces gráfica de usuário e linguagens visuais [9]. Assim, a integração da perspectiva semiótica ao desenvolvimento de uma interface simples e intuitiva, provida de uma linguagem visual para definição da *strings* de busca, bem como de ferramentas para a geração de *strings* de busca para os diversos sistemas de busca acadêmica, permite o desenvolvimento de uma interface alternativa para a definição de *strings* de buscas.

Por meio da linguagem visual para definição de *strings* de busca proposta neste projeto, o usuário poderá interagir com sua *string* por meio de elementos gráficos, podendo inserir termos, manipular relações e/ou alterar informações. Buscando, por meio da representação visual ao invés de uma sintaxe básica contra-intuitiva [1], facilitar o entendimento da estrutura booleana de uma *string*. Ou seja, para usuários que não possuem conhecimento de lógica booleana, tem-se na representação visual da *string* um formato de visualização que ele possa compreender mais facilmente. Além disso, a partir de uma única *string* o usuário será capaz de realizar a busca em diversos sistemas de busca acadêmica por meio da tradução da *string* gerada. Tendo como objetivo da

tradução da *string* definida visualmente para *strings* de outros sistemas específicos, a eliminação da necessidade do usuário aprender como definir uma *string* de busca nestes sistemas, e ter que aprender apenas como definir uma *string* de busca na linguagem proposta.

1.3 Metodologia

A metodologia para o desenvolvimento deste trabalho apresenta-se dividida em seis passos, de modo a atingir os objetivos geral e específicos definidos. Uma visão geral pode ser visualizada na Figura 1.2.

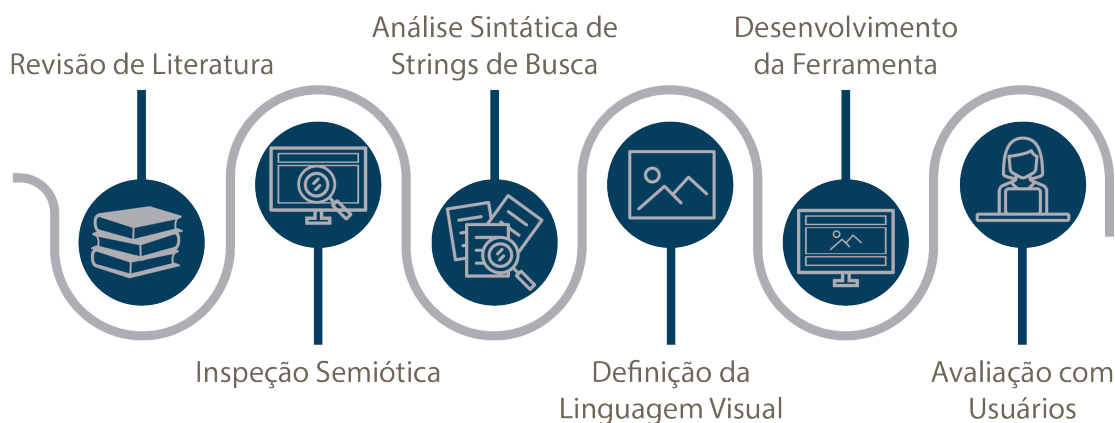


Figura 1.2. Passos da Metodologia

O primeiro passo foi a realização de uma revisão de literatura, buscando trabalhos relacionados ao tema proposto. Através do estudo dos trabalhos encontrados na literatura foi possível traçar um objetivo mais contextualizado aos problemas julgados importantes de serem tratados neste trabalho.

Sistemas de busca acadêmica de interesse para este trabalho foram investigados, selecionando dentre estes os três que juntos representavam maior diversidade nos elementos e opções disponíveis para definição da busca. Por meio do Método de Inspeção Semiótica (MIS) foi feita a inspeção das interfaces destes sistemas. A inspeção ocorreu com o objetivo de identificar quais são os elementos presentes nestas interfaces e como uma busca acadêmica é realizada. As linguagens de busca destes três sistemas foram analisadas, observando-se o modo como cada sistema trata as diferentes opções de busca.

Tendo então uma base teórica a partir da revisão de literatura e conhecendo os elementos e ações existentes em interfaces de busca acadêmica, foi definida a linguagem visual de busca acadêmica, objetivo principal deste trabalho. Além disso, o processo de definição contou com a aplicação de noções de *design* e da escola Gestalt, resultando em uma linguagem visual com elementos representativos da sintaxe de uma *string* de busca composta por combinações lógicas (OR e AND) e conceitos de agrupamento de elementos. Posteriormente à definição da linguagem visual, foi desenvolvida uma ferramenta que implementa essa linguagem. O desenvolvimento da ferramenta se deu através da programação de uma interface *web*, utilizando além das linguagens padrão da *web*, HTML5 e CSS3, a linguagem de programação JavaScript.

A linguagem visual definida e a ferramenta *web* que a implementa, foram avaliados através de testes com usuário. Foram realizados dois testes, em que o primeiro objetivou a avaliação da representação visual da *string* de busca, de forma estática. O segundo ocorreu no sentido de avaliar a interface de busca proposta como um todo, ou seja, avaliar tanto a representação visual da *string* de busca, quanto a ferramenta *web* desenvolvida para uso desta linguagem. Finalmente, os resultados obtidos por meio da avaliação foram analisados, destacando quais as contribuições este trabalho apresenta para a definição de *strings* de busca acadêmica.

Capítulo 2

Fundamentação Teórica

Uma vez que este trabalho apresenta a proposta de uma linguagem visual, o entendimento de conceitos relacionados se faz necessário. Assim, neste capítulo é realizada uma discussão acerca de linguagens visuais e também quanto ao estado da arte no que tange linguagens visuais e recuperação de informação. Tendo na Seção 2.1 a apresentação de conceitos relevantes para o entendimento de linguagens visuais no contexto deste trabalho. Uma discussão sobre trabalhos relacionados e que se aproximam deste é feita na Seção 2.2. Por fim, na Seção 2.3, o presente trabalho é inserido dentro do contexto de linguagens visuais apresentado.

2.1 Linguagem Visual

Nem sempre a comunicação humana se deu através de sequências estruturadas de palavras faladas ou escritas. Sendo que os primeiros relatos da comunicação entre os humanos são de pinturas rupestres e gesticulação. Assim, observando a história e desenvolvimento da cultura de comunicação, percebe-se que as linguagens visuais estiveram sempre presentes, seja em pinturas, mapas, projetos de arquitetura, partituras, entre outros. Com o advento da tecnologia, as linguagens visuais passaram a ser vistas sob um novo paradigma, sendo tratadas como componente chave da interação humano-computador. Linguagens visuais podem ser vistas através de multifacetadas, onde em cada uma se encontra uma forma possível de expressão visual. Contudo, para este trabalho, tem-se em foco apenas linguagens visuais empregadas para comunicação no meio digital. Sendo assim, entende-se uma linguagem visual como um conjunto de diagramas expressos como sentenças válidas nessa linguagem [29].

Nesse contexto, partindo da ideia de que um diagrama é um conjunto de símbolos, e que um conjunto de diagramas forma uma sentença, existem variadas configurações

a partir das quais tem-se a construção de uma linguagem visual. Muitas alternativas de *design* são empregadas na definição de uma linguagem visual [21], como por exemplo o uso de fluxogramas, representações *box-line*, abstração processual, iteração e sequência de execução. Comumente, as linguagens visuais são definidas para um domínio específico [21], sendo alguns deles: música (controle para sintetizadores de música), construção de interfaces (componentes pré-definidos para a construção de interfaces), processamento de imagens (ícones representando funções de processamento), visualização científica (simulações) e gráficos (edição de curvas).

Uma vez que o objetivo deste trabalho é a definição de uma linguagem visual para busca acadêmica, na Subseção 2.1.1 as abordagens para especificação de uma linguagem visual são brevemente explicadas. Linguagens visuais no contexto computacional e suas principais aplicações consideradas neste trabalho são assunto na Subseção 2.1.2.

2.1.1 Especificação

Dados os diferentes domínios para os quais podem ser definidas linguagens visuais, as técnicas para especificação se dividem em abordagens gramaticais, lógicas e algébricas [30; 29].

A abordagem gramatical baseia-se em formalismos gramaticais usados na especificação da linguagem, onde a estrutura gramatical é modificada. Em uma linguagem sequencial pode-se distinguir claramente a sintaxe e a semântica da linguagem. A sintaxe define a estrutura das sentenças, enquanto a semântica atribui um significado à sentença. De acordo com Chomsky, a gramática é baseada em um conjunto finito de sentenças observadas (o corpo da linguagem). A projeção desse conjunto de sentenças em um conjunto infinito de frases gramaticais gera o estabelecimento de leis e regras que determinam a composição dos componentes da linguagem. Dentro da abordagem gramatical, a especificação de uma linguagem visual pode partir de uma gramática de grafos, na qual a construção se baseia em arestas e nós, ou pode partir de uma gramática de conjuntos múltiplos. Outras formas de especificação são a gramática de formas e gramática visual.

Na abordagem lógica tem-se o uso da lógica matemática de primeira ordem ou outras formas de lógica matemática que geralmente derivam da inteligência artificial. Em uma abordagem lógica, o mesmo formalismo pode ser usado para especificar tanto a sintaxe como a semântica de um diagrama. Sendo a programação lógica um conjunto de regras definidas em algum predicado de primeira ordem, essas regras podem ser entendidas como a representação de uma fórmula lógica. Contudo, em uma representação visual a estruturação dos elementos se dá pelas relações espaciais entre eles. Assim,

as relações espaciais dos elementos na linguagem devem ser integrados à representação lógica. Usando as coordenadas de objetos é possível definir relações espaciais tais como interseções ou hierarquia. Um caminho inverso é integrar diretamente expressões visuais em um formalismo lógico, combinando assim as capacidades de raciocínio dos sistemas lógicos com a expressividade da comunicação gráfica.

A terceira abordagem para a especificação de uma linguagem visual é a especificação algébrica. Uma especificação algébrica consiste em funções de composição que constroem imagens complexas a partir de elementos de imagem mais simples. Nessa abordagem, a ideia é mapear o domínio em estruturas de dados abstratas, definindo funções e predicados, em que o conceito chave é a hierarquia. Portanto, os menores elementos são objetos-base tratados como estruturas de dados e as funções e predicados são usadas para combinar objetos-base em objetos-compostos de tipos mais complexos.

Dentro de cada uma dessas abordagens para especificação de linguagens visuais, existe uma variedade de sub-abordagens, que têm a base em uma mesma abordagem geral, mas diferem entre si. As abordagens gramatical, lógica e algébrica, apesar de apresentarem aspectos próprios na especificação de linguagens visuais, estão intimamente relacionadas. Logo, ao observar abordagens em aplicações reais é possível identificar conexões gerais entre elas. Em adicional, linguagens visuais são objetos de estudo em variadas áreas, como ciência da computação, psicologia e artes, sendo estudadas a partir de diferentes perspectivas e contextualizações.

2.1.2 Linguagens Visuais em Computação

Linguagens visuais no contexto computacional são empregadas para facilitar a comunicação com o usuário, estando intrinsecamente relacionadas à IHC [29]. É da natureza humana a capacidade de percepção visual do ambiente em que se está inserido, onde a capacidade de reconhecimento de padrões, objetos e formas se torna maior quanto mais estímulos são recebidos pelo cérebro. Além disso, o cérebro humano tem a capacidade de transformar pensamentos em representações visuais. Assim, tem-se como papel principal das linguagens visuais em computação o de vir a ser uma ferramenta de comunicação e interação entre humanos e computadores. Uma linguagem visual traz benefícios se sua sintaxe gráfica é compreensível e traz um retorno concomitante no lado cognitivo. Ou seja, se apesar do custo para aprender esta linguagem, a maioria dos usuários é capaz de perceber o que ela representa rapidamente e com mais profundidade do que na versão textual.

No domínio computacional, linguagens visuais abrangem uma ampla classe de entidades e aplicações que buscam a comunicação e interação visual a partir da pers-

pectiva da IHC. Pesquisas em visualização de software e animação de algoritmos têm seus esforços centrados na definição de linguagens visuais geradas por computadores, de modo que as estruturas estática e dinâmica dos algoritmos sejam descritas de forma significativa. Outras pesquisas são centradas no raciocínio diagramático, caracterizando como humanos e computadores entendem e raciocinam a partir de diagramas. Todas essas pesquisas apresentam em comum a tentativa de facilitar os processos mentais envolvidos na programação explorando a interface de usuário através de uma variedade de estímulos visuais [5; 25; 36; 17].

Linguagens de programação que apresentam alto nível de abstração e sistemas que utilizam informações visuais são descritas como Linguagens Visuais [32]. Uma linguagem visual computacional é classificada de acordo com o domínio de aplicação e o tipo de design aplicado, sendo as duas principais classificações: linguagens de programação visuais (*visual programming languages*) e sistemas de visualização (*visualization systems*) [31; 27; 24; 6; 29]. Linguagens de programação visuais utilizam diversos elementos gráficos para especificar um programa. O sistema é especificado em duas ou mais dimensões (linguagens textuais são consideradas de uma dimensão). Sistemas de visualização fornecem um modelo de interação com o sistema por meio de componentes visuais de forma a aumentar a compreensão. O programa é especificado em forma textual e os componentes visuais são usados para ilustrar algum aspecto do programa ou execução. Muitos programas de visualização de sistemas foram propostos, sendo, em sua maioria, definidos em contextos específicos.

Pode-se dizer que linguagens visuais são parte essencial na comunicação humano-computador. Quando aplicadas de forma a implementar os componentes de uma linguagem, tem-se como resultados linguagens de programação, de busca ou de um sistema específico. Quando aplicadas no sentido de fornecer uma interface interativa, observa-se como resultado sistemas de visualização e manipulação da informação. Entretanto, mesmo tratando apenas de linguagens visuais para comunicação no meio digital, os conceitos e representações válidos em uma linguagem são diferentes para outro. Assim, conclui-se que cada linguagem visual é única, devendo ser interpretada a partir do contexto e aplicabilidade para os quais foi definida.

2.2 Trabalhos Relacionados

Linguagens visuais são vistas através de multifacetadas, com uso e aplicação em diferentes áreas. Uma das faces de linguagens visuais é aquela voltada para a computação, cujo objetivo central é o de facilitar a comunicação do sistema com o usuário. Assim, serão

discutidos trabalhos que tratam de linguagens visuais computacionais e que possuem algum tipo de relação com busca e recuperação de informação, assunto deste trabalho. Para tanto, a literatura foi explorada a fim de encontrar trabalhos científicos relacionados, posteriormente divididos em três temas: busca de artigos, busca em banco de dados e representações.

No primeiro tema, tem-se trabalhos relacionados a interfaces de busca de artigos, acadêmicos ou não. Dada a importância de se realizar revisões de literatura no meio acadêmico e a diversidade de ofertas de sistemas de busca acadêmica, muitos trabalhos se dedicam a avaliação da interface desses sistemas.

A combinação de termos, de modo a se definir uma *string* ou *query* de busca, é necessária não somente em sistemas de busca (acadêmica ou geral), como também em sistemas de banco de dados. Em ambos, o objetivo ao enviar uma busca ou consulta, é o de receber dados relacionados aos termos. De modo que, os resultados serão melhores, quanto mais completa estiver a *string* ou *query*.

Outros trabalhos apresentam a proposta de uma linguagem visual para construção de *strings* de busca. Essas linguagens são construídas a partir de elementos gráficos e organização espacial dos mesmos, de modo a se obter uma representação que faça sentido no contexto em que foi proposta.

Além disso, observa-se na maioria dos trabalhos relacionados, que buscam não apenas pela definição de uma linguagem visual, como também pela construção de uma interface interativa. De modo a ter a linguagem visual como uma opção de interação durante o processo de busca em meio digital. Nas subseções a seguir, são discutidos exemplos de trabalhos dentro de cada um desses quatro temas descritos anteriormente.

2.2.1 Busca de Artigos

A busca por artigos considera tanto artigos científicos, quanto artigos de conteúdo diverso. Assim, tem-se aqui alguns trabalhos que tratam da busca de artigos, onde cada trabalho apresenta uma visão de busca e objetivos de estudo diferentes. Em [13], os autores apresentam um estudo de usabilidade com usuários, nas interfaces de busca básica (Figura 2.1) e de busca visual (Figura 2.2) do *EBSCOHost* no contexto de uma grande universidade de artes liberais. *EBSCOHost*¹ é um serviço de bases de dados utilizado por instituições em todo o mundo. O estudo ocorreu no sentido de confirmar a hipótese de que a busca visual é uma alternativa viável para a busca textual, através do entendimento sobre o comportamento de busca do usuário e os benefícios potenciais de uma abordagem visual. Portanto, buscou-se responder a seguinte questão: “até que

¹<https://www.ebscohost.com/>

ponto a interface de pesquisa básica e a interface de pesquisa visual do *EBSCOHost* suportam a pesquisa do aluno?”. Os testes ocorreram com estudantes que apresentavam algum conhecimento na realização de buscas acadêmicas, dividindo os participantes em quatro sessões de duplas e outras doze sessões individuais.

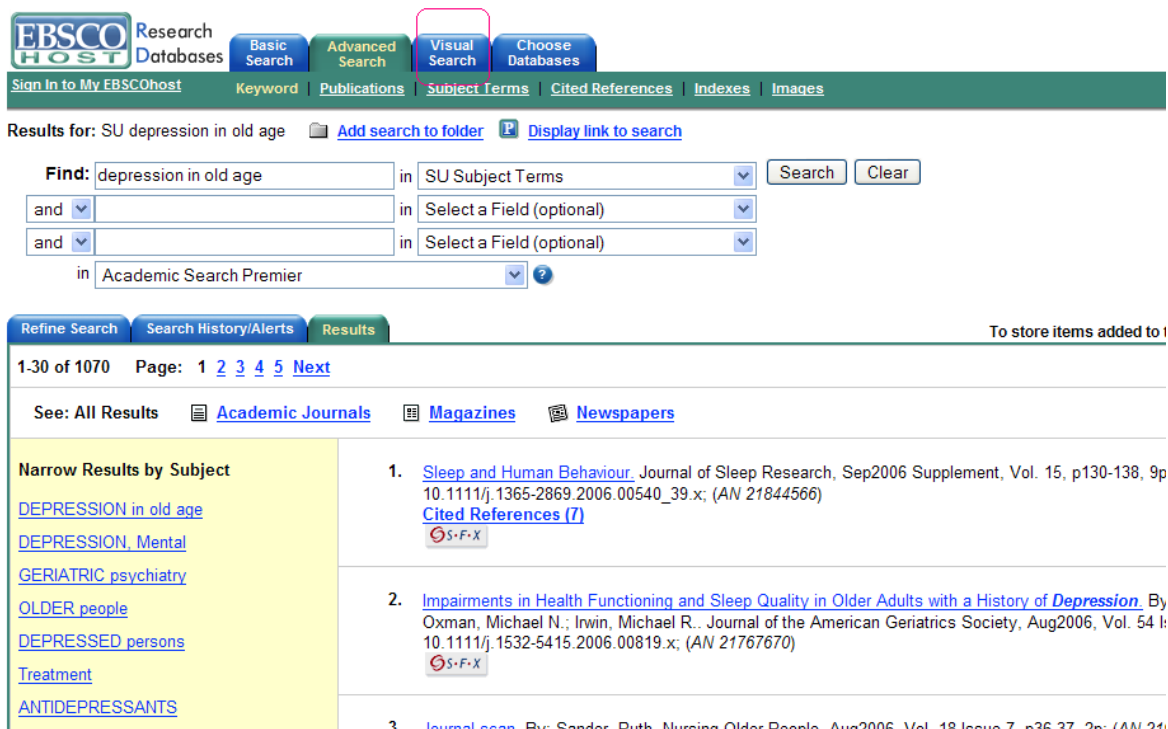


Figura 2.1. Fragmento da interface de busca básica do *EBSCOHost*

Figura extraída em março de 2017 de
<http://fiberlibrarian.blogspot.com.br/2006/09/ebscohost-visual-search.html>

Com o objetivo de descobrir como cada interface (textual e visual) suportava o processo intelectual dos participantes na elaboração de um tópico, foram definidas tarefas, partindo de uma busca inicial para um estreitamento do tópico de busca, solicitando aos participantes que realizassem buscas considerando os sub-tópicos do assunto escolhido. Em adicional, foi investigado o quão bem os usuários foram capazes de encontrar e usar os *widgets* de interface e quão satisfeitos os alunos se sentiram depois de usar as interfaces.

Como pontos fortes da busca textual os participantes indicaram uma maior possibilidade de opções de busca, como o uso de campos específicos e filtros, além de já serem familiarizados com o modelo de interface apresentado. Em contrapartida, apontaram como pontos fracos da busca textual o fato de que para restringir a busca deve-se inserir palavras-chave por si mesmo, enquanto que na busca visual os sub-tópicos são apresentados pelo sistema. Além do agrupamento dos resultados em sub-tópicos, a

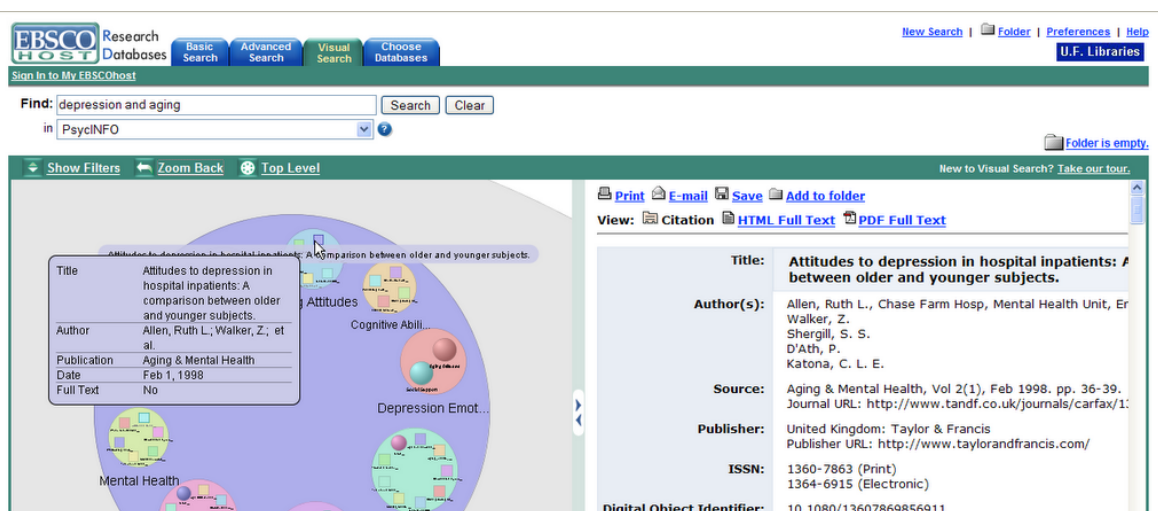


Figura 2.2. Fragmento da interface de busca visual do *EBSCOHost*

Figura extraída em março de 2017 de <http://fiberlibrarian.blogspot.com.br/2006/09/ebscost-visual-search.html>

busca visual da *EBSCOHost* fornece a vantagem de apresentar uma visão geral dos resultados, deixando mais claro os caminhos pelos quais se pode seguir. O estudo mostrou que na maior parte dos cenários avaliados, a busca visual ajudou os usuários de uma forma que a busca básica não fez. Contudo, os participantes ficaram confusos quanto ao formato em que os resultados são apresentados e alguns se sentiram inseguros quanto ao conteúdo por este ter sido um primeiro contato com uma interface a qual não estavam acostumados.

Diferentemente da interface do *EBSCOHost*, onde a busca visual é expressa no momento da apresentação dos resultados, no presente trabalho, ela é expressa no momento da definição dos tópicos de busca. Entretanto, tanto na interface visual do *EBSCOHost*, quanto na linguagem visual proposta neste trabalho, percebem-se esforços em comum, no sentido de buscar alternativas para facilitar a realização de buscas acadêmicas e obter resultados mais precisos.

Outro trabalho com contexto em interfaces de busca de artigos acadêmicos é apresentado em [15], onde é feito estudo da comunicabilidade do Portal de Periódicos da CAPES. O portal disponibiliza às instituições de ensino e pesquisa do Brasil, a produção científica nacional e internacional. O estudo investiga potenciais problemas na interface e no sistema de busca do portal que possam dificultar a interação do usuário e a busca de informação. Para a avaliação do portal foi feita a inspeção da interface com aplicação do Método de Inspeção Semiótica (MIS) e posterior avaliação com usuários usando o Método de Explicitação do Discurso Subjacente (MEDS). A avaliação e aperfeiçoamento de interfaces para recuperação de informação só é possível

mediante a compreensão de como os usuários utilizam e interagem com o sistema, identificando qual o grau de satisfação com os recursos disponíveis e se o mesmo é eficiente, útil e confiável.

Através da inspeção semiótica da interface do Portal de Periódicos da CAPES, foram identificados obstáculos na interface que impedem ou dificultam a compreensão do funcionamento do sistema. Um dos problemas apontados está no que se refere a ajuda e documentação do portal, que mediante análise se mostrou insuficiente e ineficiente, de modo que na maioria das vezes em que tem uma dúvida ou não sabe como agir, o usuário necessita aprender por tentativa e erro. Além disso, não fica claro quais são os critérios adotados pelo sistema para ordenação e apresentação dos resultados e quais estratégias de busca são mais adequadas. Foram identificadas também incoerências na apresentação de alguns elementos da interface e informações duplicadas com a disponibilidade de três modos diferentes na busca avançada. Como forma de corrigir esses problemas encontrados, sugere-se então o uso mais expressivo de *tooltips*, além de uma documentação mais elaborada e completa, e também a simplificação das informações e campos de dados na interface, de modo a evitar duplicidade de conteúdo.

Para o estudo com usuários, foram selecionados 12 alunos de doutorado das áreas de Ciências Humanas, Linguística, Letras e Artes, Ciências Biológicas, Ciências Exatas e também três bibliotecários. Foram considerados oito fatores que afetam o processo de busca: contexto, usuário, consulta, base de dados, sistema de busca, processo de busca, resultados da busca e interface. Durante os testes, foi solicitado a cada usuário a realização de três tarefas de busca, seguidas de perguntas referentes à interface, ferramentas de busca, construção da consulta, qualidade dos resultados, estratégias de busca e mudanças que ocorrem no processo de busca de informação. Constatou-se a necessidade de que os sistemas de recuperação de informação suportem a interação de usuários com diferentes capacidades e demandas. Ou seja, que esses sistemas apresentem alternativas simples e claramente definidas para usuários novatos, e alternativas de busca avançada para usuários mais experientes. A partir desse estudo, evidenciou-se que o modo como o sistema suporta as estratégias pessoais do usuário para condução de uma busca, exerce influência direta no sucesso e satisfação do usuário, assim como os recursos que o sistema oferece.

Ao apontar as necessidades de aprimoramento na interface do Portal de Periódicos da CAPES, Gomes et al. trazem a luz a importância em reconhecer as necessidades dos usuários em sistemas de recuperação da informação para que os recursos disponíveis sejam vistos como ferramentas de auxílio e facilitação do processo de busca. Assim, apesar de não tratar de uma interface visual para buscas ou alternativas para a busca textual, o estudo apresentado contribui não apenas para a definição da linguagem

visual apresentada neste trabalho, como também para a definição dos objetivos que se pretendem alcançar. Destacando-se como resultado deste trabalho o desenvolvimento de uma interface mais simples e coesa, onde se busca uma padronização dos elementos apresentados de modo que uma a partir da definição de uma *string* ela possa ser traduzida para outros sistemas.

Ainda no contexto de sistemas para busca de artigos, tem-se na Wikipédia uma ferramenta comumente utilizada, principalmente para a busca de artigos de conhecimento geral. Um *redesign* da interface de busca da Wikipédia é proposta por Sandhu & Liu. O estudo avalia características de interface de pesquisa interativa (ex.: caixa de pesquisa, controle de recursos, navegação e layout). Os autores projetaram duas propostas de interface em momentos distintos da pesquisa, realizando testes com potenciais usuários, mostrando as diferentes propostas de interface. Em um momento inicial da pesquisa foi projetada uma interface, tomando por base informações acerca das necessidades e preferências de informação dos usuários identificadas em estudos relacionados existentes na literatura. Foram adicionados à interface recursos de interesse para o estudo, como preenchimento automático, refinamento da consulta, categorização e visualização flexível de resultados. Partindo da proposta inicial, foram selecionados oito estudantes, desde iniciantes no meio acadêmico à doutorandos, aos quais foi solicitada a realização de duas tarefas de busca. Foi possível então identificar a preferência e opinião dos usuários quanto aos recursos interativos propostos na interface, a partir de onde o estudo prosseguiu a fim de propor um *redesign* mais completo e efetivo da interface de busca da Wikipédia.

Na segunda interface projetada, características da primeira interface que receberam um *feedback* positivo foram mantidas e as que receberam *feedback* negativos foram abandonadas, além disso, características sugeridas pelos usuários foram consideradas e algumas foram adicionadas à nova interface. Nesta interface, ao inserir um termo de busca, é permitido ao usuário o refinamento da busca, que ocorre através da apresentação de categorias relacionadas a busca, conforme apresentado na Figura 2.3. A página de resultados é apresentada em formato de lista, imagem ou palavras-chave, com as categorias relacionadas dispostas à esquerda, de modo que o usuário possa limitar os resultados apresentados a qualquer momento. Quando os resultados são listados, é exibido um título clicável e uma parte do conteúdo logo abaixo. A apresentação dos resultados em imagens também apresenta um título clicável e parte do conteúdo, além de uma imagem *thumbnail*, que pode ser clicada a fim de exibir mais detalhes. E por fim, quando os resultados são organizados em palavras-chave, a mesma configuração apresentada para refinamento da busca é apresentada, com a adição de palavras-chave relacionadas a cada uma das categorias.

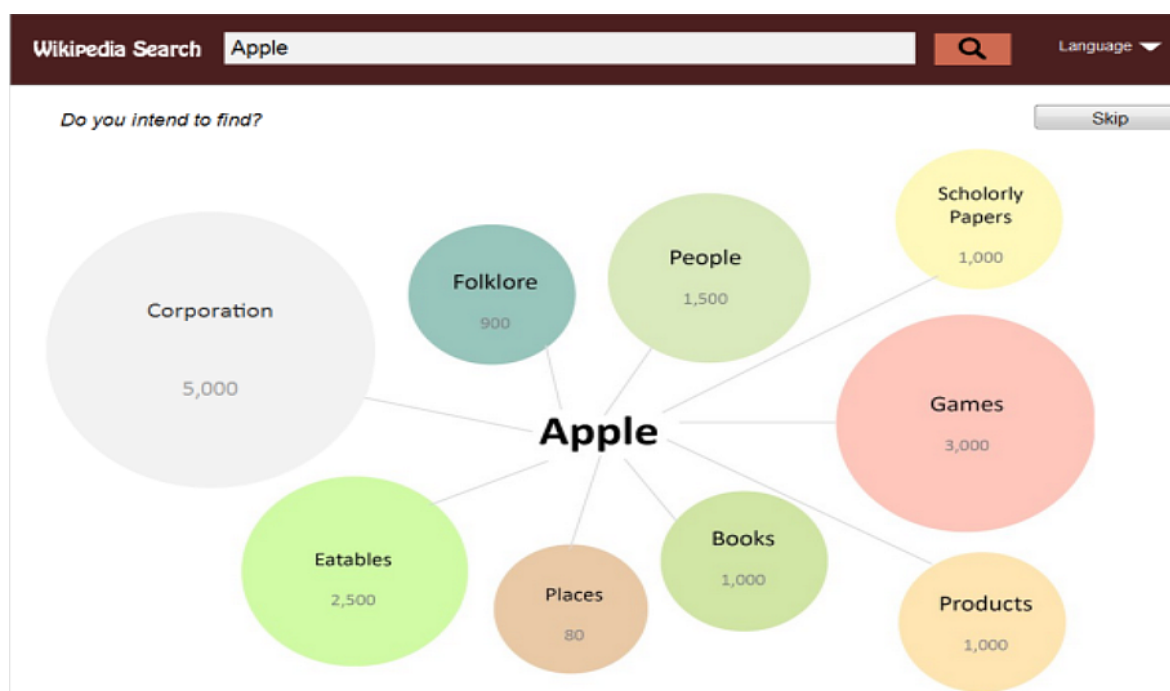


Figura 2.3. Exemplo da proposta de *redesign* da Wikipédia

Figura extraída de [34]

Apesar de categorias geradas automaticamente com base na frequência de cliques serem populares, neste estudo, os usuários demonstraram preferência por categorias criadas de acordo com esquemas de hierarquia tradicional, normalmente caracterizados como sistematização e generalidade. Quanto aos diferentes formatos de exibição dos resultados, a maioria dos usuários consideraram úteis e que podem ajudar a explorar melhor os resultados, contudo, alguns apontaram que a visualização de metadados apenas é útil para usuários avançados e que querem analisar tudo que está disponível para determinada categoria.

A aplicabilidade de interfaces interativas vai de encontro a proposta de se utilizar de meios visuais para melhorar o processo de busca como um todo. Assim, neste trabalho e no estudo apresentado, a interface interativa é vista como uma alternativa à interface padrão de busca, onde é fornecido ao usuário elementos de interação que facilitem o processo de busca e em consequência gerem resultados mais expressivos para o usuário. Outro ponto a destacar se refere à investigação da percepção do usuário mediante características consideradas de interesse em interfaces interativas, de onde se pode chegar a conclusão de que os usuários, em um primeiro momento, preferem interfaces mais simples e diretas, partindo para a necessidade de maior controle da busca quando se tornam mais especialistas no uso do sistema.

2.2.2 Busca em Banco de Dados

Considerando banco de dados como um conjunto de informações relacionadas entre si, é natural que em algum momento se deseje buscar por informações e relacionamentos específicos dentro deste conjunto. Assim, existe uma variedade de modelos e estruturação de consultas, de modo que algumas são representações visuais.

O conceito de *linked data* é relativo a dados ligados entre si, com a função de publicar e estruturar dados na *web*. Uma grande porção de *linked data* está disponível no formato RDF (*Resource Description Framework*) e pode ser consultado usando a linguagem de consulta padronizada SPARQL. *QueryVOWL* [20; 19] é uma linguagem de consulta visual baseada em grafos, definida para mapeamentos SPARQL. Realiza um mapeamento para SPARQL com o uso de elementos gráficos de VOWL (notação visual para a ontologia OWL - *Web Ontology Language*). Além de prover um conjunto de elementos visuais para representar conceitos e componentes, contém elementos de funcionalidade interativa para auxiliar na criação de consultas SPARQL.

QueryVOWL foi definida como uma linguagem de consulta visual que é intuitiva e fácil de usar, mantendo-se flexível e preservando a maior parte da expressividade do SPARQL. A proposta é expressar critérios definidos pelo usuário para pesquisar gráficos RDF específicos em *linked data*. A ideia da linguagem é modelar visualmente um grafo parcial do que se presume existir no conjunto de dados, além de possibilitar a inclusão de restrições. Quando um grafo *QueryVOWL* é aplicado a um determinado conjunto de dados RDF, todos os subgrafos do conjunto de dados que correspondem à consulta são recuperados, permitindo que todo o grafo seja explorado de forma dinâmica. Em contraste, no modelo SPARQL puro, os conjuntos de dados recuperados são explicitados em listas ou tabelas.

O conceito de *QueryVOWL* se concentra na seleção de um elemento (um nó de classe, um nó literal ou um rótulo de propriedade) escolhido pelo usuário como termo de busca. Na Figura 2.4, é ilustrada uma consulta visual em *QueryVOWL*. No contexto da consulta, o usuário deseja obter como resultados dados sobre duas pessoas que estrearam filmes juntos com pelo menos um de seus filhos.

A avaliação da linguagem definida por Haag et al. se deu através da implementação de dois protótipos e posterior condução de um estudo qualitativo com usuários. O primeiro protótipo foi uma aplicação web, que implementa os principais elementos da linguagem de consulta visual e fornece uma oportunidade para experimentar a aparência de uma implementação *QueryVOWL* interativa. O outro protótipo foi baseado em uma aplicação para *desktop*, implementando todos os elementos da linguagem, mas apresentando um grau limitado de interatividade. Ambos os protótipos foram

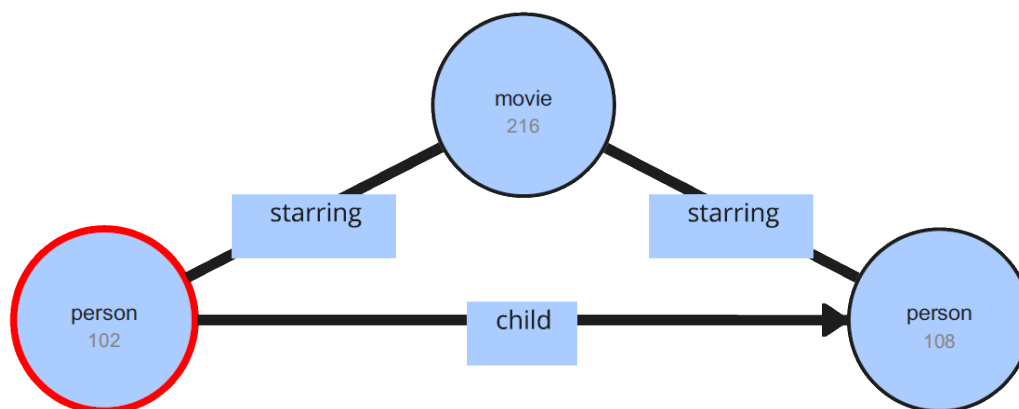


Figura 2.4. Exemplo de consulta em *QueryVOWL*

Figura extraída de [20]

implementados seguindo conceitos de manipulação de objetos de arrastar e soltar, e dimensionar nós, além de gerar a consulta automaticamente. O estudo qualitativo se deu com a seleção de oito participantes e a definição de oito tarefas para definição de consultas na linguagem. Contudo, apenas a implementação *web* foi testada. Uma vez que a consulta desejada foi definida na tarefa, alguns participantes adaptaram sua consulta inicial para chegar à solução correta. A partir do estudo, observou-se que usuários leigos poderiam lidar bem com a estrutura de consulta básica, excluindo alguns aspectos mais específicos da visualização que não estavam imediatamente claros.

Assim como a linguagem visual proposta neste trabalho para definição de *strings* de busca acadêmica, *QueryVOWL* é uma alternativa a busca padrão existente dentro de um contexto, onde se usam elementos visuais para facilitar a interação do usuário com o sistema de recuperação da informação. Ressalta-se também que apesar da diferença dos contextos de aplicação, neste trabalho a condução da proposta da linguagem e avaliação ocorreu de forma semelhante, onde se partiu de uma adaptação de um modelo existente para um modelo visual, e se implementou uma ferramenta *web* seguindo princípios semelhantes de manipulação de objetos.

Em outro exemplo de busca em banco de dados, o domínio de aplicação é o de dados XML, onde Choi & Wong apresentam a linguagem de consulta visual *VXQ*. A proposta da linguagem é que usuários com pouco conhecimento sobre XML e sua linguagem de consulta, *XQuery*, sejam capazes de gerar consultas por meio de abstrações intuitivas de ambos. As consultas formuladas pelo usuário são reescritas para seu equivalente semântico, de modo que as consultas reescritas tenham melhor desempenho. A motivação para a definição da linguagem *VXQ* se deu posteriormente a observação de que formular uma consulta *XQuery* às vezes pode ser uma tarefa não trivial, e

exige certo nível de esforço, inclusive para um usuário com conhecimento sobre XML. Assim, o objetivo de *VXQ* é o de disponibilizar uma alternativa para construção de consultas *XQuery* em uma interface amigável, sem expor ao usuário a complexidade da linguagem.

VXQ apresenta um formato de consulta interativa, em que uma consulta é formulada a partir do desenho de grafos e diagramas, e seleção de opções no contexto da linguagem *XQuery*, excluindo a necessidade do usuário lembrar a sintaxe completa da linguagem base. Além disso o sistema verifica os diagramas sendo definidos, limitando as opções que o usuário pode selecionar quando necessário, diminuindo assim as chances de definição de buscas que não gerarão resultados.

No sistema que implementa a linguagem *VXQ*, a tela é dividida em dois espaços distintos, conforme pode ser visualizado na Figura 2.5. Na esquerda, é representado um grafo fonte com o documento, permitindo aos usuários explorar e selecionar partes dos documentos que desejam manter. Do lado direito, é apresentado o painel de consulta, no qual os usuários visualizam a consulta e seu projeto estrutural. A geração de uma consulta *XQuery* ocorre inicialmente com a exploração do esquema em grafo do documento XML, seguida da especificação da estrutura de saída e restrições. Sendo que tanto no grafo representativo do documento XML, quanto no grafo representativo da consulta, a estrutura se inicia a partir de um nó raiz. A criação de nós no grafo que definem a consulta pode ocorrer ao arrastar um nó do grafo fonte, pela expansão de um nó-pai ou pela reestruturação da consulta criando novos elementos.

Na comparação da linguagem visual *VXQ* com outras linguagens no mesmo contexto, foi observado que *VXQ* é mais expressiva que as propostas anteriores. Em adicional, *VXQ* suporta a reescrita de consultas, otimizando consultas para processadores de consulta subjacentes e apresenta as consultas reescritas visualmente para edição posterior. Assim, tem-se em *VXQ*, não apenas uma linguagem para consultas, como também um sistema para otimizar consultas ainda no momento da definição. Comparando-se ao trabalho discutido neste documento, pode-se destacar a motivação de criar uma alternativa para a definição de consultas, onde o uso de representações visuais tende a promover a definição de uma linguagem mais fácil de entender e consequentemente, de usar.

Uma visão mais voltada para interfaces interativas na busca de dados relacionados é apresentado por Zhang & Marchionini, onde é apresentado o *redesign* do navegador e interface de busca relacional *RB* (Figura 2.6). A proposta é fornecer uma ferramenta para busca e navegação em grandes coleções de informações, onde é interessante ao usuário a possibilidade de ver rapidamente as partições da coleção e quantos itens estão disponíveis em diferentes partições. Para gerar a nova versão do navegador, chamada

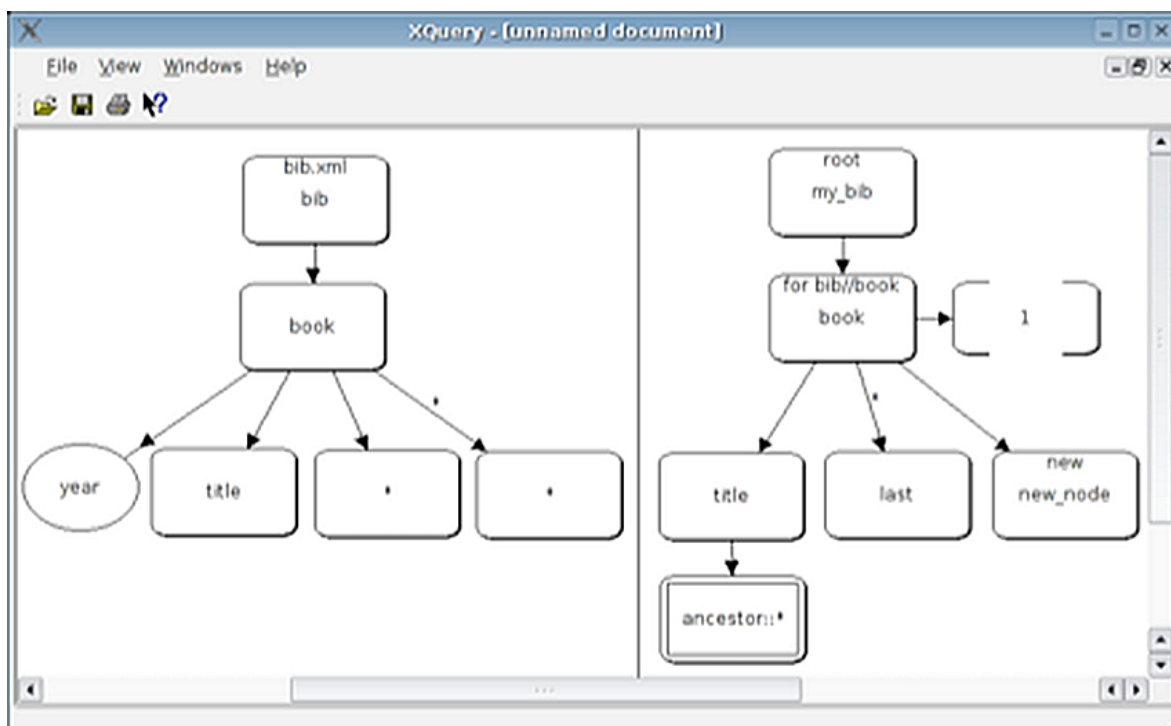


Figura 2.5. Exemplo do sistema que implementa *VXQ*

Figura extraída de [7]

RB++ e apresentada na Figura 2.7, é realizada a comparação com outras interfaces. Sendo que a comparação é feita a partir de estudos com usuários e aplicação de mais de uma dúzia de diferentes instâncias de bancos de dados, com posterior definição do modelo de interação do usuário.

Categorias facetadas ajudam no entendimento da composição de uma coleção de informações, permitindo que os usuários façam diferentes cortes na informação, podendo visualizá-la através de diferentes perspectivas. Nesse contexto se encontra o navegador de busca relacional *RB*, onde os diferentes valores de faceta são empregados como objetos de consulta para uma pesquisa de modo que todo o espaço possa ser particionado rapidamente com movimentos simples do mouse. Dentre as melhorias realizadas, na nova interface são exibidas visualmente na mesma tela mais de duas facetadas. Além disso, a interface de *RB++* permite mais flexibilidade para explorar relacionamentos e adiciona uma função de filtragem dinâmica para o conjunto de resultados.

Os testes com usuário foram aplicados com a participação de 17 estudantes de graduação e pós-graduação que tinham familiaridade com a realização de buscas. O objetivo foi o de comparar a eficácia, a eficiência e a satisfação do usuário na realização de tarefas usando a interface tradicional *RB* e usando a interface refatorada *RB++*.

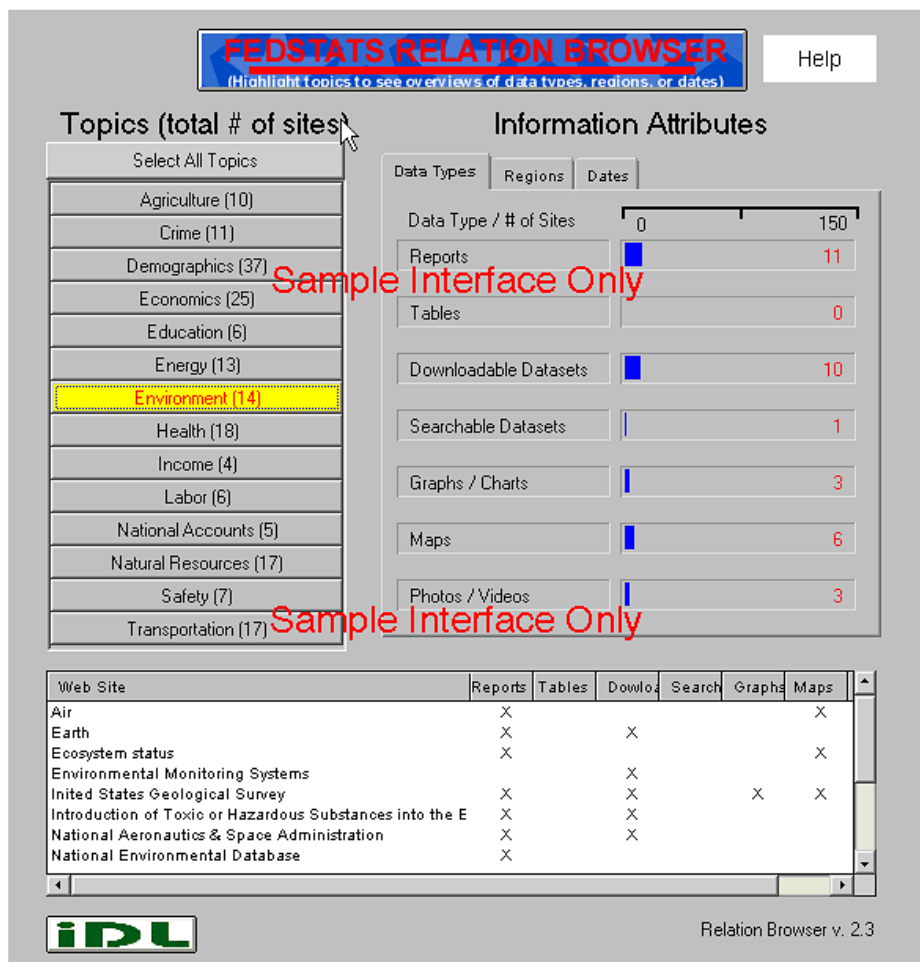


Figura 2.6. Fragmento da interface de busca do navegador *RB*

Figura extraída em março de 2017 de
<https://ils.unc.edu/relationbrowser/index.php?page=history>

O estudo também ocorreu no sentido de explorar a interface *RB++*, procurando identificar novos padrões de interação ocasionados pelo *redesign* da interface original. O estudo demonstra a eficácia da nova interface para tarefas de busca e navegação em comparação com a dificuldade de execução da mesma busca na sintaxe SQL. Principalmente para pesquisas mais complexas, os participantes consideraram a interface *RB++* mais fácil de usar, em especial pela exibição visual dos itens e a busca dinâmica de palavras-chave.

RB++ provê aos usuários um espaço para visualizar um *overview* de categorias de informação, filtros dinâmicos, exploração dos resultados e pesquisa de funções. Quando comparada à interface desenvolvida a partir da linguagem visual de busca do presente trabalho, destaca-se mais uma vez o uso de recursos visuais para facilitar o processo de busca, deixando de lado a necessidade do usuário compreender estruturas sintáticas

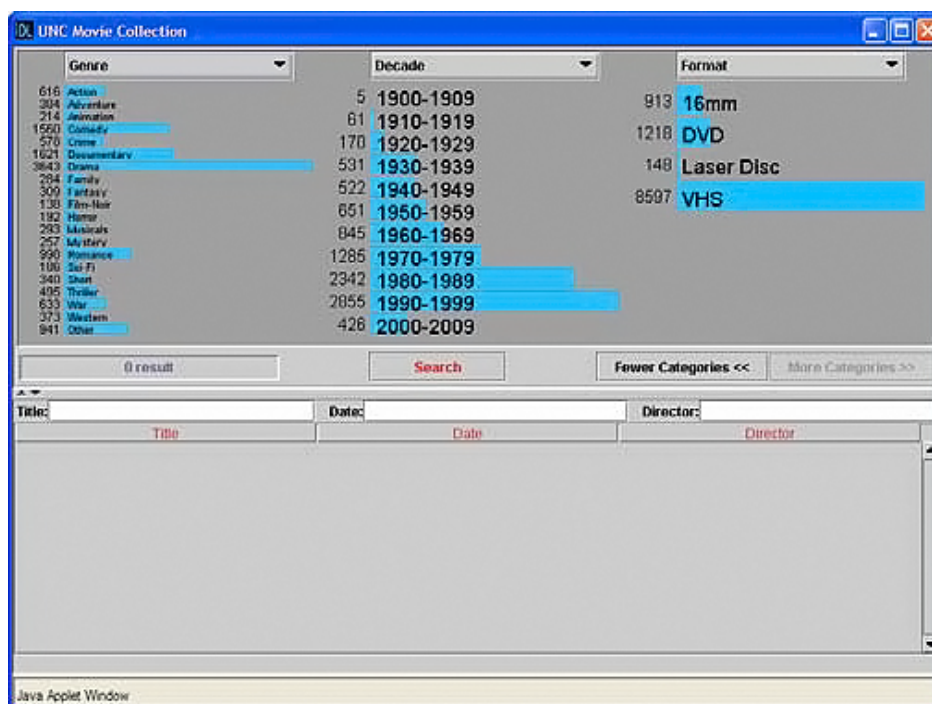


Figura 2.7. Fragmento da interface de busca do navegador *RB++*

Figura extraída em março de 2017 de
<https://ils.unc.edu/relationbrowser/index.php?page=history>

complexas.

2.2.3 Representações Visuais

Enquanto os outros trabalhos apresentados possuem um foco maior em criar alternativas de representação, tratando mais de interfaces interativas e visualização dos resultados, outros trabalhos falam diretamente da construção de uma linguagem visual. Sendo discutidos aspectos referentes ao processo de definição da linguagem, o significado dos elementos e suas abstrações, bem como exemplos dentro de um contexto de uso.

No trabalho de Spoerri, é descrita a representação *InfoCrystal*, que pode ser utilizada como uma ferramenta de visualização ou como uma linguagem de consulta visual. É fornecido ao usuário meios gráficos para especificação de consultas booleanas, em que se pode selecionar as relações de interesse e não se faz necessário o uso de operadores lógicos e parênteses. O objetivo da representação é fornecer aos usuários ferramentas de recuperação visual que permitam explorar, manipular e relacionar grandes espaços de informação aos seus interesses de forma interativa.

Na representação, o espaço de informação pode ser explorado ao longo de várias dimensões, onde os usuários podem criar abstrações através da manipulação da

informação. Durante o processo de busca, é enviado aos usuários um *feedback* visual dinâmico, por meio do qual se pode observar os efeitos de uma consulta e saber como proceder. Além disso, *InfoCrystal* possibilita a formulação de forma gráfica e flexível de consultas.

Para representar uma consulta booleana, os pesquisadores aplicaram inicialmente o conceito do diagrama de Venn para visualizar as relações dos conjuntos de informação através da interseção de formas geométricas que representam cada conjunto. Depois, procurando representar cada possibilidade dessa relação, o diagrama de Venn foi explodido em subconjuntos disjuntos, apresentado na Figura 2.8. Os subconjuntos receberam uma representação gráfica, onde o formato reflete o número de critérios satisfeitos pelo conteúdo que ele representa dentro da informação consultada. Por fim, todos os subconjunto foram adicionados dentro de uma borda, onde foram representados os conjuntos originais.

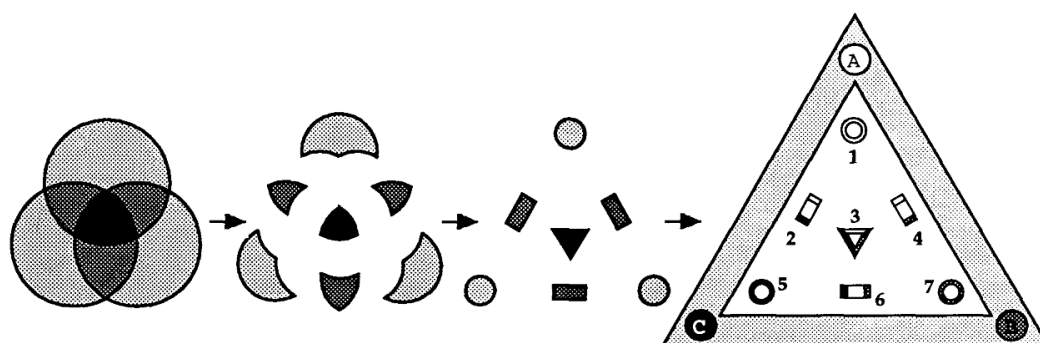


Figura 2.8. Construção da representação *InfoCrystal*

Figura extraída de [35]

A ideia da representação é que a visualização enfatize a informação qualitativa ou quantitativa associada. Assim, cada elemento utilizado na representação possui um significado atrelado. Como é o caso do formato de codificação, que indica o número de critérios aos quais os conteúdos associados satisfazem. Na Figura 2.9 é exibida a representação de uma consulta em *InfoCrystal*. A consulta contém os termos: *human factors*, (*visual* OR *graphical*), *query language*, e *information retrieval*. O elemento central indica quantos documentos recuperados na busca satisfazem todos os critérios. Assim, caminhando nos demais ícones representados, o usuário pode acessar os documentos resultantes que satisfazem o conjunto de critérios mais próximo.

O problema que se busca resolver com a *InfoCrystal* é semelhante ao problema proposto neste trabalho. Ambos procuram aumentar a capacidade de acesso de informação dos usuários, propondo uma forma para visualizar elementos abstratos e

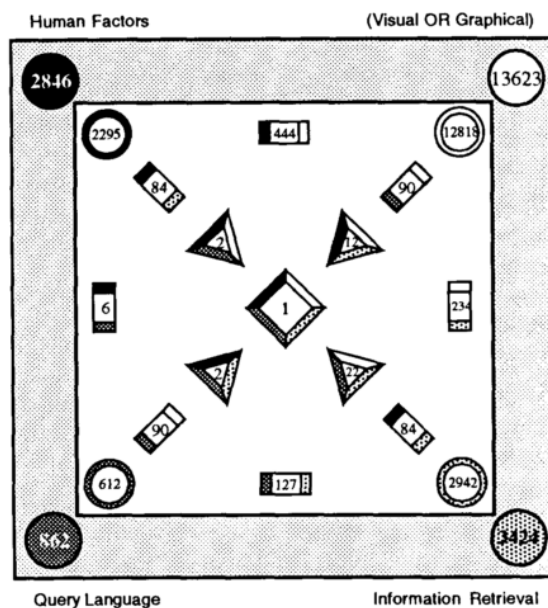


Figura 2.9. Representação de uma *string* em *InfoCrystal*

Figura extraída de [35]

formular consultas. Entretanto, em *InfoCrystal* a consulta é tratada de um modo mais complexo, combinando a representação da *string* com a apresentação dos resultados. Diferentemente de *InfoCrystal* que não apresenta um domínio de aplicação específico, a linguagem visual proposta neste trabalho é específica para o contexto de buscas acadêmicas, permitindo que problemas pertinentes a este domínio sejam tratados com maior atenção.

VISUAL [3] é uma linguagem visual de consulta baseada em elementos gráficos e definida no contexto de bases de dados científicas. O corpo de uma consulta em *VISUAL* é composto por objetos iconizados, uma caixa de condições (aritméticas ou expressões), definições de consulta interna e referências às consultas externa ou interna. As bases de dados científicas nas quais a linguagem é aplicada são caracterizadas por apresentar relações espaciais, com sequências e objetos complexos, onde as consultas são de natureza exploratória. Os objetos de consulta gráfica tentam imitar as práticas dos pesquisadores para representar relacionamentos semânticos, como hierarquia e relações espaciais.

Por ser uma linguagem orientada a objetos, *VISUAL* trata uma consulta como a implementação de um objeto, onde objetos interagem uns com os outros para o processamento da consulta. Os componentes de um objeto podem ser elementos atômicos, outros objetos ou coleções (conjuntos e sequências). Assim, um objeto de consulta é composto por outros objetos de consulta, permitindo que a consulta seja modula-

rizada, de modo que os objetos especificados e submetidos ao sistema sejam salvos e reutilizados.

Uma consulta em *VISUAL* é composta por um conjunto de janelas organizadas hierarquicamente, em que uma janela se refere a zero ou mais objetos de consulta externos. Cada objeto de consulta é composto de um cabeçalho e um objeto de corpo da consulta. No cabeçalho são descritos o nome da consulta e os parâmetros de entrada e saída. Quando duas janelas têm o mesmo nome, o resultado da consulta é a união dos resultados de ambas. O corpo da consulta consiste em objetos gráficos, condições e definições da consulta. As condições são expressões aritméticas ou definidas pelo usuário. A definição da consulta pode ser expressa através de operadores de associação \in , \notin ou de comparação \supset , \supseteq , \equiv , \subset , \subseteq . Nesta linguagem, um objeto gráfico representa o modelo e classe de dados, podendo ser domínio, método, intervalo, ou região de imposição espacial do objeto, em que cada modelo apresenta propriedades únicas (cor, forma) definidas pela especificação da consulta. Na Figura 2.10 é apresentado um exemplo de consulta na linguagem *VISUAL*. Nesse exemplo, tem-se dois objetos de consulta, *PARTICLE-SEQUENCE* e *PARTICLES-IN-WINDOW*, onde o primeiro localiza os trechos resultantes da consulta definida no segundo objeto. A consulta ilustrada é sobre o uso de um método específico do domínio “singleEvolves”, partindo de uma partícula *p* para outra partícula *P*.

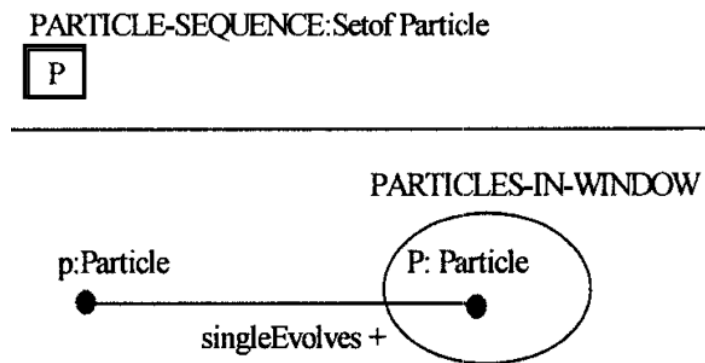


Figura 2.10. Representação de uma consulta em VISUAL

Figura extraída de [3]

A representação feita na linguagem *VISUAL* requer um entendimento maior do contexto da busca, uma vez que são esses conceitos que regem como será feita a formação da representação. Comparando com o trabalho apresentado aqui, encontram-se semelhanças no sentido de que ambos tratam de linguagens especificamente para a definição da consulta, não apresentando aspectos relacionados à apresentação dos re-

sultados.

2.3 Contextualização

No decorrer deste capítulo foram discutidos aspectos relacionados à linguagens visuais e também trabalhos disponíveis na literatura relacionados ao tema. Dada a discussão apresentada, considera-se então as linguagens visuais como sendo a organização de elementos gráficos (formas, linhas, cores) para expressar algo dentro de um domínio e contexto específicos. Onde a expressão pode ser no sentido de definir a abstração de uma linguagem ou servir como representações de conceitos já definidos.

A especificação de linguagens visuais pode ocorrer levando em consideração três diferentes abordagens. Tendo na abordagem gramatical uma visão mais próxima da linguagem sequencial, onde são definidos com clareza a sintaxe e a semântica da linguagem. Na abordagem lógica, o formalismo lógico rege a especificação da linguagem, onde as regras provenientes deste formalismo são aplicadas na composição da representação visual. Por fim, na abordagem algébrica, tem-se como conceito-chave a hierarquia entre os elementos, onde parte-se da definição de um elemento atômico para a construção de elementos compostos.

Quando se observa a aplicação de linguagens visuais dentro da computação, dois aspectos diferentes são observados. Um em que as linguagens visuais são tratadas como linguagens de programação, de busca, ou similares, aplicadas para a definição e construção de sistemas, consultas ou outras finalidades. No segundo aspecto, observam-se linguagens visuais onde o objetivo principal é o de possibilitar a visualização do sistema através de elementos gráficos, ou seja, possibilitar, por exemplo, que um conjunto de dados seja visualizado em uma abstração diferente da textual, podendo ser apresentado em um gráfico interativo.

Assim, considerando-se a faceta de linguagens visuais para comunicação no meio digital, trabalhos relacionados foram identificados na literatura. Destacando-se nas diferentes classificações dos estudos encontrados (busca de artigos, consulta em banco de dados e representações visuais), a existência de multifacetadas dentro da faceta computacional de linguagens visuais. Entretanto, a maioria destes trabalhos é anterior à situação atual, onde a Internet exerce papel fundamental na comunicação humana. Levando-se a considerar não que o uso de linguagens visuais não é uma questão de pesquisa já fechada, mas uma questão que necessita ser revista e investigada quanto as circunstâncias atuais.

Finalmente, ao considerar os conceitos definidos e aplicações de linguagens vi-

suais, tem-se no presente trabalho a proposta de uma linguagem visual, cujo domínio de aplicação se encontra no processo de definição de buscas acadêmicas. Onde a linguagem visual proposta é especificada a partir de uma abordagem lógica, em que os formalismos lógicos da construção de uma *string* de busca acadêmica exercem influência direta na definição dos significados dos elementos na linguagem. Em adicional, uma vez que linguagens visuais são ubíquas às interações humanas, é vantajoso que elas sejam aplicadas de modo a facilitar ou gerar alternativas à comunicação entre humanos e sistemas computacionais.

Capítulo 3

Inspeção Semiótica de Interfaces de Busca Acadêmica

Interfaces de busca acadêmica são aquelas que oferecem opções específicas para a definição de consultas para recuperação de artigos e outros documentos acadêmicos. Considerando-se estas interfaces, tem-se na *string* de busca um fator determinante para a obtenção de resultados adequados à necessidade de informação do usuário. A construção da *string* de busca é um processo cíclico, em que inicialmente o usuário define uma busca e a refina até se sentir satisfeito ou desistir da busca.

Existe uma grande variedade de sites e bibliotecas digitais para a realização de buscas acadêmicas disponíveis na Internet. Dentre elas, encontram-se sistemas voltados para uma área do conhecimento única (ex.: Ciência da Computação, Ciências Humanas), e sistemas que retornam documentos em qualquer uma dessas áreas. Em razão disso, observa-se que estes sistemas apresentam características próprias, onde há diferenças na forma como são realizadas as buscas padrão e avançada. Assim, uma vez que este trabalho propõe uma linguagem visual para a realização de buscas acadêmicas, faz-se pertinente a investigação das interfaces de busca acadêmica existentes e como ocorre a construção de uma *string* de buscas.

Para realizar esta investigação optou-se pelo uso do Método de Inspeção Semiótica (MIS), uma vez que este permite uma análise científica a partir da inspeção de interfaces. Na Seção 3.1 é descrito o Método de Inspeção Semiótica, e em seguida como ele foi aplicado nesta pesquisa. Sendo que na Seção 3.2 é apresentado o processo de preparação e aplicação do MIS. A triangulação das inspeções dos três sistemas e os resultados obtidos a partir daí são assunto na Seção 3.3.

3.1 Método de Inspeção Semiótica (MIS)

A investigação de interfaces é parte dos estudos em IHC, que compreende métodos de avaliação próprios para a realização da avaliação de interfaces dentre os quais se destaca o MIS, aplicado neste trabalho. O MIS é um método embasado na teoria da Engenharia Semiótica, que por sua vez é uma teoria da IHC.

Semiótica é o estudo dos signos, processos de significação e como signos e significação tomam parte na comunicação [9]. Signos são elementos da interface, cuja expressão e conteúdo são estabelecidos por processos de significação e, através dos quais são formadas mensagens por meio de processos de comunicação. A Engenharia Semiótica apresenta uma visão teórica abrangente de IHC, permitindo “entender os fenômenos envolvidos no design, uso e avaliação de um sistema interativo” [33].

Os principais focos da Engenharia Semiótica são os aspectos básicos do *design* e o uso de tecnologias da informação. Tecnologias interativas objetivam, não somente habilitar e apoiar a interação, como também transmitir aos usuários do sistema a visão de comunicação do projetista sobre a quem o sistema se destina, que problemas podem ser resolvidos e como interagir com o sistema [9]. A interface de um sistema e seus padrões de interação são caracterizados por metamensagens transmitidas do projetista ao usuário sob uma perspectiva de metacomunicação [12].

A metamensagem é composta por signos, classificados como metalinguísticos, estáticos e dinâmicos [10]. Os signos metalinguísticos estão presentes na documentação, comunicando diretamente ao usuário sobre o sistema, bem como apresentando aspectos relevantes sobre o mesmo. Os signos estáticos expressam o estado do sistema, interpretados independentemente das relações temporais e causais. Os signos dinâmicos expressam o comportamento do sistema quando há interação do usuário, sendo ligados aos aspectos temporais e causais da interface, ou seja, à própria interação.

No MIS é realizada a avaliação da metacomunicação do projetista para o usuário, objetivando a reconstrução da metamensagem [33; 10]. A avaliação por meio do MIS é composta de cinco etapas, indicadas na Figura 3.1, antecedidas pelo processo de preparação para aplicação do método. O processo de preparação é composto pela definição do objetivo da aplicação e geração de um cenário. Durante a aplicação do método nos passos 1, 2 e 3, o avaliador reconstrói a metamensagem do projetista com base nos signos inspecionados. No passo 4, é definida uma única metamensagem, a partir do constate das metamensagens geradas nos passos anteriores. Por fim, no passo 5 o avaliador aprecia a qualidade da metacomunicação.

A aplicação do MIS pode ser realizada em duas perspectivas diferentes: técnica e científica [10; 11]. Na perspectiva técnica, o avaliador busca encontrar rupturas e

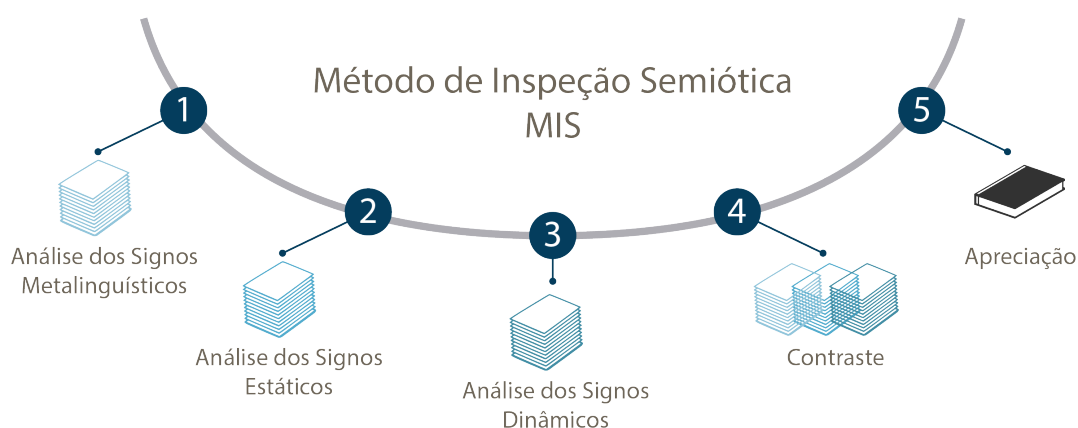


Figura 3.1. Passos de execução do MIS

problemas na transmissão da metamensagem do projetista para o usuário, de modo a fornecer material para melhorias. Na aplicação científica, procura-se a obtenção de novos conhecimentos ou hipóteses científicas que sirvam de embasamento para a pesquisa, por meio das características dos signos do sistema. Contudo, o conhecimento gerado na aplicação do MIS científico deve ser validado e, para tal, se utiliza o método de triangulação. A triangulação dos resultados pode ser feita a partir da comparação com os resultados das avaliações de outros sistemas sob o mesmo contexto (fontes endógenas) ou, resultados de avaliações de sistemas de domínios diferentes mas relacionados entre si (fontes exógenas). O objetivo da triangulação é promover a diversidade e expor descobertas à contradição. Neste trabalho, o MIS foi aplicado sob uma perspectiva científica, com o intuito de identificar elementos e características presentes em interfaces de busca acadêmica.

3.2 Aplicação do MIS

Inicialmente, dado o objetivo de inspecionar interfaces de busca acadêmica através da aplicação científica do MIS, foram selecionadas interfaces de busca para inspeção. Uma vez que o presente estudo e os pesquisadores envolvidos fazem parte da área de Ciência da Computação, optou-se por se inspecionar sistemas de busca acadêmica nesta área. Assim, procurou-se dentre os sistemas disponíveis, aqueles que apresentavam juntos uma maior diversidade de elementos para definição e realização da busca. Os sistemas selecionados são listados na Tabela 3.1. Foram definidas duas categorias de sistemas de busca acadêmica, onde uma contém aquelas cujo conteúdo é publicado pelo indexador (bibliotecas digitais da ACM e da IEEE), e a outra contém conteúdo publicado por

outros meios (site de busca Google Scholar).

Tabela 3.1. Sistemas de busca acadêmica

Nome	Endereço web
ACM	http://dl.acm.org/
IEEE	http://ieeexplore.ieee.org/
Google Scholar	https://scholar.google.com/

A preparação para aplicação do método consiste não apenas da seleção dos sistemas a serem inspecionados, mas também da definição de uma questão de pesquisa, inspeção informal dos sistemas, definição de escopo e foco, e por fim a descrição de um cenário de aplicação. Assim, os passos de preparação são apresentados a seguir:

Questão de pesquisa: Investigar os tipos de signos relevantes para a busca acadêmica e como são apresentados ao usuário. Além disso, verificar quais ações estão disponíveis para interação do usuário e como ocorre a definição da *string* de busca.

Inspeção informal: As interfaces de busca acadêmica selecionadas estão disponíveis online de forma gratuita e seu uso é expressivo no meio acadêmico de Ciências da Computação. Todas as interfaces apresentam um campo específico para busca, onde é possível ao usuário realizar uma busca rápida, incluindo ou não termos de busca avançada. Estas interfaces também fornecem ao usuário a opção de realizar uma busca avançada, apresentando para isso as opções implementadas para tal busca, como limite de ano, nome de autor, entre outros.

Escopo e foco da avaliação: Serão avaliadas as principais telas de cada interface, com foco na busca padrão e na busca avançada disponibilizadas ao usuário.

Cenário: “Danny acabou de entrar no mestrado em Ciência da Computação e para iniciar sua pesquisa ela decidiu executar uma revisão sistemática da literatura sobre seu tema de estudo. Após entender o método a ser utilizado para a revisão sistemática, ela selecionou várias bibliotecas de busca acadêmica que seriam interessantes. Após navegar em algumas interfaces, ela percebeu que mesmo entrando os mesmos termos nos diferentes sistemas, eles pareciam se comportar de maneira diferente. Além disso, ela percebeu que existia a possibilidade de realizar uma busca avançada, inserindo limitações ou termos específicos. Assim, Danny resolveu testar cada uma das interfaces de busca que encontrou, formulando a melhor *string* de busca para cada uma, para obter os melhores resultados possíveis.”

Posteriormente, os passos de aplicação do MIS foram aplicados a fim de inspecionar as três interfaces selecionadas. Assim, para cada uma das interfaces foi realizada a reconstrução da metamensagem do projetista para os signos metalinguísticos, estáticos

e dinâmicos, e em sequência feito o contraste dessas metamensagens e reconstrução da metamensagem do sistema como um todo. Nas subseções a seguir são discutidas, em separado, as metamensagens resultantes da análise de cada um desses sistemas de busca acadêmica, reconstruídas a partir do seguinte modelo: “*aqui está a minha (do designer) compreensão de quem é você (usuários), o que eu aprendi que você quer ou precisa fazer, em que formas prefere, e por quê. Este é o sistema que eu, portanto, projetei para você, e esta é a maneira que você pode ou deve usá-lo, a fim de cumprir uma série de propósitos que se enquadram nesta visão*” [12].

3.2.1 Metamensagem da Interface da Biblioteca Digital da ACM

Quem é você: Você é um pesquisador da área de Ciências da Computação e está interessado em pesquisar tópicos da área de computação em uma biblioteca digital na qual possa confiar, entendendo o nível de confiança na biblioteca pela definição de termos de uso, política de privacidade, código de ética e contato com os desenvolvedores. Além disso, é provável que você seja um membro da *Association for Computing Machinery*, ou gostaria de ser em algum momento, podendo assim acessar todos os recursos disponibilizados. Você tem interesse em ficar informado a respeito de novidades da ACM e eventos na área. Provavelmente você tem um tempo limitado e quer realizar sua busca de forma rápida, seja ela simples ou avançada, tendo a disposição tarefas que otimizem o processo de busca, como filtros de busca.

O que você quer ou precisa fazer: Você quer uma biblioteca digital para pesquisa, descoberta e uma plataforma de *networking*, que contenha coleções de texto completas de todas as publicações da ACM, incluindo revistas, anais de conferências, revistas técnicas, boletins e livros. Uma coleção de curadoria e publicações *hosted full-text* a partir de editores selecionados. Acesso ao *ACM Guide to Computing Literature*, uma base de dados bibliográfica abrangente focada exclusivamente em computação. Um conjunto ricamente interligado de conexões entre autores, obras, instituições e comunidades especializadas. Você quer acessar essa biblioteca em uma interface para dispositivos móveis.

Você quer que a biblioteca digital tenha uma tela inicial, onde possa visualizar as informações sobre o conteúdo que é fornecido, anúncios de eventos ou cursos relativos à biblioteca, bem como um modo de acesso rápido à tópicos recentes. Você quer um modo de busca avançada, com links para filtros de busca, onde possa buscar em publicações da ACM, em grupos de interesse especial (SIG), em conferências, em coleções especiais ou em conteúdo hospedado. Você também quer refinar sua busca por tipo de literatura

e buscar por publicador, ou por toda a base. Você precisa de uma barra de buscas e um botão para enviá-la, localizada no canto superior direito da página, para inserir seus termos de busca, e quer que esta barra esteja disponível em todas as telas.

Você também quer que a biblioteca digital tenha uma tela contendo apenas o espaço onde você vai inserir sua busca, sem precisar visualizar outras informações. Você quer poder realizar uma nova busca, editar ou salvar a busca atual, ou realizar uma busca avançada. Você quer poder decidir se seus resultados serão buscados nas duas bases da ACM e saber a quantidade de resultados para cada decisão que fizer. Quando realiza uma busca você quer saber qual base de dados foi consultada e quantos documentos essa base de dados possui. E você também quer poder escolher se quer expandir a busca para a outra base de dados disponibilizada pela ACM. Você quer exportar seus resultados para diferentes formatos (bibtex, endnote, acmref, csv) para facilitar o gerenciamento de referências.

Sobre a sua busca, você quer saber quantos resultados foram obtidos, exibindo-os em várias páginas. Além disso, você quer poder ordenar seus resultados, seja por mais relevante, mais atual, contagem de citações ou número de downloads (últimas 6 semanas, 12 semanas ou total). Você quer refinar sua busca, seja por pessoa (instituições, autores), publicação, conferência ou ano. Você quer visualizar em um gráfico qual a incidência de resultados em cada ano. Você quer visualizar informações básicas sobre cada resultado, como título, autores, ano, onde foi publicado, quantidades de downloads, link para versão PDF do trabalho, parte limitada do resumo e palavras-chave. Você também quer saber quais são as próximas conferências cujos trabalhos serão publicados pela ACM.

Você quer que sejam exibidas no topo da página as opções de nova busca, editar/salvar, busca avançada e exportação de resultados, e também as informações sobre base de buscas. Na lateral da página você quer que sejam exibidas opções de refinamento da busca e abaixo informações sobre as próximas conferências. No centro da página você quer que sejam exibidos os resultados da busca, com as referidas informações básicas de cada resultado. Tanto no topo, como no fim da página, você quer que seja exibida a quantidade de resultados e controle de navegação de páginas de resultados.

Na busca avançada, você quer manipular seus termos de busca, seja editando, salvando ou inserindo novos termos e limitações à sua busca. Você quer selecionar em qual base de dados da ACM deseja realizar sua pesquisa, clicando na opção referente. Você quer escolher onde procurar seus termos dentro do documento (ex.: qualquer campo, título, resumo) e definir se os termos devem ser idênticos, próximos ou não existirem no trabalho pesquisado. Você precisa que a sintaxe da *query* seja exibida de acordo

com os filtros selecionados, sendo atualizada sempre que você realiza alguma alteração. Você quer que a sintaxe da *query* mostre separadamente os termos que possuem filtros diferentes, indicando quando determinado termo é pesquisado no resumo ou em outro lugar específico; quer também que seja indicado se um termo deve obrigatoriamente aparecer (representado por “+”) ou quando um termo não deve aparecer (representado por “-”).

Formas que você pode ou deve utilizar o sistema: Você pode acessar a biblioteca digital da ACM por meio do navegador web de sua preferência. Em dispositivos móveis, você pode optar por acessar a interface web, ou uma interface específica para dispositivos móveis, devendo se cadastrar para obter uma conta da ACM DL, instalar o aplicativo e depois entrar no aplicativo com a sua conta. Você pode acessar a biblioteca digital ACM como membro da ACM, membro da SIG ou em modo público, sendo que os recursos acessíveis por você dependerá do seu nível como usuário. Apenas membros da ACM podem ter acesso online para o CACM (*Communications of the ACM*). Você pode escolher se inscrever como um indivíduo ou em nome de uma biblioteca, universidade ou empresa.

Você pode navegar na tela inicial, por onde pode acessar tópicos recentes, ou realizar buscas avançadas. Na barra de buscas, você deve digitar os termos que deseja buscar e depois pode clicar no botão “Search” ou apertar a tecla **Enter** para enviar sua busca. Para expandir sua busca da base completa de texto da ACM, para incluir também resultados do ACM Guide, você deve clicar sobre a opção, e caso deseje limitar a busca, deve clicar no mesmo local, onde agora a opção será a de limitar. Você pode escolher em qual formato exportar seus resultados, entre uma das opções disponíveis. Quando você passar o mouse em cima de algum link, o texto é realçado e para selecioná-lo você deve clicar sobre ele. Depois de ter feito uma busca, você pode escolher fazer uma nova busca, para isso, você pode retornar à página inicial ou clicar no link “new search”. Clicando em “edit/save query” ou em “advanced search” você será encaminhado para uma página, onde você pode manipular seus termos de busca de modo avançado.

Para navegar entre as páginas de resultado, você pode selecionar o número da página que deseja, sendo exibidos a página atual e links diretos para mais nove, ou pode avançar. Para refinar sua busca, você pode escolher uma das opções disponíveis e clicar sobre ela; ao clicar, a opção é expandida e são exibidos os elementos de refinamento, os quais você pode escolher um e clicar sobre ele. Para definir alguma opção, como modo de ordenação dos resultados, onde procurar no documento e outros, basta clicar na caixa correspondente e clicar na opção desejada no menu suspenso que irá aparecer. Na busca avançada, você tem opções para selecionar a base de dados e onde procurar nos documentos. Ao clicar no link “show query syntax”, a sintaxe da sua busca é exibida

de modo que você pode editar ou apenas visualizar termos e filtros usados; você pode esconder essa visualização clicando no link “hide query syntax”. Você pode desfazer a edição na sintaxe da *query* clicando em “reset query”, retornando ao estado anterior.

Você pode inserir mais termos de busca, com uma filtragem de documentos diferentes da anterior, clicando no botão “+” logo em frente ao espaço para digitar os termos; ou pode excluir uma combinação de termos e filtros clicando no botão “-”. Para limpar sua busca, você pode clicar em “clear” ou em “reset query” para retornar ao estado anterior da *query*. Para incluir uma expressão como termo de busca, ela deve vir entre aspas, caso contrário, as palavras serão consideradas em separado. Você pode digitar termos com as mesmas restrições no mesmo campo ou cada termo em um campo separado, mas quando finalizar, a busca será atualizada e aqueles campos que possuem as mesmas restrições serão agrupados. Se você quer agrupar termos, de modo que a busca seja feita pela interseção dos dois grupos, você deve editar a sintaxe da *query* inserindo um “AND”, caso contrário, todos os termos serão considerados um único grupo de termos. Você pode separar os termos de um mesmo grupo por vírgulas ou apenas espaços em branco, que o resultado obtido será igual.

Evidências: a inspeção semiótica da interface da biblioteca digital da ACM considerou os signos metalinguísticos, estáticos e dinâmicos da interface. Entretanto, por razões da quantidade de evidências referentes aos pontos observados e comentados a partir da metamensagem, apenas alguns serão apresentados nas Figuras 3.2, 3.3, 3.4 e 3.5 .

- The ACM Digital Library** is a research, discovery and networking platform containing:
- **The Full-Text Collection** of all ACM publications, including journals, conference proceedings, technical magazines, newsletters and books.
 - A collection of curated and **hosted full-text** publications from select publishers.
 - **The ACM Guide to Computing Literature**, a comprehensive bibliographic database focused exclusively on the field of computing.
 - A richly interlinked set of **connections** among authors, works, institutions, and specialized communities.

Figura 3.2. Fragmento 1 da interface da ACM DL

3.2.2 Metamensagem da Interface da Biblioteca Digital da IEEE

Quem é você: Você está interessado em pesquisar conteúdo científico ou técnico das áreas de engenharia elétrica, ciência da computação ou eletrônica, em uma biblioteca

s.cfm?query=visual%20language&filtered=&within=owners.owner=HOSTED&dte=&bfr=&srt=_score

ACM DL DIGITAL LIBRARY

Searched for *visual language* [new search] [edit/save query] [advanced search]

Searched The ACM Full-Text Collection: 432,449 records [Expand your search to The ACM Guide to Computing Literature: 2,500,820 records] ?

112,527 results found Export Results: bibtex | endnote | acmref | csv

Refine by People
Names
Naur, Peter (167)
Wexelblat, Richard L. (98)
McKinley, Kathryn S. (93)
Lindsey, C. H. (90)
Graham, Susan L. (87)
Taylor, Dave (85)
Noble, James (84)
Ma, Kwan-Liu (83)
Shneiderman, Ben (83)

Result 1 - 20 of 112,527 Result page: 1 2 3 4 5 6 7 8 9 10 >>

Sort by: relevance
relevance
publication date
citation count
downloads (6 Weeks)
downloads (12 months)
downloads (overall)

1 [Increasing XML interoperability in Visual Rewriting Systems](#)
Tiago Lopes Telecken, José Valdeni de Lima
December 2005 WebMedia '05: Proceedings of the 11th Brazilian Symposium on Multim
Publisher: ACM
Bibliometrics: Citation Count: 0
Downloads (6 Weeks): 0, Downloads (12 Months): 1, Downloads (Overall): 198
Full text available: PDF
To the data exchange mechanism of current visual language specification systems, different languages are used

Figura 3.3. Fragmento 2 da interface da ACM DL

Advanced Search

Select items from The ACM Guide to Computing Literature ?

Where Any field matches all of the following words or phrases: "visual language" - +

Where Title matches any of the following words or phrases: query - +

Where Abstract matches any of the following words or phrases: search - +

Where Any field matches none of the following words or phrases: string - +

SEARCH [clear]

[sign in required to save query] [hide query syntax]

Edit Query Query syntax is generated automatically; editing below will override this, to revert back, [Reset Query](#)

(+"visual language" -string) AND acmdlTitle:(query) AND recordAbstract:(search)

View Full Query Syntax

```
"query": { ("visual language" -string) AND acmdlTitle:(query) AND recordAbstract:(search) }
"filter": {owners.owner=GUIDE}
```

Figura 3.4. Fragmento 3 da interface da ACM DL

digital que forneça diferentes tipos de inserção da busca, na qual possa confiar, entendendo o nível de confiança na biblioteca pelo contato com os desenvolvedores, definição de termos de uso, política de privacidade, e também informações sobre a organização responsável. Você pode ser um membro da IEEE, ou estar acessando o conteúdo da biblioteca em sua universidade, ou ainda comprou conteúdo específico. Você pode ter pouco ou muito conhecimento na definição de *strings* de busca, ou pouco ou muito tempo para a definição da busca, de modo que deseje variados tipos de busca, podendo escolher qual usar de acordo com suas necessidades.

O que você quer ou precisa fazer: Você quer uma biblioteca digital para pesquisa em uma ampla quantidade e variedade de conteúdo, nas áreas de engenharia

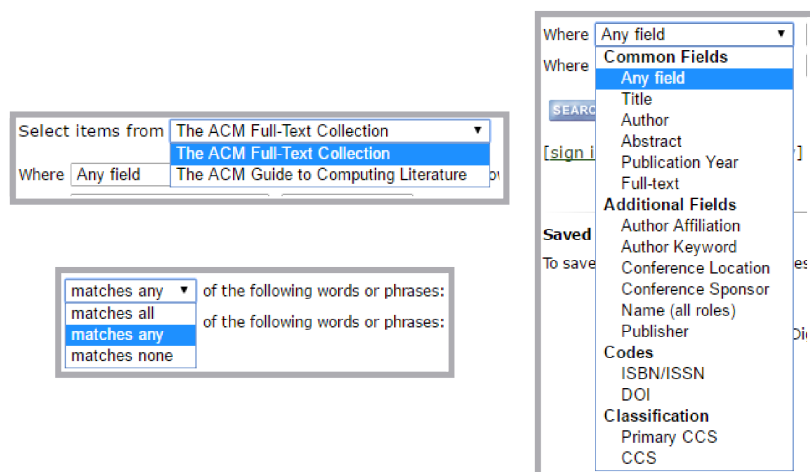


Figura 3.5. Fragmento 4 da interface da ACM DL

elétrica, ciência da computação e eletrônica. Você quer diversos modos de navegação na biblioteca, com conteúdo agrupado por tipo (livros, publicações de conferências, cursos, jornais e revistas, ou em toda a base da IEEE), por tópicos e palavras-chave, listas com os termos mais buscados, além da navegação padrão.

Você precisa que a barra de busca padrão esteja localizada no topo de todas as páginas, e que em frente ao campo para inserção dos termos tenha um botão para envio da busca. Você precisa que no topo da interface seja exibida a barra de buscas, e logo abaixo, informações sobre a quantidade de resultados recuperados e opções de ordenação e agrupamento de resultados em páginas. As opções de refinamento da busca, você deseja que sejam exibidas na lateral esquerda, sendo exibido no centro os resultados e informações básicas sobre cada um. Ao fim da página, você quer que esteja disponível o controle de navegação de páginas. Quer realizar buscas básicas, por nome de autor ou publicação; além de opções de busca avançada, por comando, por citação, alertas de busca e histórico. Você precisa que o modo e tipo de busca selecionados seja mostrado em destaque em relação às outras opções disponíveis.

Sobre a sua busca, você quer saber quantos resultados foram obtidos para sua *string* de busca. Sendo que deseja que os resultados sejam exibidos em uma quantidade definida por você em diversas páginas e escolher o modo de ordenação (relevância, mais novos primeiro, mais antigos primeiro, mais citados em artigos, mais citados em patentes, ordem alfabética crescente, ordem alfabética decrescente). Para cada documento recuperado que atendem aos seus critérios de pesquisa, você quer que sejam exibidas informações básicas: título, autores, ano, onde foi publicado, ano, volume, páginas, DOI, quantidade de citações, editor, resumo, versão disponível online e para download em PDF, copyright, além de uma cadeado indicando se o documento está

disponível para livre acesso ou se você deve comprá-lo. Contudo, para conseguir visualizar mais resultados em um espaço, você prefere que o resumo de cada documento esteja escondido, podendo ser expandido quando você desejar. Você quer que o controle de navegação de páginas apareça no fim da página. E também quer, depois de realizar uma busca, poder editá-la e receber sugestões de novas buscas. Além disso, você precisa de um modo para exportar as referências (plain text, BibTex, RefWords, RIS, CSV) dos seus resultados, exportando para todos os documentos ou apenas para os que você selecionou.

Você quer refinar sua busca e seus resultados, limitando por tipo de publicação, ano, autor, afiliação, título da publicação, editor, local da conferência, nível e tipo. E opções pré-definidas para estruturar sua busca. Os filtros, exibidos na lateral esquerda, estão agrupados por tipo, e cada tipo pode ser escondido ou expandido conforme você queira.

Na busca avançada, você quer diferentes formas de inserir sua busca: palavras-chave, comandos ou citação. Na busca por palavras-chave, você quer facilmente inserir termos em quantos campos quiser, definindo para cada campo onde procurar no documento e o operador lógico para agrupamento com o termo seguinte. Além disso, você quer uma lista com opções de refinamento pré-definidas, as quais você pode aplicar facilmente à sua busca. Você precisa, que ao definir uma busca, na página de resultados, sua *string* seja apresentada, exibindo a hierarquia de execução da busca, onde os termos processados em primeiro são exibidos nos parênteses mais internos. Quando você define um limite de anos, seja por intervalo ou um ano específico, você deseja que esse(s) ano(s) sejam exibidos no resultado da busca. Você também deseja um modo de busca por citação, no qual você pode inserir o número DOI, título ou outras informações de referência do documento desejado.

Você quer a opção de realizar pesquisas por comandos, que produzam resultados mais segmentados, combinando os termos da sua busca com operados lógicos e parênteses, e que exista um menu listando os campos de dados e operadores disponíveis. Um caractere curinga, para utilizar em buscas com termos incompletos ou buscas com uso da raiz de uma palavra. Você também quer que a busca tenha um critério de processamento, sendo este da esquerda para a direita e que a ordem de precedência seja determinada pelo uso de parênteses, OR e AND. A partir da entrada de uma palavra, você quer que a busca inclua variações dela, incluindo plural, singular e outros, e também em versões com algum erro gramatical como a troca das letras **s** e **z**.

Formas que você pode ou deve utilizar o sistema: Você pode acessar a biblioteca digital da IEEE por meio do navegador web de sua preferência. Se você ainda não tem uma conta IEEE, você pode se registrar, podendo acessar os vários sites

da IEEE com essa conta, gerenciar sua conta e participar de discussões. Você pode navegar no conteúdo agrupado por tipo (livros, publicações de conferências, cursos, jornais e revistas, ou em toda a base da IEEE). A navegação também pode ser guiada por tópicos, com 16 categorias pré-definidas, por palavras-chaves inseridas após seleção de um tópico, ou ainda através de listas com os termos mais buscados.

Na busca global, você pode buscar por uma ou mais palavras-chave, (básica, autor, publicação) clicando no botão respectivo abaixo da barra de buscas e preenchendo os campos específicos de cada tipo. Tendo realizado uma busca, você quer inserir mais termos na caixa de texto para refinar os resultados, na lateral esquerda da sua página de resultados. Para cada documento recuperado, você pode visualizar informações básicas de cada resultado, podendo clicar ao lado do título do documento para selecioná-lo, e pode visualizar o resumo clicando em seu respectivo botão “Abstract”, devendo clicar de novo para esconder o resumo. Para obter o documento completo, você deve clicar no link para download do PDF ou versão online, podendo fazer o download se estiver habilitado, ou sendo solicitado a comprar o documento para acesso.

Ao clicar na opção para download de citações, você pode escolher o formato de saída (plain text, BibTex, RefWords, RIS, CSV) e se deseja incluir o resumo na referência. Você pode escolher se quer que sejam exibidos todos ou apenas os de acesso livre, definir quantos documentos serão exibidos por página e definir o modo de ordenação (relevância, mais novos primeiro, mais antigos primeiro, mais citados em artigos, mais citados em patentes, ordem alfabética crescente, ordem alfabética decrescente), navegando pelas páginas de resultado, ou ir diretamente para a primeira ou última páginas. Você pode refinar seus resultados inserindo novas palavras na busca, selecionando um tipo de conteúdo pré-definido, limitando o ano, ou por outros modos, como autor, afiliação, título da publicação, editor, local da conferência, nível e tipo. No filtro de ano, você pode definir um intervalo, seja por manipulação direta ou inserindo os anos; pode também entrar com um único ano. Quando você seleciona um filtro, deve clicar no botão “Apply Refinements” para que o filtro seja aplicado à sua busca.

Diferente da busca básica, onde você só pode filtrar os resultados depois da pesquisa, na busca avançada, você pode preencher critérios para uma pesquisa avançada. Na busca avançada, você pode escolher entre três modos de inserir sua busca. Na busca por palavras-chave, você deve preencher quantos campos quiser com seus termos, selecionando onde este termo será buscado nos documentos (ex.: metadados, título, resumo) e definindo os operadores lógicos – AND, OR e NOT – que agrupam os termos. Você pode aplicar filtros à sua busca, selecionando os que deseja em uma lista de filtros pré-definidos, estruturando a busca para um conjunto de editores, tipo de conteúdo e ano de publicação, marcando as caixas apropriadas na área “Publisher”. Você pode

definir se deseja exibir todos os resultados para aquela busca, ou apenas os resultados cujo documento você terá acesso ao texto completo, clicando na opção referente ao que deseja. Termos inseridos no mesmo campo no modo de busca avançada, são todos considerados na busca, de modo que os documentos resultantes contêm todos eles; para considerar um termo ou outro, você deve inserir manualmente o operador OR entre as palavras. Termos inseridos em diferentes campos, são considerados em separado, de modo que a busca processa as combinações dentro de cada campo, para depois realizar o agrupamento dos campos na ordem em que aparecem.

Se você quer inserir uma expressão como termo de busca, ela deve vir entre aspas duplas “**expression**”, caso contrário, cada palavra será considerada em separado. Palavras e expressões na sua busca podem ser combinadas com o uso de operadores, sendo estes: AND, OR, NOT, NEAR, ONEAR. O operador lógico NEAR deve vir seguido de um número e de uma termo de busca, onde o número indica a quantidade de vezes que esse termo deve aparecer no documento recuperado. Você não pode usar parênteses para agrupar termos combinados com o operador OR em um campo de dados, mas pode usar parênteses para agrupar strings de busca. Para excluir um termo do resultado, você deve inserir o operador NOT. Ainda na busca avançada, você pode ordenar os campos de termos clicando no botão em frente ao campo que deseja que fique acima, e pode também incluir ou excluir um campo.

Sinais de pontuação, *stop words* e caracteres especiais (exceto &, + e /) são ignorados na busca, de modo que a busca por “solid-state”, retorna resultados com ocorrências de: “solid-state”, “solid state” e “solid_state”. O asterisco (*) é tratado como um caractere curinga e pode ser usado para indicar termos incompletos ou termos cuja raiz se repete em mais de uma palavra. Na busca por comando, elementos agrupados por parêntese têm precedência sobre os outros, da mesma forma que o operador OR tem precedência sobre AND. Além disso, a busca é processada da esquerda para a direita. O resultado é buscado em outras versões da palavra, incluindo plural, singular e outros, e também em versões com algum erro gramatical como a troca das letras **s** e **z**.

Na busca por comando, você pode definir se a busca será realizada apenas nos metadados ou se levará em consideração todo o texto e metadados, podendo selecionar comandos pré-definidos em um menu *drop-down*, ou apenas digitando seus termos de busca, campos de dados e operadores. A busca deve seguir o padrão “**field-name**:**term**”. Os campos de dados (*data fields*) disponíveis para definição da *string* de busca são: abstract, accession number, article number, author affiliation, author keywords, authors, document title, DOI, full text & metadata, IEEE terms, index terms, INSPEC controlled terms, INSPEC non-controlled terms, ISBN, issue, meta-

data, MeSH terms, parent publication number, publication number, publication title, standard number, standard dictionary terms, topic.

Evidências: conforme o método aplicado na inspeção da interface da biblioteca digital da IEEE, foram inspecionados signos metalinguísticos, estáticos e dinâmicos. Contudo, como foram obtidas muitas evidências dos aspectos da interfaces discutidos na metamensagem, apenas alguns serão apresentados nas Figuras 3.6, 3.7, 3.8, 3.9 e 3.10 .

About IEEE *Xplore*® Digital Library

The IEEE *Xplore* digital library is a powerful resource for discovery of and access to scientific and technical content published by the IEEE (Institute of Electrical and Electronics Engineers) and its publishing partners.

IEEE *Xplore* provides web access to more than four-million full-text documents from some of the world's most highly cited publications in electrical engineering, computer science and electronics. More than two-million documents are in robust, dynamic HTML format.

The content in IEEE *Xplore* comprises:

- 170+ journals
- 1,400+ conference proceedings
- 5,100+ technical standards
- Approximately 2,000 eBooks
- 400+ educational courses

Approximately 20,000 new documents are added to IEEE *Xplore* each month.

Access to full-text documents in IEEE *Xplore* can be obtained from:

- A broad range of institutional subscription options
- Exclusive subscriptions available to IEEE members and IEEE society members
- Online purchase of individual documents (at discounted prices for IEEE members)

Figura 3.6. Fragmento 1 da interface da IEEE

3.2.3 Metamensagem da Interface do Site de Busca Google Scholar

Quem é você: Você está interessado em pesquisar tópicos (artigos acadêmicos ou documentos jurídicos) de qualquer área do conhecimento, sendo que você quer apenas inserir termos em campos específicos. Você quer pesquisar em um site de busca científica no qual possa confiar, entendendo o nível de confiança no site pela definição de termos de uso, política de privacidade e informações sobre o desenvolvedor. Provavelmente você pesquisa muito sobre o mesmo assunto e quer saber quando há novas publicações.

The screenshot displays the IEEE search results page for the query "visual language". The search bar at the top contains the text "visual language" and a search button. Below the search bar, there are tabs for "Basic Search", "Author Search", and "Publication Search", along with links for "Advanced Search" and "Other Search Options". The results section shows "Displaying results 1-25 of 474 for 'visual language'". A "Sort By" dropdown menu is open, showing options: Relevance (selected), Relevance, Newest First, Oldest First, Most Cited [By Papers], Most Cited [By Patents], Publication Title A-Z, and Publication Title Z-A. On the left, there is a "Refine results by" sidebar with filters for Content Type, Year, and Author. The main results list includes two entries: "Visual language task in EEG source localization MRI" and "Visual language as a mean of communication in the field of information technology".

Figura 3.7. Fragmento 2 da interface da IEEE

The screenshot shows the "Advanced Search Options" section of the IEEE interface. It features three tabs: "Advanced Keyword/Phrases" (selected), "Command Search", and "Citation Search", along with a "Preferences" link. Below the tabs, there is a heading "ENTER KEYWORDS OR PHRASES, SELECT FIELDS, AND SELECT OPERATORS" and a note: "Note: Refresh page to reflect updated preferences." The search options are set to "Search : Metadata Only" (selected) and "Full Text & Metadata". The search criteria are entered in a structured format: "visual language" in Metadata Only, followed by "OR string" in Metadata Only, and "OR search" in Metadata Only. At the bottom, there are buttons for "Add New Line", "Reset All", and a large orange "SEARCH" button.

Figura 3.8. Fragmento 3 da interface da IEEE

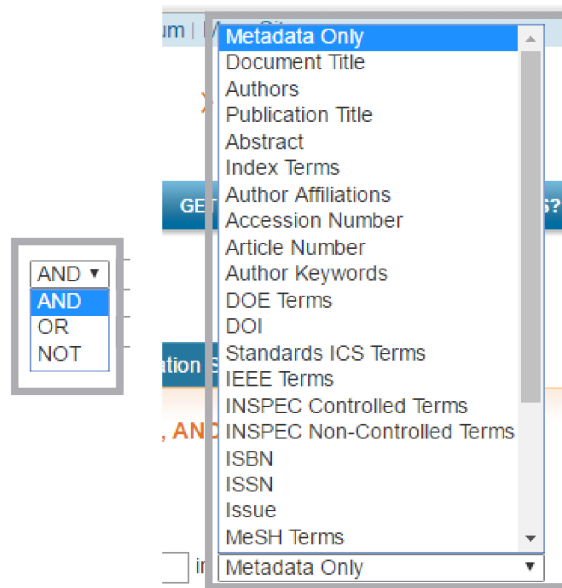


Figura 3.9. Fragmento 4 da interface da IEEE

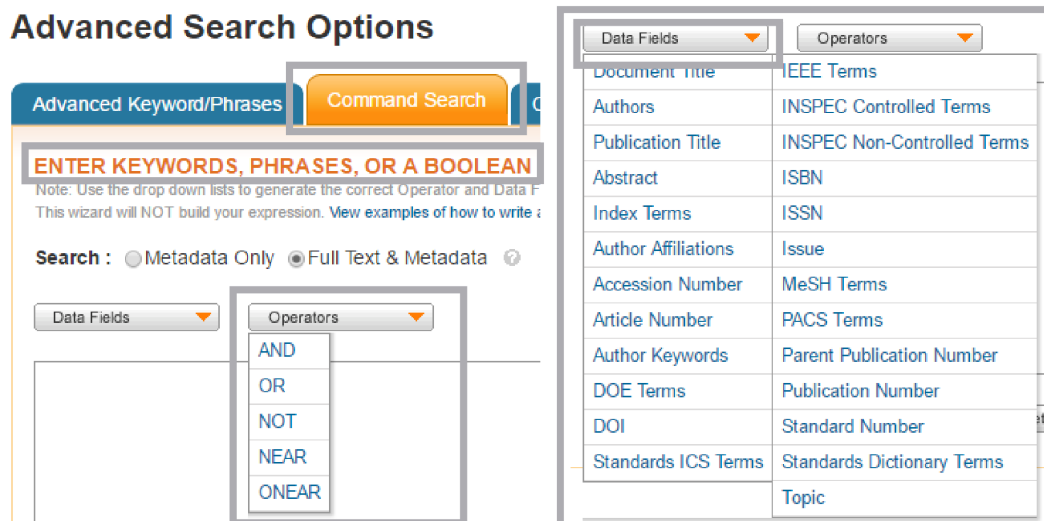


Figura 3.10. Fragmento 5 da interface da IEEE

O que você quer ou precisa fazer: Você quer um site de busca acadêmica simples e amplo, onde pode pesquisar por várias disciplinas e fontes: artigos, teses, livros, resumos e opiniões judiciais, de editoras acadêmicas, organizações profissionais, bibliotecas online, universidades e outros sites. Você quer acessar o documento buscado de forma completa (caso seja um usuário habilitado), acompanhar as novidades em qualquer área de pesquisa e também saber quem está citando determinado trabalho. Você quer realizar buscas simples ou avançadas.

Você quer um espaço onde pode inserir sua busca e um botão para enviá-la, ou apenas pressionar a tecla **Enter**. Você quer apenas com um clique definir se os resultados conterão citações e/ou patentes. Você precisa de uma barra de buscas, localizada no canto superior direito da página, para inserir seus termos de busca, e quer que esta barra esteja disponível em todas as telas. Você quer, em um outro espaço, definir configurações gerais para suas buscas, como quantidade de resultados por página, exibição de um link para exportação em padrão *BibTex*.

Sobre a sua busca, você quer saber quantos resultados foram obtidos e quanto tempo foi gasto para a busca retornar resultados, exibindo-os em várias páginas. Você quer que o controle de navegação de páginas apareça no fim da página. Você quer poder alternar o tipo de arquivo que está buscando (artigo acadêmico, documentos jurídicos, ou ainda documentos em sua própria biblioteca). Você quer poder definir um ano inicial para a data de publicação dos trabalhos resultantes, poder ordenar os resultados por relevância ou por data, e decidir se seus resultados incluirão patentes e/ou citações. Você precisa de um botão para criar um alerta para a busca realizada. Você quer que essas opções apareçam na lateral direita da interface.

No centro da página você quer que sejam exibidos os resultados da busca, com informações básicas de cada resultado. Sendo estas informações: tipo do documento (citação, patente ou artigo), título, autores, ano, onde foi publicado, quantidade de citações, artigos relacionados, versões disponíveis, além de um link para download do PDF (quando disponível), link para citar, salvar e realizar outras ações.

Você quer um modo de busca avançada e quer filtros pré-definidos para inclusão e exclusão de termos de busca, inserindo termos em que a busca deverá conter exatamente todas as palavras inseridas ou pelo menos uma, uma frase exata, ou não conter as palavras inseridas; você também quer definir em qual parte dos documentos(ex.: resumo, título, texto completo) será aplicada a busca, definir autor, local de publicação e/ou ano. Você precisa que a barra de busca apresente a sintaxe da busca atual, sendo que alterações feitas na caixa de busca avançada só devem ser efetivadas quando clicar no botão ou apertar a tecla **Enter**. Você quer que seja indicado se um termo não deve aparecer nos resultados (representado por “-”).

Formas que você pode ou deve utilizar o sistema: Na barra de buscas, você deve digitar os termos que deseja buscar e depois pode clicar no botão ao lado ou apertar a tecla **Enter** para enviar sua busca. Você pode navegar pelas páginas de resultado, indo para a próxima página ou para a anterior. Para definir o tipo de arquivos resultantes, limitar o ano ou ordenar sua busca, você deve clicar nas opções disponíveis na lateral direita da interface. Quando você passar o mouse em cima de algum link, o texto é realçado e para selecioná-lo você deve clicar sobre ele.

Você deve clicar no link para download do PDF quando disponível, ou pode apenas salvar o documento em sua biblioteca. Você pode realizar uma busca avançada, inserindo termos nos campos de refinamento da busca. Você pode entrar em um modo de busca avançada clicando no botão para isso, e inserir os termos de busca nas opções pré-definidas. Para incluir uma frase como termo de busca, ela deve vir entre aspas ou inseridas no campo de busca avançada específico, caso contrário, as palavras serão consideradas em separado. Agrupamentos de termos com parênteses e “AND” não são considerados, sendo que este último é considerada uma palavra comum. Na busca avançada, você pode separar os termos de um mesmo grupo por vírgulas ou apenas espaços em branco, que o resultado obtido será igual.

Evidências: a inspeção semiótica da interface do Google Scholar gerou evidências diversas a partir dos signos metalinguísticos, estáticos e dinâmicos, sendo os pontos observados discutidos na metamensagem do sistema. Assim, apenas algumas dessas evidências serão apresentados nas Figuras 3.11, 3.12 e 3.13.

Stand on the shoulders of giants.

Google Scholar provides a simple way to broadly search for scholarly literature. From one place, you can search across many disciplines and sources: articles, theses, books, abstracts and court opinions, from academic publishers, professional societies, online repositories, universities and other web sites. Google Scholar helps you find relevant work across the world of scholarly research.

Features of Google Scholar

- Search all scholarly literature from one convenient place
- Explore related works, citations, authors, and publications
- Locate the complete document through your library or on the web
- Keep up with recent developments in any area of research
- Check who's citing your publications, create a public author profile

Figura 3.11. Fragmento 1 da interface do Google Scholar

3.3 Resultados da Triangulação

Além da análise das interfaces de busca acadêmica da ACM, IEEE e Google Scholar, realizadas através da inspeção semiótica, objetivou-se uma análise com uma visão científica, a fim de identificar questões pertinentes para a posterior proposta de uma linguagem visual para busca acadêmica. Os resultados da inspeção dos três sistemas foram triangulados, destacando-se aspectos em comum e únicos de cada interface. A

The screenshot shows a Google Scholar search interface. The search bar contains the text "visual language". Below the search bar, it indicates "About 50,300 results (0.16 sec)". On the left side, there is a sidebar with filters: "Articles" (selected), "Case law", "My library", "Any time" (with options for "Since 2016", "Since 2015", "Since 2012", and "Custom range..."), "Sort by relevance" (selected), "Sort by date", "include patents" (checked), and "include citations" (unchecked). The main content area displays three search results:

- Visual language discrimination in infancy** [PDF]
WM Weikum, A Vouloumanos, J Navarra... - ..., 2007 - science.sciencemag.org
Abstract This study shows that 4-and 6-month-old infants can discriminate languages (English from French) just from viewing silently presented articulations. By the age of 8 months, only bilingual (French-English) infants succeed at this task. These findings reveal ...
Cited by 190 Related articles All 18 versions Web of Science: 115 Cite Save
- An integrated data flow visual language and software development environment**
JR Rasure, CS Williams - Journal of Visual Languages & Computing, 1991 - Elsevier
The Khoros infrastructure consists of several layers of interacting subsystems. A user interface development system (UIDS) combines a high-level user-interface specification with methods of software development that are embedded in a code generation tool set. The ...
Cited by 144 Related articles All 2 versions Cite Save
- VIVA: A visual language for image processing**
SL Tanimoto - Journal of Visual Languages & Computing, 1990 - Elsevier
Visualization of Vision Algorithms (VIVA) is a proposed visual language for image processing. Its main purpose is to serve as an effective teaching tool for students of image processing. Its design also takes account of several secondary goals, including the ...
Cited by 142 Related articles All 2 versions Cite Save

Figura 3.12. Fragmento 2 da interface do Google Scholar

The screenshot shows the search filters section of Google Scholar. The search bar contains "allintitle: visual language". Below the search bar, there are several filter options:

- Find articles**
 - with all of the words: "visual language"
 - with the exact phrase: [empty]
 - with at least one of the words: [empty]
 - without the words: [empty]
 - where my words occur: anywhere in the article
- Return articles authored by**: [empty] (e.g., "PJ Hayes" or McCarthy)
- Return articles published in**: [empty] (e.g., J Biol Chem or Nature)
- Return articles dated between**: [empty] - [empty] (e.g., 1996)

A search button with a magnifying glass icon is located at the bottom of the filter section.

Figura 3.13. Fragmento 3 da interface do Google Scholar

partir da triangulação foi possível o delineamento da linguagem de busca acadêmica, especificando quais os elementos comuns a este tipo de linguagem. Por fim, foram discutidas estratégias de interação observadas nas interfaces em estudo.

3.3.1 Apreciação da Inspeção em Interfaces de Busca Acadêmica

Os três sistemas inspecionados apresentam semelhanças em muitos aspectos. Assim, escolheu-se apresentar a apreciação da inspeção realizada de modo a combinar o que se observou nesses sistemas, realçando as diferenças quando existem. Nos três sistemas a apresentação dos signos avaliados é consistente e faz uso de representações comuns à sistemas *web* em geral e de busca. A barra de buscas, elemento de grande importância no contexto de interfaces de busca, é apresentada em todas as páginas, sendo reduzida no espaço onde são exibidos os resultados e destacada quando o usuário ainda não inseriu a busca (com exceção da biblioteca digital da IEEE), instigando-o a inserir seus termos de busca naquele local. Os demais elementos são organizados de acordo com o modelo julgado adequado para melhor exibição dos resultados e facilitar a navegação do usuário por eles e possíveis edições na busca. Assim, os resultados são exibidos no centro da tela, logo abaixo da barra de buscas e ao lado das opções de refinamento da busca disponíveis.

A interação ocorre por meio de cliques do mouse (ou toques) e inserção de texto pelo teclado, sendo necessários poucos cliques para a realização de tarefas, sejam elas mais simples ou mais complexas. A inserção de texto pode ocorrer utilizando linguagem natural (ex.: inglês), combinada com elementos da interface (ex.: seleção de opções de refinamento) e/ou com a sintaxe do sistema de busca (ex.: expressões referentes aos operadores lógicos). Uma *string* de busca pode ser definida a partir de uma interface para busca simples ou interfaces para busca avançada. Onde a busca avançada é apresentada de formas diferentes em cada sistema, podendo ocorrer por meio de uma interface com campos pré-definidos, ou também por manipulação direta da sintaxe.

Na busca simples é exibida apenas um campo para inserção de termos, de modo que não há qualquer indicação sobre como o usuário deve proceder para inserir uma combinação maior de termos e grupos de termos. Ou seja, não é indicado se devem ser usadas vírgulas ou espaços para separar os termos, aspas duplas ou aspas simples para expressões, sendo que as representações são diferentes em cada sistema. Por exemplo, o espaço entre termos é interpretado na ACM como sendo o operador OR, enquanto que no Google Scholar o mesmo espaço é interpretado como sendo AND. Além disso, quando o usuário deseja realizar buscas mais complexas a partir da busca simples, o

uso explícito dos operadores lógicos e opções de refinamento, pode não ser interpretado pelo sistema da forma como o usuário imagina. Na IEEE, apesar de a busca retornar resultados para a inserção explícita de operadores lógicos na busca simples, observa-se que esses resultados são diferentes quando a mesma estrutura é definida na busca avançada. Sendo que na IEEE o uso de operadores lógicos deve ser acompanhado da expressão *matchBoolean = true* na URL da busca, onde é indicado ao sistema que elementos booleanos serão usados na definição da *string* de busca. Na ACM e no Google Scholar, uma inserção explícita dos operadores lógicos AND e NOT na busca simples, não resulta na aplicação deles na busca, mas sim neles como termos comuns na busca; de modo que apenas o operador OR é considerado, sendo AND representado por +, e NOT representado por -.

Existem várias formas para refinar os resultados ou filtrar a busca a partir de determinados fatores. Em todos os sistemas existem, por exemplo, opções para limitar ano, definir nome de autor ou editor, inserir termos obrigatórios ou termos a serem excluídos da busca. Contudo, a forma como as opções de refinamento são apresentadas difere de um sistema para outro. Na ACM e na IEEE, essas opções são apresentadas ao lado esquerdo dos resultados da busca, sendo apresentada uma lista pré-definida de termos que podem ser aplicados; além disso, esses mesmos refinamentos podem ser definidos pelo usuário no modo de busca avançada. No Google Scholar, são apresentados ao lado esquerdo dos resultados apenas opções para limitar ano e tipo de publicação, sendo que os demais refinamentos da busca estão presentes no modo de busca avançada, com campos pré-definidos a serem preenchidos pelo usuário.

É interessante notar que nos sistemas da ACM e IEEE, que possuem domínio específico e conteúdo limitado ao seu repositório, a quantidade de opções de refinamento é maior e opções como nome de autor são pré-carregadas a partir dos resultados da busca. Ou seja, são exibidos possíveis valores que os filtros de refinamento dos resultados podem assumir. Por outro lado, no Google Scholar, as opções de refinamento da busca ocorrem apenas através do ano, autor e editor, e não há um pré-carregamento de nenhum desses valores.

Nas interfaces de busca avançada dos três sistemas, as opções de refinamento e elementos de combinação são exibidos aos usuários através de campos específicos para aquela informação e para aquele sistema. Alguns elementos da busca avançada são expressos de forma igual ou bem próxima entre estes três sistemas, como o filtro de busca por nome de autor: ACM, Google Scholar – *author* :< *name* >, IEEE – “*Authors*” :< *name* >. Enquanto que outros elementos são diferentes em cada sistema, como a definição do local de busca nos documentos como sendo o título dos documentos: ACM – *acmdlTitle*(< *title* >), IEEE – “*DocumentTitle*” :< *title* >,

Google Scholar – *allintitle* :< *title* >. Na busca avançada da ACM, o usuário pode visualizar alterações na busca e na sua sintaxe sempre que insere/edita um elemento. Já na IEEE e no Google Scholar, o usuário só pode visualizar alterações na busca e na sua sintaxe ao inserir/editar um elemento, apenas depois de concluir a ação.

Em geral, a sintaxe de busca deve ser aprendida através da observação da URL resultante da busca realizada. Apesar de a biblioteca da ACM apresentar a sintaxe da busca inserida no modo de busca avançada, essa apresentação não está visível o tempo todo (o usuário deve clicar em um botão para exibir), e conforme representado na Figura 3.14, o agrupamento de dois grupos OR com uso do operador lógico AND só é possível se o usuário editar sua *string* diretamente na sintaxe. Na IEEE, o usuário pode realizar busca avançada ou por citação, preenchendo campos com opções pré-definidas, e um modo de busca por comando (similar à sintaxe de busca na ACM), em que se tem uma lista de campos de dados e operadores para definição da busca. As possíveis dificuldades que o usuário possa vir a ter na interação com a biblioteca digital da IEEE são facilmente resolvidas com consulta ao manual e exemplos de uso. Além disso, ao fornecer a busca por comando, a IEEE dá mais poder ao usuário de inserir elementos na *string* na ordem em que desejar. Enquanto que no Google Scholar, não existe na interface uma forma de manipulação direta da sintaxe de busca.

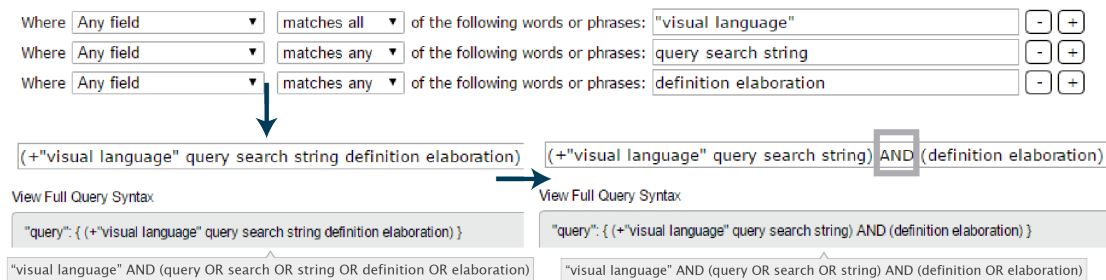


Figura 3.14. Exemplo de agrupamento na ACM

Em resumo, os sistemas de busca acadêmica da ACM, IEEE e Google Scholar, atendem às expectativas de usuários que desejam realizar uma busca simples, com poucos termos e aplicação de filtros. Contudo, em todos se observou um fraco *feedback* quanto à aplicação da *string* de busca e como ela está sendo processada. Além disso, o uso dos sistemas no modo avançado, com a definição da busca utilizando a linguagem do sistema, é dificultado pelo fato de o usuário não ter um manual na ACM e no Google Scholar que liste quais expressões representam os campos de dados no sistema.

3.3.2 Linguagem de Busca Acadêmica

Durante a realização da inspeção semiótica nas três interfaces de busca acadêmica selecionadas, foi observado que muitos dos elementos existentes são comuns a todas. Entretanto, alguns destes elementos observados não estão presentes em interfaces de busca de assuntos gerais (ex.: Google), sendo apresentados quase que exclusivamente dentro deste contexto específico de busca. Desse modo, encontra-se em interfaces de busca acadêmica a expressão de uma linguagem própria empregada no processo de busca. Sendo esta linguagem dividida em diferentes níveis de profundidade, partindo de uma linguagem voltada para uma busca simples, até uma linguagem voltada para uma busca mais avançada e refinada. Onde esses níveis de profundidade definem os elementos de conteúdo, combinação e contexto da linguagem de busca acadêmica.

Os **elementos de conteúdo** da linguagem de busca acadêmica são aqueles que representam os dados de interesse do usuário. Ou seja, representam o conjunto de palavras e expressões na linguagem. Esses elementos são definidos pelo próprio usuário e inseridos em um espaço específico para isso na interface dos sistemas. Logo, a partir desses elementos é possível a construção de uma busca acadêmica, sendo estes elementos:

- **Termo:** representa uma palavra na *string* de busca. Tem-se em um termo, o menor elemento da linguagem de busca, por meio do qual são construídas as demais relações.
- **Expressão:** representa um conjunto de palavras na *string* de busca. Ou seja, são palavras que o usuário deseja buscar de modo que apareçam nesta ordem exata nos documentos resultantes, sem que haja outras palavras entre elas. Como é o caso da busca por “linguagem visual”.

Em um nível mais aprofundado da busca, encontram-se os **elementos de combinação**. Onde estes representam as possíveis combinações entre os demais elementos da linguagem. Ou seja, ao definir uma *string* de busca, o usuário pode combinar termos e expressões, gerando uma estrutura de busca complexa. Essas combinações são estruturadas a partir da lógica booleana e da hierarquia do agrupamento de elementos. Assim, os elementos sintáticos são:

- **Grupo:** representa um agrupamento de termos e/ou expressões na *string* de busca. São os diferentes níveis de hierarquia dentro da *string*, onde os elementos são conectados por operadores lógicos.

- **String:** representa a expressão completa de busca, sendo definida pelo usuário e utilizada pelo buscador. Uma *string* é composta por termos, expressões e grupos de elementos, que traduzem a busca de informação definida pelo usuário.
- **AND:** determina que dois termos ou conjuntos de termos ligados pelo operador AND devem ser aplicados na busca e constar nos documentos resultantes.
- **OR:** determina que dois termos ou conjuntos de termos ligados pelo operador OR devem ser aplicados na busca e pelo menos um deles deve constar nos documentos resultantes.
- **NOT:** determina que um termo ou conjunto de termos precedido pelo operador NOT deve ser aplicado em negação na busca, não devendo constar nos documentos resultantes.

Os **elementos de contexto** da linguagem de busca acadêmica são específicos ao contexto científico, ou seja, têm um componente semântico associado. Em geral, são expressos através de elementos de refinamento da busca, onde estão intrinsecamente relacionados à recuperação de documentos acadêmicos. Os elementos expressos por este aspecto da linguagem são construídos a partir dos elementos de identidade do documento (ex.: título, autor, ano) e demais metadados de textos científicos disponibilizados na Internet. Entretanto, no refinamento da busca, os elementos são apresentados de diversos modos. Alguns sistemas são mais completos no sentido de apresentar mais elementos de refinamento, enquanto outros são menos completos. Porém, observa-se que sistemas com mais possibilidades de refinamento da busca possuem uma base de dados possivelmente menor e mais coesa, onde a maioria (ou todos) os documentos têm os mesmos campos de metadados. Sistemas com uma quantidade menor de elementos de refinamento, recuperam documentos a partir de uma base de dados maior e menos coesa, onde os metadados dos documentos são variados e não padronizados. Pode-se dizer que todos os sistemas de busca acadêmica apresentam elementos para refinamento da busca, seja em menor ou maior quantidade. Sendo os elementos de contexto básicos:

- **Autor:** todos os documentos recuperados na busca deverão conter o nome do autor definido na *string* de busca. Servindo tanto como um filtro para os resultados obtidos, quanto como um elemento de busca.
- **Publicação:** todos os documentos recuperados na busca deverão ter o mesmo editor (ex.: evento, revista) definido na *string* de busca.

- **Ano:** todos os documentos recuperados na busca deverão ser limitados ao(s) ano(s) definido(s) na *string* de busca. Por padrão, quando apenas um ano é inserido, o limite superior é a data atual da busca.
- **Busca no documento:** a *string* de busca será aplicada em qualquer campo dos documentos existentes na base de dados do sistema de busca.
- **Busca no resumo:** a *string* de busca será aplicada apenas no resumo dos documentos existentes na base de dados do sistema de busca.
- **Busca no título:** a *string* de busca será aplicada apenas no título dos documentos existentes na base de dados do sistema de busca.

Partindo deste conjunto de elementos, os sistemas de busca apresentam abstrações próprias destes elementos em suas interfaces. Entretanto, por mais diferentes que sejam essas abstrações, observam-se representações específicas para busca simples e busca avançada. Onde essas representações ocorrem de acordo com a proposta e finalidade do sistema de busca, de modo que alguns sistemas podem apresentar outros elementos em sua composição. Assim, conclui-se que, diferentes sistemas apresentam diferentes representações da linguagem de busca acadêmica.

Dado que a linguagem de busca acadêmica discutida aqui foi definida a partir da inspeção de três sistemas, a mesma não pode ser generalizada para todos os sistemas de busca. Entretanto, pode-se dizer que essa linguagem pode ser aplicada em quaisquer sistemas de busca, onde os elementos básicos de uma consulta são definidos. Além disso, é importante considerar a existência de outros elementos dentro do contexto de interfaces de busca acadêmica e que não foram possíveis de serem observados nesta análise.

3.3.3 Estratégias de Interação

Considerando que a linguagem de busca acadêmica é aplicável para todos os sistemas de busca analisados, tem-se a linguagem voltada para dois diferentes tipos de usuário: comuns (não conhecem muito sobre lógica booleana ou do sistema e fazem buscas esporadicamente, geralmente apenas digitam termos) e avançados (têm algum conhecimento de lógica booleana e do sistema e fazem buscas com maior frequência, geralmente aplicam filtros para definição da *string*).

Nos sistemas de busca acadêmica inspecionados através do MIS, a linguagem de busca acadêmica é definida a partir de diferentes abstrações dos elementos componentes na interface e na sintaxe de busca. Dado então que estes sistemas possuem a mesma

linguagem geral, é natural que as estratégias para exibição desta linguagem ocorram na mesma direção. Assim, foram identificadas quatro estratégias de interação a partir das representações definidas por estes sistemas de busca acadêmica, discutidas a seguir.

- Estratégia 1 – busca simples: representam a informação mínima a ser oferecida pelo usuário para que a busca seja feita. Ou seja, dentro da linguagem de busca acadêmica, são os elementos definidos no nível da busca simples, os termos de busca. Normalmente, os sistemas de busca abstraem os elementos de busca simples em um campo em branco para que o usuário entre um ou mais termos que devem compor a *string* de busca. Estão presentes em todas as páginas da interface, aparecendo em destaque na página inicial da busca e em tamanho menor no topo das demais páginas. São expressos pela barra de buscas, o botão para envio da busca e linguagem natural na inserção dos termos de busca (Figura 3.15). Vale ressaltar que os sistemas definem como estes elementos serão combinados (por exemplo, se usando um OR ou AND). No entanto, a definição da combinação é implícita (não é apresentada em signos estáticos na interface) e o usuário só pode entendê-los através do resultado da busca (signos dinâmicos) ou de instruções, que podem estar disponíveis sob demanda (signos metalinguísticos).

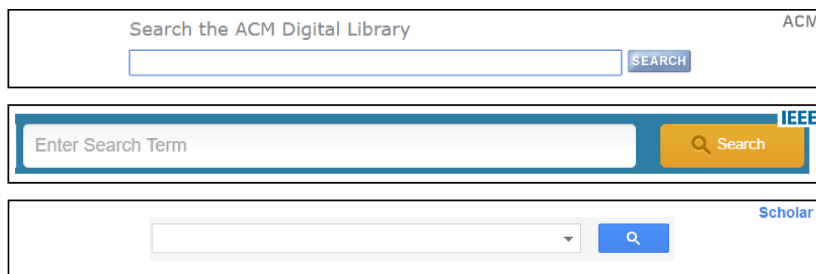


Figura 3.15. Exemplo de signos de busca simples

Espera-se que o usuário organize a *string* de busca de acordo com suas expectativas de resultados, onde os termos e expressões são definidos em seu idioma ou no idioma em que se espera obter os resultados (ex.: inglês, português). Outro ponto importante a ser destacado aqui é o fato de que no contexto da busca simples o usuário pode inserir buscas mais complexas, definindo-a por meio da sintaxe de busca do sistema e explicada mais adiante.

- Estratégia 2 – busca avançada na interface: a busca avançada é apresentada a partir de duas abordagens diferentes. Sendo que em uma dessas abordagens os elementos de busca avançada são representados na interface e organizados de

forma intuitiva, guiando o usuário quanto às combinações disponíveis e qual informação ele deve inserir. Espera-se que o usuário ao interagir com a linguagem em um nível avançado já conheça e tenha utilizado o sistema para realizar buscas. Normalmente, os elementos que regem a combinação são disponibilizados através de campos específicos para a inserção de informação. Alguns sistemas apresentam esses elementos diretamente em sua forma natural, enquanto outros os apresentam em um modo que julgam mais intuitivo. Por exemplo, quando se tem o elemento de combinação AND, pode-se ter um campo onde o usuário insere um termo e então escolhe o operador AND como elemento de combinação. Em outra interface, este mesmo elemento de combinação é expresso através de frases como “contenha todos” ou “com todas as palavras”. Ou seja, tem-se na busca avançada na interface estratégias definidas a fim de, não exigir do usuário o conhecimento da sintaxe da linguagem de representação da *string*. A interface da busca avançada pode ser acessada a qualquer momento por demanda nos três sistemas inspecionados por meio do botão próximo a barra de busca padrão. Alguns sistemas apresentam filtros pré-definidos para busca e campos para inserção de *strings*, permitindo a definição de elementos específicos do contexto acadêmico. Exemplos das representações da busca avançada na interface em cada uma das três interfaces inspecionadas são apresentados a seguir na Figura 3.16.

- Estratégia 3 – busca avançada na sintaxe: na outra abordagem de representação da busca avançada, os elementos de combinação são expressos pela sintaxe do sistema de busca. Quando o usuário insere uma *string* de busca, simples ou avançada, o sistema a “enxerga” em uma sintaxe de busca própria. A sintaxe de busca é definida pelos termos de busca, operadores lógicos e hierarquia condicionada por parênteses. Sendo que as relações formais entre termos e combinações são determinadas pela sintaxe. Assim, a *string* de busca definida pelo usuário é traduzida para uma representação que o sistema compreende, ou seja, traduzida para uma representação onde as relações formais entre cada um dos elementos componentes são definidas de acordo com o sistema. Portanto, ao considerar todos os elementos da linguagem no sistema de busca, a sintaxe é também uma expressão da busca avançada. A diferença entre a busca avançada na interface, é que a partir da sintaxe o usuário tem maior controle e liberdade para inserir elementos em sua *string*. Podendo manipular e organizar a estrutura da *string* de busca de variadas maneiras, mesmo que nem sempre o sistema seja capaz de interpretá-la como esperado pelo usuário. A sintaxe de busca pode ser visualizada sempre que o usuário realiza uma busca, seja na página de resultados, barra

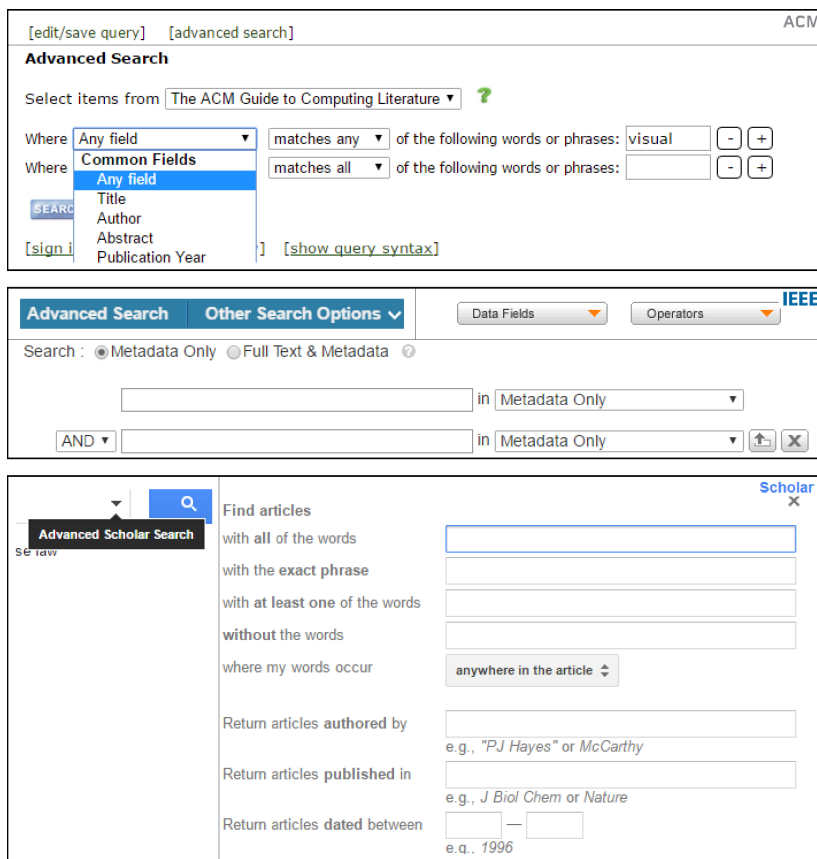


Figura 3.16. Exemplos de signos de busca avançada na interface

de navegação (URL) do *browser* ou ainda em um espaço destinado à visualização e edição da sintaxe. Poucos sistemas apresentam uma interface específica para interação com a sintaxe, sendo que na maioria das vezes para conhecer o formato sintático o usuário deve observar a URL resultante da busca realizada e como o sistema interpreta cada um dos elementos de combinação que ele inseriu. Com isso, observa-se que as interfaces desses sistemas desencorajam o uso direto da sintaxe de busca ou presumem que os usuários não vão querer usá-la. As representações sintáticas da busca nos três sistemas são apresentadas na Figura 3.17, onde é apresentado o campo onde a sintaxe pode ser editada, o modo como o sistema exibe a sintaxe ao exibir os resultados da busca e a sintaxe apresentada na URL da busca.

- Estratégia 4 – refinamento dos resultados da busca: o processo de busca é um ciclo, partindo da definição de uma *string*, seguindo para a recuperação de documentos e refinamento. Onde esse refinamento pode ocorrer tanto na definição da *string* de busca, limitando a quantidade de resultados, quanto na visualização

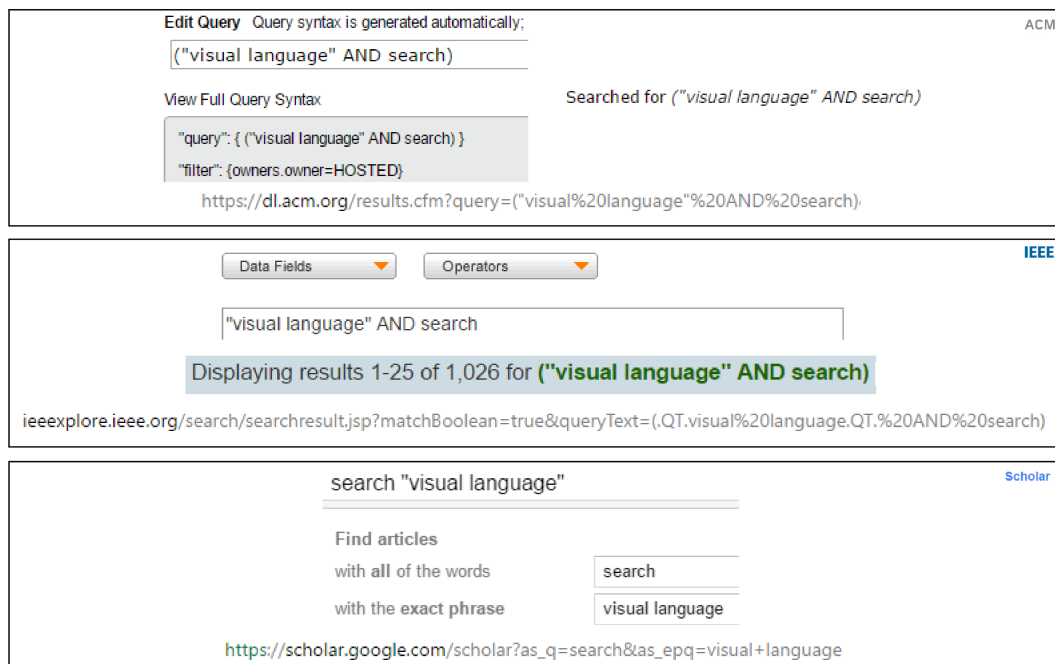


Figura 3.17. Exemplos de signos sintáticos da busca

dos resultados, gerando um subconjunto da busca realizada. Quando uma busca é feita, os resultados recuperados são exibidos através de signos próprios de cada documento, em uma listagem desses documentos. Além disso, a interface de exibição de resultados também apresenta a *string* de busca e uma série de opções de refinamento, relativas aos elementos de busca avançada. Assim, considera-se que o sistema de busca compreende a necessidade do usuário em alterar informações inseridas na *string* de busca ou de limitação da busca. Tendo na interface de resultados recuperados, a apresentação de estratégias para refinamento da busca depois que ela é definida e realizada. Na Figura 3.18, são apresentados exemplos dos signos da interface dos sistemas de busca inspecionados, onde estes se referem às estratégias de exibição de resultados direcionada ao refinamento da busca.

Considerando que um artigo científico apresenta vários metadados, é possível que a *string* de busca seja direcionada especificamente a algum deles. Levando a possibilidades de definição do campo de busca da *string* como, por exemplo, sendo o título do documento, o resumo, ou apenas a conclusão. Além disso, em alguns sistemas o refinamento da busca possibilita a filtragem por metadados como número DOI, ISSN e outros que se referem diretamente a dados de publicação científica. Critérios de refinamento são aplicados com o objetivo de conduzir os resultados obtidos na busca para um determinado grupo de documentos. O

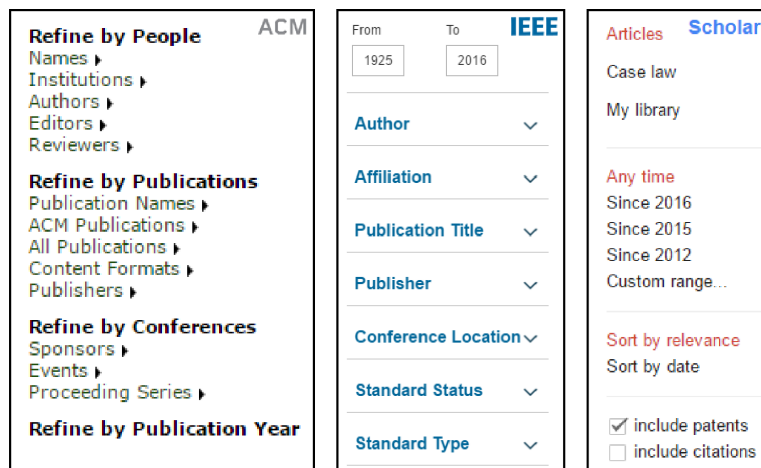


Figura 3.18. Exemplos de signos de refinamento dos resultados da busca

nome de autor, por exemplo, pode ser empregado como um campo para definição de um termo de busca, onde a informação é o nome do autor. Em outros sistemas, o nome do autor serve também como um critério para filtrar os resultados, recuperando apenas documentos que contenham o nome daquele autor. Outro critério de refinamento que integra a linguagem de busca acadêmica, é a limitação do ano em que o documento foi publicado. Uma vez o ano se referindo a quando o documento foi publicado, o usuário pode direcionar os resultados dentro de um intervalo de anos, ou determinar a partir de qual ano ou até qual ano ele deseja obter resultados para aquela consulta.

A linguagem de busca acadêmica, definida anteriormente, apresenta elementos organizados em três diferentes níveis de profundidade (conteúdo, combinação e contexto). Da mesma forma, as estratégias de interação utilizadas pelos sistemas de busca acadêmica, também são organizadas a partir destes níveis. Assim, tem-se na Estratégia 1, o foco nos aspectos lexicais relacionados ao conteúdo. Nas estratégias 2 e 3, o foco é em permitir que o usuário interaja com os elementos de combinação, seja essa interação pela interface ou diretamente pela sintaxe. Por fim, na Estratégia 4, é considerado na interação com a interface que o usuário tem um conhecimento semântico do contexto, que neste caso é representado por publicações científicas.

Considerando então a linguagem de busca acadêmica, tem-se nestas estratégias de interação modos de abstração da linguagem específicos para os sistemas inspecionados. É interessante notar que apesar das diferenças entre as representações da linguagem de busca acadêmica nos três sistemas, as estratégias permanecem as mesmas. Além disso, estas mesmas estratégias e elementos de linguagem, possivelmente são um conjunto único do contexto de busca acadêmica. Desse modo, observa-se que

os sistemas de busca acadêmica estão inseridos em um contexto de interação e abstração de elementos próprio. Permitindo-se assim que diferentes estratégias de interação sejam definidas, onde a linguagem representada é sempre a mesma. Tendo, portanto, neste trabalho, a abstração visual da linguagem de busca acadêmica como sendo uma estratégia de interação alternativa a estas apresentadas nos sistemas da ACM, IEEE e Google Scholar.

Capítulo 4

A Linguagem Visual VILAS

Como objetivo principal deste trabalho teve-se a definição de uma linguagem visual, chamada VILAS (*Visual Language for Academic Search*). A definição da linguagem se deu posteriormente à inspeção de interfaces de busca acadêmica, identificando os elementos que compõem a linguagem de busca acadêmica e as estratégias de interação utilizados por estes sistemas de busca. Tem-se, portanto, na linguagem visual VILAS, uma representação visual dos elementos identificados como elementos básicos da linguagem de busca acadêmica e discutidos na Subseção 3.3.2.

Na Seção 4.1 é descrito brevemente o processo de construção de VILAS, apresentado exemplos das diferentes abstrações propostas até se chegar a representação atual. Os elementos que compõem a linguagem visual e sua respectiva representação são assunto na Seção 4.2. Uma discussão acerca da correspondência entre os elementos de busca representados em VILAS e nos sistemas de busca acadêmica da ACM, IEEE e Google Scholar, é feita na Seção 4.3.

4.1 Processo de Definição de VILAS

Buscou-se no *design* da linguagem visual uma organização perceptiva dos elementos da linguagem de busca acadêmica, onde o processo de definição da representação visual foi apoiado nas teorias da escola Gestalt. Os princípios da Gestalt são um conjunto de princípios que ajudam a explicar como a mente agrupa elementos na formação de grupos, ou seja, como se dá a organização perceptiva. De acordo com a Gestalt, em representações visuais são imprescindíveis fatores como equilíbrio, clareza e harmonia visual. Na nomenclatura de [16], os sete princípios são:

- **Unidade:** um único elemento, que se encerra em si mesmo. Uma unidade pode

ser configurada como um todo, onde podem haver relações entre os elementos (subunidades). Unidades formais podem ser segregadas ou percebidas por meio de elementos isolados ou combinados, como pontos, linhas, planos, volumes, cores, sombras, brilhos, texturas e outros.

- **Segregação:** a percepção é capaz de, por meio da desigualdade de estímulos, segregar uma ou mais unidades. O contraste em uma unidade leva a capacidade perceptiva de separar, identificar, evidenciar ou destacar unidades.
- **Unificação:** os estímulos produzidos pelo objeto consistem na percepção de igualdade ou semelhança. A composição do objeto se dá a partir de fatores de harmonia, equilíbrio, ordenação visual e coerência da linguagem.
- **Fechamento:** a organização da forma tende a uma continuidade, gerando a sensação de fechamento visual.
- **Continuidade:** as partes de uma forma ou objeto se sucedem de modo coerente, sem que haja quebras ou interrupções na sua trajetória ou fluidez visual. Os elementos tendem a acompanhar uns aos outros, seja por meio do movimento, cores, brilhos, pontos, linhas ou outros.
- **Proximidade:** elementos próximos são percebidos como um todo. Há a tendência em agrupar os estímulos mais próximos entre si, como forma, cor, tamanho, brilho e outros.
- **Semelhança:** igualdade de forma e de cor desperta a sensação de agrupamento, de que objetos que recebem estímulos mais semelhantes entre si pertencem a um mesmo grupo.
- **Pregnância da forma:** sendo o princípio básico da percepção visual da Gestalt, tem-se a partir da pregnância, que uma estrutura resultante da percepção de qualquer padrão de estímulo é tão simples quanto possível dada uma condição.

Apoiando-se então nos princípios Gestalt e nos elementos classificados como essenciais para interfaces de busca acadêmica, foram feitas várias tentativas para encontrar aquela que melhor representava uma *string* de busca, considerando para tanto operadores lógicos e agrupamento de termos. O processo de definição da representação visual de VILAS ocorreu através de discussões entre a autora deste trabalho e uma especialista em artes e prototipação. Foram considerados várias formas de representação, sendo apresentadas no conjunto de Figuras 4.1 aquelas que mais se aproximam

a proposta final para representação da *string* de buscas. Procurou-se representar a estrutura da *string* de buscas em um formato que facilitasse a compreensão do usuário quanto à estrutura da *string*, organizando a representação de modo a utilizar apenas os elementos gráficos necessários ao entendimento da estrutura.

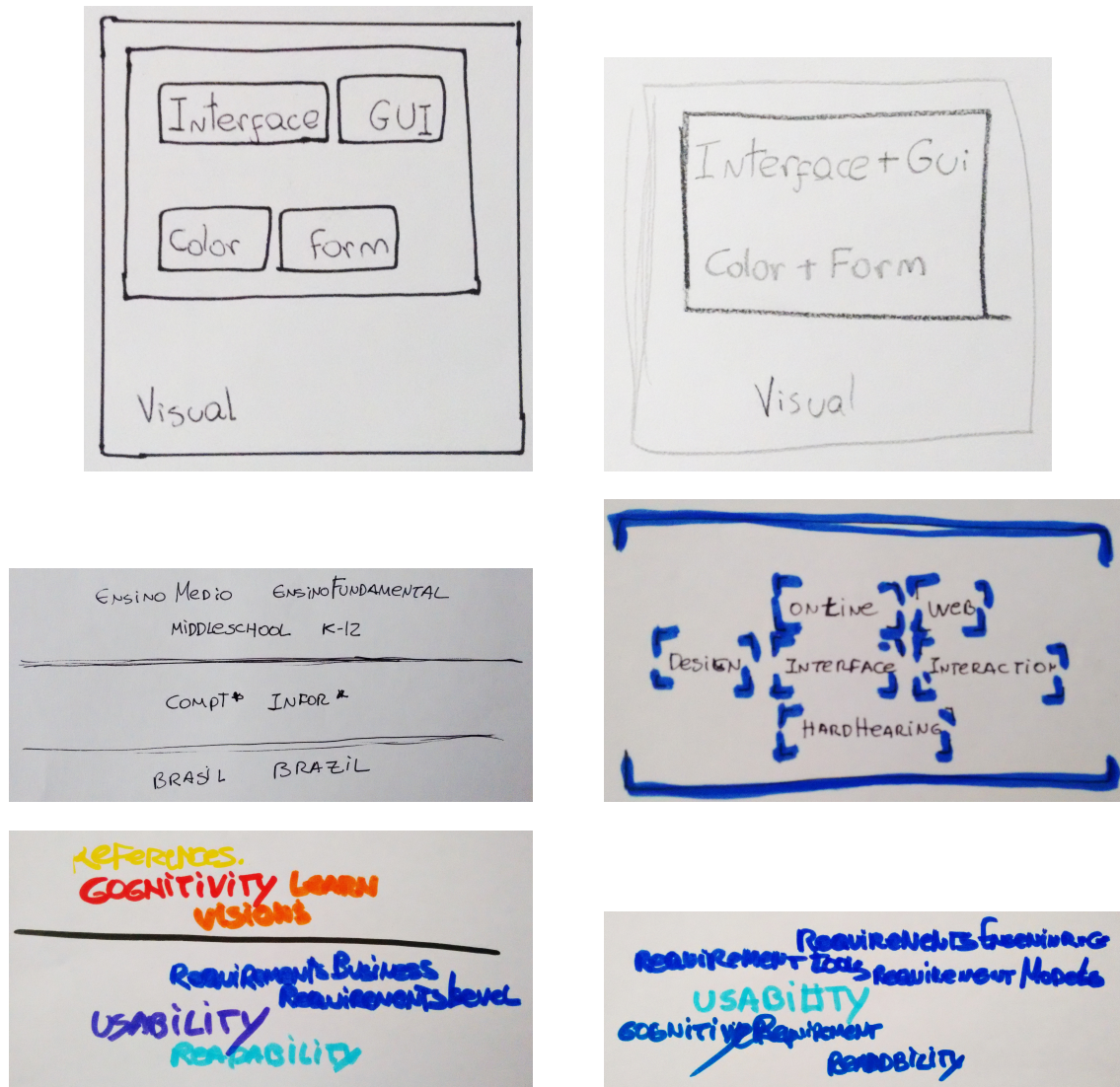


Figura 4.1. Protótipos de VILAS

4.2 Representação Visual

A Figura 4.2 representa uma *string* de busca composta de palavras, operadores lógicos e consequentemente grupos. Uma vez no contexto de *strings* de busca, os estímulos semelhantes levam o observador a entender aqueles elementos como parte de um

grupo. Enquanto que os estímulos diferentes levam à percepção de grupos diferentes. Contudo, ao observar a representação como um todo, é perceptível que aqueles grupos e elementos unitários pertencem a um todo maior, compondo portanto uma *string* de busca completa. Os estímulos enviados pelo distanciamento das palavras tendem à percepção da segregação dos elementos distantes na representação. Enquanto que os estímulos enviados pelas cores e organização dos elementos, levam à percepção de continuidade, que mesmo em grupos separados, aqueles elementos fazem parte de um mesmo todo, que por sua vez é representado pela ordenação visual e coerência.

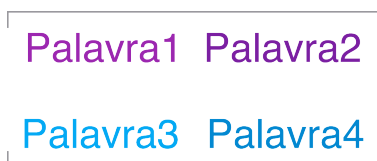


Figura 4.2. VILAS - Exemplo da representação de uma *string*

Ao inserir o operador lógico AND entre duas palavras, o usuário deseja recuperar documentos que obrigatoriamente contenham aquelas duas palavras. Na Figura 4.3, a representação visual de dois termos ligados por AND é apresentada. As palavras inseridas na *string* de busca certamente possuem alguma semelhança quanto ao teor da pesquisa sendo realizada, contudo, todas elas são consideradas como parte essencial do todo, significando, possivelmente, diferentes conceitos e elementos da pesquisa. Assim, os estímulos visuais representam que estas palavras pertencem a um mesmo grupo, mas formam grupos próprios por si próprias. Ou seja, ao apresentar cada termo entre um AND em uma linha e cores diferentes, reforça-se que cada linha representa um termo que deve estar presente nos resultados da busca.

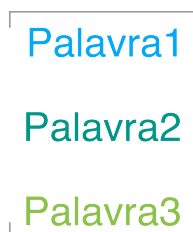


Figura 4.3. VILAS - Exemplo da representação do agrupamento AND

O operador lógico OR representa, na sintaxe de busca, o desejo do usuário em recuperar documentos que contenham pelo menos uma daquelas palavras conectadas pelo operador. Em VILAS a representação visual definida para dois termos conectados por OR pode ser vista na Figura 4.4. Geralmente, se o desejo do usuário é obter

resultados com qualquer uma daquelas palavras buscadas, essas palavras apresentam algum nível de semelhança e proximidade entre si. Portanto, os estímulos visuais para representação dessas palavras devem, ao mesmo tempo tratá-las como unidades e como partes de um grupo, ressaltando a semelhança e proximidade. Por isso, palavras ligadas com OR são dispostas uma ao lado da outra e nas mesmas cores, indicando que pelo menos uma delas deve estar nos resultados recuperados.

Palavra1 Palavra2 Palavra3

Figura 4.4. VILAS - Exemplo da representação do agrupamento OR

Se palavras ligadas por OR são representadas visualmente próximas e recebem os mesmos estímulos de cor e organização, conclui-se que a representação de palavras que compõem uma expressão seja semelhante. Como pode ser observado na Figura 4.5, em VILAS, a união das palavras traz a significação de que nenhuma outra palavra deve existir entre aquelas nos resultados. Mantendo a unidade de cada palavra por meio da diferenciação em iniciais maiúsculas, demonstrando que ainda que unificadas, cada palavra ainda é uma unidade dentro da representação.

UmaExpressão

Figura 4.5. VILAS - Exemplo da representação de uma expressão

Quando o desejo do usuário é que, uma determinada palavra ou expressão, não apareça no resultado da busca, faz-se necessário o uso do operador lógico NOT. Assim, conforme Figura 4.6, em VILAS, um termo que deve ser excluído da busca é adicionado à *string* de busca textual e visual através de um AND, sendo representando visualmente com um traço, indicando que será removido/riscado da busca.

~~Palavra~~

Figura 4.6. VILAS - Exemplo da representação NOT

Outros elementos da busca, como as opções de refinamento, não foram considerados tão relevantes no que diz respeito à representação visual da sintaxe lógica da *string*, e por isso não são representados nesta versão de VILAS. Tem-se nos filtros e campos de buscas, elementos adicionais, que de certa forma ajustam os resultados a serem obtidos. Na ferramenta que implementa a linguagem VILAS algumas dessas opções de refinamento são inseridas, onde o usuário pode defini-las através de campos

específicos na interface. A ferramenta desenvolvida é melhor apresentada e discutida no Capítulo 5.

Uma *string* mais complexa, ocasionalmente gera a conceitualização de sub-grupos dentro de outros sub-grupos, criando uma hierarquia condicionada por parênteses na representação textual. Na Figura 4.6, uma *string* mais complexa pode ser percebida, onde os estímulos de distanciamento levam ao fechamento dos sub-grupos AND, enquanto que os estímulos de proximidade levam ao fechamento dos sub-grupos OR.

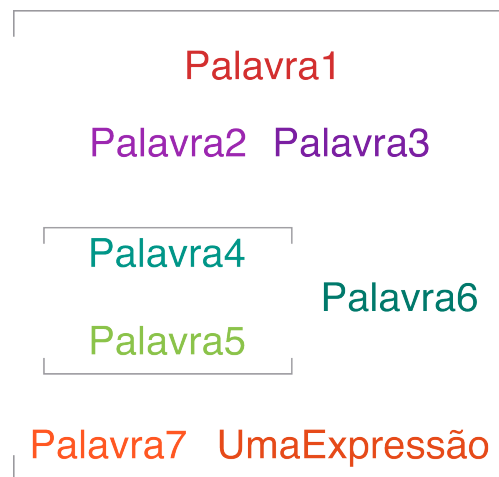


Figura 4.7. VILAS - Exemplo genérico de uma *string* complexa

Em um contexto de uso real, a mesma estrutura representada neste último exemplo, pode representar a *string* expressa textualmente por **(Design AND (Tela OR Interface) AND ((Computador AND Smartphone) OR Responsivo) AND (Flat OR "Material Design"))** e visualmente na Figura 4.8. Neste momento, é interessante atentar que os estímulos visuais descritos anteriormente para a organização da estrutura definida, são relativos à estrutura apresentada pelos elementos da *string* em formato textual. Em outras palavras, é como se a representação visual da *string* dissesse ao usuário que a busca resultante deve conter pelo menos um termo de cada linha, o que seria equivalente a dizer que deve conter pelo menos um termo de cada cor. Onde, na *string* aqui apresentada, *substrings* válidas encontradas a partir da combinação dos elementos de cada linha são: **(Design AND Tela AND Responsivo AND Flat)**, **(Design AND Tela AND Computador AND Smartphone AND Flat)**, podendo haver outras variadas *substrings*.

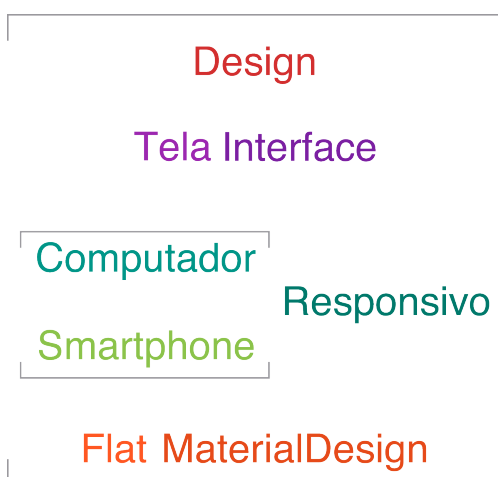


Figura 4.8. VILAS - Exemplo de uma *string* complexa equivalente à representação textual

4.3 Discussão

Os elementos que compõem a linguagem VILAS foram selecionados e definidos levando-se em conta as especificidades de cada um dos sistemas avaliados e para os quais a linguagem fornece suporte para a geração da *string* de buscas. Para tanto, procurou-se manter uma linguagem genérica, com elementos comuns ou possíveis de construção em todos esses sistemas. Logo, apoiou-se na linguagem de busca acadêmica e nas estratégias de interação identificadas e discutidas no Capítulo 3. Entretanto, na versão atual da linguagem, nem todos os elementos definidos na linguagem de busca acadêmica foram implementados, de modo que o foco foi na representação visual da estrutura da combinação de termos e operadores lógicos.

Tanto a linguagem VILAS, quanto os sistemas de busca acadêmica da ACM, IEEE e Google Scholar apresentam os mesmos elementos básicos em sua composição. Sendo assim, faz-se aqui uma discussão comparativa quanto as diferentes representações da linguagem de busca acadêmica, discutida na Subseção 3.3.2, em cada uma dessas abstrações. É importante notar que, em alguns casos, uma mesma representação é observada em mais de um sistema, de modo que são discutidas apenas uma vez relacionando todos.

- **Termo:**

ACM, IEEE, Google Scholar: os termos são a menor parte de uma *string*, representando as palavras que o usuário deseja buscar.

VILAS: as palavras ou expressões inseridas pelo usuário na caixa de busca são consideradas termos. São a menor parte de uma *string*.

- **Expressão:**

ACM, IEEE: uma expressão é representada por palavras entre aspas duplas “ ”, não podendo ser definida de outra forma.

Google Scholar: uma expressão é representada por palavras entre aspas duplas “ ”, ou palavras inseridas no campo de busca avançada *with the exact phrase*. A sintaxe utilizada é a inserção das palavras que compõem a expressão depois de “*as_epq =*”.

VILAS: uma expressão é representada por duas palavras unidas e diferenciadas por inicial maiúscula. Devendo ser inserida, na versão atual da ferramenta, em um campo de específico para expressões.

- **Grupo:**

ACM, IEEE, Google Scholar: grupos são condicionados por parênteses, limitando o resultado da aplicação de um operador lógico. Por exemplo, quando se tem a *string* (*coresAND(formsORlinhas)*), o usuário está explicitando ao sistema que deseja documentos resultantes com o termo *cores* e combinações entre *forms* e *linhas*.

VILAS: grupos são formados quando aplicado algum operador lógico (AND ou OR), discutidos mais adiante. Dentro da *string* são distinguidos como blocos de elementos com uma representação em comum, podendo ser aninhados dentro de outros grupos, criando assim um conceito de hierarquia na representação

- **String:**

ACM, IEEE, Google Scholar: as *strings* são inseridas na barra de busca padrão ou nos campos de busca avançada. Além disso, os usuários também podem manipulá-la por meio da URL da página *web* do sistema, desde que conheça a sintaxe do sistema de busca.

VILAS: todos os elementos exibidos no espaço de manipulação compõem a *string*. Estes elementos são resultados da inserção de termos, agrupamentos lógicos ou aplicação de filtros.

- **AND:**

ACM: o usuário pode inserir o operador AND explicitando-o na definição da *string* de busca em letras maiúsculas. Na busca avançada, a opção de seleção correspondente é *matches all*, cuja sintaxe é o uso do sinal + imediatamente antes da palavra que se deseja buscar.

IEEE: por padrão, todas as palavras inseridas no campo de buscas são concatenadas com um AND, ou seja, a busca retorna documentos que contenham todas elas. Na busca avançada ou na busca por comando, o usuário pode selecionar a opção AND, ou ainda, pode inserir o operador junto aos termos de busca, em letras maiúsculas.

Google Scholar: por padrão, o espaço dado entre palavras que não estão entre aspas duplas é considerado pelo sistema como sendo operador lógico AND. Outras formas de inserir o operador é explicitando-o na definição da *string* de busca em letras maiúsculas. Na busca avançada, o campo correspondente é *with all of the words* e a sintaxe utilizada é a escrita do termo depois da expressão “*as_q =*”.

VILAS: ao concatenar dois elementos com AND forma-se um novo grupo, onde os elementos estão organizados em ordem vertical e com espaçamento duplo entre eles, cercados por um colchetes com bordas no topo e no rodapé. Além disso, as cores dos termos ou subgrupos internos são diferentes, evidenciando assim que os resultados devem conter pelo menos um elemento de cada uma dessas linhas.

- **OR:**

ACM: por padrão, o espaço dado entre palavras que não estão entre aspas duplas é considerado pelo sistema como sendo o operador lógico OR. O usuário pode inserir o operador OR explicitando-o na definição da *string* de busca em letras maiúsculas. Na busca avançada, a opção de seleção correspondente é *matches any*.

IEEE: o usuário não pode inserir o operador OR na caixa de busca padrão, a menos que ele acrescente na URL do sistema de busca a expressão *matchBoolean=true*, que permite ao sistema reconhecer os operadores lógicos e é desabilitado neste primeiro momento. Na busca avançada ou na busca por comando, o usuário pode selecionar a opção OR, ou ainda, pode inserir o operador junto aos termos de busca, em letras maiúsculas.

Google Scholar: o usuário pode inserir o operador OR explicitando-o na definição da *string* de busca em letras maiúsculas. Na busca avançada, o campo correspondente é *with at least one of the words*. A sintaxe segue o mesmo padrão da sintaxe para o operador AND, devendo inserir o termo após a expressão “*as_{oq} =*”.

VILAS: ao concatenar dois elementos com OR forma-se um novo grupo, onde os elementos estão organizados em ordem horizontal e com espaçamento simples entre eles. As cores dos termos são as mesmas, representando que os resultados

devem conter pelo menos um destes termos, ou seja, pelo menos um termo na cor X.

- **NOT:**

ACM: na busca avançada, o campo correspondente é *matches none*, cuja sintaxe é o uso do sinal – imediatamente antes da palavra que não se deseja incluir na busca.

IEEE: na busca avançada ou na busca por comando, o usuário pode selecionar a opção NOT, ou ainda, pode inserir o operador junto aos termos de busca, em letras maiúsculas.

Google Scholar: na busca avançada, o campo correspondente é *without the words*, cuja sintaxe é o uso do sinal – imediatamente antes da palavra que não se deseja incluir na busca..

VILAS: quando um termo recebe o operador NOT, este termo é riscado e é concatenado à toda a *string* de busca por meio de um AND.

- **Autor:**

ACM: na busca avançada, o usuário pode selecionar a opção *author*, inserindo o nome de autor que deseja buscar, de modo que todos os documentos resultantes serão daquele autor. Pode também definir um nome de autor diretamente na caixa de buscas, utilizando a sintaxe de busca *persons.authors.personName* :. Outra opção é limitar os resultados pelo nome de autores que aparecem nos documentos, selecionando o nome que desejar posteriormente à busca, na página de resultados.

IEEE: na busca avançada ou na busca por comando, o usuário pode selecionar a opção *author*, inserindo o nome de autor que deseja buscar, de modo que todos os documentos resultantes serão daquele autor. Pode também definir um nome de autor diretamente na caixa de buscas, utilizando a sintaxe de busca “*Authors*” :. Outra maneira é limitar os resultados pelo nome de autores que aparecem nos documentos, selecionando o nome que desejar posteriormente à busca, na página de resultados, ou inserir manualmente um nome.

Google Scholar: na busca avançada, o usuário pode inserir um nome de autor no campo *Return articles authored by*. Pode também definir um nome de autor diretamente na caixa de buscas, utilizando a sintaxe de busca *author* :.

VILAS: a definição do nome de autor é válida para toda a *string*. Quando selecionada a opção, o usuário deve inserir um ou mais nomes de autores.

- **Publicação:**

ACM: na busca avançada, o usuário pode selecionar a opção *publication*, inserindo o nome do editor que deseja buscar, de modo que todos os documentos resultantes serão daquele editor. Pode também definir um editor diretamente na caixa de buscas, utilizando a sintaxe de busca *acmdlPublisherName* :. Outra alternativa é limitar os resultados pelo nome de editores que aparecem nos documentos, selecionando o nome que desejar posteriormente à busca, na página de resultados.

IEEE: o editor pode ser selecionado a partir de uma lista pré-definida de editores, tanto na busca avançada, quanto na página de resultados. Outra forma de inserir o um editor é diretamente na URL do sistema, inserindo a sintaxe “*refinements =*” e o código do editor dado pelo sistema.

Google Scholar: na busca avançada, o usuário pode inserir um nome de editor no campo *Return articles published in*. Outra forma é inserir diretamente na URL do sistema, inserindo a sintaxe “*as_publication =*” e o nome do editor.

VILAS: a definição de *publication* é válida para toda a *string*. Quando selecionado, o usuário deve inserir um ou mais nomes de editores.

- **Ano:**

ACM: na busca avançada, o usuário pode selecionar a opção *year*, definindo um intervalo de anos, de modo que todos os documentos resultantes estarão contidos neste intervalo. Pode também definir um limite de ano entre os anos dos documentos resultantes, na página de resultados. Outra forma de inserir o filtro é diretamente na URL do sistema, inserindo a sintaxe *filtered = dte = 0000bfr = 0000*, onde primeiro se insere o ano inicial e depois o ano limite.

IEEE: o intervalo de ano pode ser definido em um campo específico na busca avançada ou na página de resultados. Outra forma de inserir o filtro é diretamente na URL do sistema, inserindo a sintaxe *ranges = 0000_0000Year*, onde primeiro se insere o ano inicial e depois o ano limite.

Google Scholar: na busca avançada ou na página de resultados, o usuário pode inserir um intervalo de ano. Outra forma de limitar o ano é diretamente na URL do sistema, inserindo a sintaxe *as_ylo = 0000* para ano inicial, e *as_yhi = 0000* para ano final.

VILAS: a definição de ano é válida para toda a *string*. Quando selecionado, o usuário deve inserir um ano inicial e se desejar um ano final.

- **Busca no documento:**

ACM, IEEE, Google Scholar: por padrão, toda a *string* será buscada por completo em todo o documento. Isso só é alterado quando o usuário define algum campo de dados na busca avançada, no qual o termo deve ser buscado.

VILAS: por padrão, toda a *string* será buscada por completo em todo o documento. Isso só é alterado quando o usuário define algum campo de dados no qual o termo deve ser buscado.

- **Busca no resumo:**

ACM: na busca avançada, o usuário pode escolher o campo de dados *abstract* para um termo ou grupo de termos. Pode também definir o campo de dados diretamente na caixa de buscas, utilizando a sintaxe de busca *recordAbstract* :.

IEEE: na busca avançada ou na busca por comando, o usuário pode escolher o campo de dados *abstract* para um termo ou grupo de termos. Pode também definir o campo de dados diretamente na caixa de buscas, utilizando a sintaxe de busca "*Abstract*" :.

Google Scholar: o sistema não apresenta este campo de busca.

VILAS: a definição do campo de dados *abstract* pode ocorrer tanto para toda a *string* ou para um termo ou grupo específico. No momento da tradução da *string*, para os sistemas que não apresentam esse campo de dados, ele é omitido, permanecendo o restante da *string* definida.

- **Busca no título:**

ACM: na busca avançada, o usuário pode escolher o campo de dados *title* para um termo ou grupo de termos. Pode também definir o campo de dados diretamente na caixa de buscas, utilizando a sintaxe de busca *acmdlTitle* :. IEEE: na busca avançada ou na busca por comando, o usuário pode escolher o campo de dados *title* para um termo ou grupo de termos. Pode também definir o campo de dados diretamente na caixa de buscas, utilizando a sintaxe de busca "*DocumentTitle*" :.

Google Scholar: na busca avançada, o usuário pode selecionar o campo de dados *title*, clicando na opção *in the title of the article*. Pode também definir o campo de dados diretamente na caixa de buscas, utilizando a sintaxe de busca *allintitle* :.

VILAS: a definição do campo de dados *title* pode ocorrer tanto para toda a *string* ou para um termo ou grupo específico.

Tem-se, então, na linguagem VILAS a capacidade de representação de uma *string* de busca a partir da linguagem de busca acadêmica, de modo que essa mesma *string* pode também ser representada pelos sistemas de busca acadêmica da ACM, IEEE e Google Scholar. Assim, conclui-se que é possível a tradução desta representação para as outras sem perdas na estruturação dos termos e combinações. Contudo, isso é válido somente quando se tem uma *string* de busca construída a partir dos elementos básicos definidos na linguagem de busca acadêmica. Logo, as particularidades de cada sistema não são possíveis de representação. Ou seja, se um sistema possui outros operadores lógicos disponíveis para definição da *string* de busca, esses operadores não serão incluídos na tradução, uma vez que não estão disponíveis na representação visual. Por fim, considera-se a linguagem VILAS, apresentada aqui, como sendo uma versão inicial proposta, a partir da qual poderão ser adicionados mais elementos e possibilidades de construção de *strings*.

Capítulo 5

A Ferramenta VILAS-Web

Para a linguagem visual VILAS foi implementada uma ferramenta *web*¹, sendo possível a interação com a interface para definição de uma *string* de busca e avaliação preliminar da linguagem visual proposta. Uma vez que a interface de usuário para recuperação de informação ajuda o usuário a expressar a necessidade de informação, formular *strings*, selecionar fontes de informação, entender resultados de busca e observar o progresso da busca [2], o objetivo da ferramenta é possibilitar ao usuário a expressão do seu desejo de informação na definição da *string* de busca com o uso da linguagem VILAS.

A ferramenta foi implementada para permitir o uso real da linguagem visual proposta, com o objetivo de avaliar a experiência dos usuários com ela. A proposta é que a partir de termos inseridos pelo usuário, ele tenha a liberdade de manipulá-los, criando grupos conectados pelos operadores lógicos (AND e OR), ao mesmo tempo em que tem a percepção da dimensão de pertencimento dos elementos dentro de cada sub-grupo, onde a *string* completa é vista como o todo. Tendo na linguagem proposta uma representação visual da *string* de busca, tem-se na ferramenta uma interface interativa que implementa essa linguagem.

VILAS é uma linguagem para busca acadêmica com abstrações visuais de elementos básicos do contexto de busca, não estando diretamente associada a nenhum outro sistema de busca existente. Além disso, outras abstrações da linguagem de busca acadêmica, apresentadas em sistemas como as bibliotecas digitais da ACM, IEEE ou Google Scholar, possuem estes mesmo elementos básicos. Desse modo, uma *string* representada em VILAS é flexível e adaptável para outras representações que compartilham os mesmos elementos. Assim, na ferramenta VILAS-Web, são implementadas não apenas funcionalidades para manipulação e edição de uma *string* de busca, como também são implementadas funcionalidades para tradução da *string* de busca para outras re-

¹<http://homepages.dcc.ufmg.br/pricilarr/vilas-web/>

apresentações. Tendo na tradução da *string* de busca a eliminação da necessidade do usuário conhecer a fundo a sintaxe de variados sistemas de busca. Ou seja, a partir da tradução, espera-se que os usuários tenham que definir apenas uma *string* de busca a partir da ferramenta VILAS-Web e então solicitar ao sistema a tradução da *string* para a sintaxe de algum dos sistemas de busca disponíveis para essa ação, de modo que o usuário não tenha nenhum esforço para definir essa *string* novamente.

5.1 Interface

Observa-se no comportamento dos usuários de interface *web* em geral, a necessidade de explorar a interface e as opções que estão disponíveis nela. Tal comportamento não é diferente em interfaces de busca, onde o primeiro instinto dos usuários é o de procurar por uma caixa de busca, digitar alguma palavra e realizar uma busca. Considerando isso, a interface da ferramenta VILAS-Web é organizada de forma semelhante a interface inicial de sistemas de busca. As cores e formas aplicadas na interface buscam neutralizar os estímulos enviados pelos elementos da interface, de modo a garantir que o destaque seja a parte central da interface, onde será exibida a *string* de buscas baseada na linguagem VILAS. Na Figura 5.1 pode ser visualizado o mapa da interface, com detalhamento de cada signo e funcionalidades imediante depois.



Figura 5.1. Mapa da interface da ferramenta VILAS-Web

1. **Menu de ajuda:** as funções de definição da *string* de busca estão disponíveis na tela inicial. Logo, o menu se dedica a listar páginas de ajuda e explicação para o

usuário. Na Figura 5.2 tem-se um fragmento das quatro páginas listadas no menu, a partir das quais o usuário pode aprender sobre a linguagem (VILAS), sobre os elementos e sua representação na linguagem (Elements), como se orientar na interface (Interface Map) e ainda aprender sobre como interagir com a ferramenta (Tutorial).



Figura 5.2. Fragmento das páginas de ajuda em VILAS-Web

2. **Campo de inserção de palavras:** no contexto da busca, este será o signo da interface que chamará mais a atenção do usuário. Contudo, a ferramenta permite apenas a inserção de uma palavra por vez. Na Figura 5.3 pode-se observar o comportamento do signo de inserção de palavras.



Figura 5.3. Inserção de palavras em VILAS-Web

3. **Campo de inserção de expressões:** este campo é similar ao anterior, com a diferença de que mais de uma palavra pode ser inserida, de modo a configurar uma expressão. Conforme apresentado no capítulo sobre a linguagem VILAS e os elementos da mesma, uma expressão compreende palavras que na busca são inseridas entre aspas duplas, indicando ao sistema de busca que se deseja resultados que retornem aquelas palavras naquela exata sequência. Na Figura 5.4 pode-se observar o comportamento do signo de inserção de expressões.



Figura 5.4. Inserção de expressões em VILAS-Web

4. **Botão de confirmar inserção:** depois de inserir uma palavra ou uma expressão (o usuário pode inserir apenas um por vez), e clicar no botão **Insert**, o termo de busca é adicionado à área de edição visual da *string* de busca, discutida no item 5. Ressalta-se que um caminho alternativo ao de clicar no botão **Insert** é o de apertar a tecla **Enter** no teclado, obtendo-se o mesmo resultado.
5. **Área de edição visual da *string* de busca:** ao inserir palavras ou expressões nos campos destinados a isso, a área de edição começa a ser populada, como pode ser observado na Figura 5.5. Os termos inseridos aparecem dispostos à esquerda da área e em cor preta, indicando que ainda não foram utilizados para a composição da *string* de busca. Tendo termos nesta área, é permitida a manipulação dos mesmo conforme o usuário deseje, de modo a construir a *string* de busca. A medida que os termos são agrupados, as cores dos mesmos são alteradas e a *string* começa a ser estruturada. Quando um termo é excluído de um grupo, o mesmo aparece na lateral direita dessa área, em oposição aos termos que ainda não foram inseridos na busca.

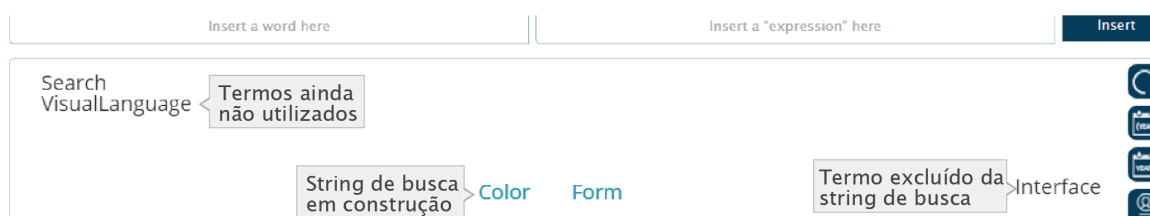


Figura 5.5. Área de edição visual da *string* de busca em VILAS-Web

6. **Área de exibição da *string* de busca:** a versão atual da *string* de busca é construída a partir da *string* visual sendo definida. A exibição da *string* genérica, apresentada na Figura 5.6, ocorre de forma dinâmica e em tempo real, refletindo todas as alterações que o usuário faz na representação visual.
7. **Botões para tradução da *string* de busca:** a partir do momento em que se tem uma *string* de busca construída, por mais simples que seja, é possível sua tradução para os sistemas de busca da ACM, IEEE e Google Scholar. Ao clicar

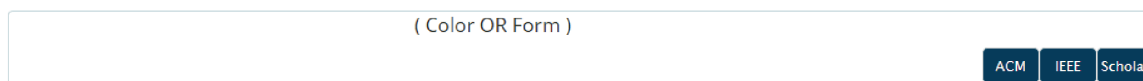


Figura 5.6. Área de exibição da *string* de busca genérica em VILAS-Web

nos botões ACM e IEEE o usuário é encaminhado para uma nova página, onde a *string* de busca é aplicada ao sistema em questão. A tradução da *string* para o Google Scholar ocorre de forma diferente, em razão de limitações da combinação de operadores lógicos e grupos, assim em casos de *strings* mais complexas, uma mesma *string* gera em um conjunto de *strings*, exibido em uma *popup* na interface da ferramenta VILAS-Web.

8. **Botão para limpar a busca:** ao clicar neste botão, tanto a área de exibição visual da *string*, quanto a área de exibição da *string* textual são reinicializadas. Ou seja, a qualquer momento, o usuário pode decidir limpar a *string* de busca sendo definida, colocando a interface em um estado inicial.
9. **Conjunto de botões para refinamento da busca:** as opções de refinamento da busca disponibilizadas na interface e apresentadas na Figura 5.7 são: limitação de ano inicial e final, nome de autor e fonte de publicação do documento. Cada opção pode ser definida uma vez, sendo válida para a *string* completa. Entretanto, o usuário pode optar por alterar ou não inserir valor no campo em questão. Essas opções de refinamento são vistas como um adicional, que restringem os resultados recuperados com base em alguma delimitação definida pelo usuário ou uma informação específica ao contexto de publicação científica do documento.

5.2 Cenário de Uso

O modelo de interação com a ferramenta VILAS-Web é baseado na manipulação direta de objetos e preenchimento de campos específicos. Conforme apresentado anteriormente, a interface é composta da tela de construção da *string* de busca e páginas de ajuda para uso da linguagem e da ferramenta. Portanto, aqui será exemplificado como ocorre a interação para a definição de uma *string* de buscas até o momento de realização da busca, considerando-se um cenário de uso como exemplo de aplicação.

John resolveu fazer seu projeto de conclusão de curso sobre segurança e privacidade em redes sociais. Mas ao pesquisar um pouco sobre o assunto ele descobriu que existem vários tipos de redes sociais, definindo como de seu interesse quatro tipos de

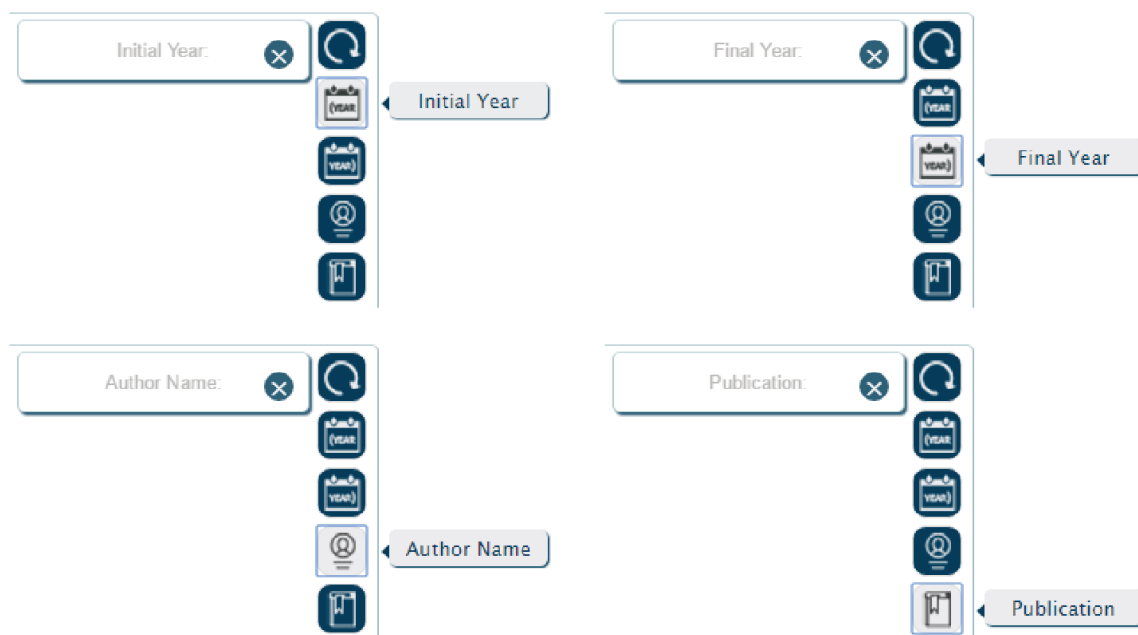


Figura 5.7. Conjunto de botões para refinamento da busca em VILAS-Web

rede social: *comunidade, compartilhamento multimídia, profissional e educacional*. Além disso, ele percebeu que nem sempre os artigos tratam de *segurança e privacidade*, podendo tratar de um ou do outro apenas. Ao fazer sua primeira tentativa de busca, John não obteve bons resultados e concluiu que tinha alguma coisa errada na forma como ele estava escrevendo a string. Ele resolveu então utilizar a linguagem VILAS para construir a string de busca podendo visualizar as relações entre cada palavra que ele inseria, para garantir que a string estaria conforme ele queria. Tendo a string, John iria buscar em vários sites de busca, o que seria facilitado pela ferramenta VILAS-Web, pois a string gerada já seria traduzida para a sintaxe das bibliotecas digitais que ele iria usar.

Inicialmente, John digita as palavras ou expressões nos campos específicos, inserindo um termo por vez. Sempre que termina de digitar um termo, ele clica no botão **Insert** ou aperta a tecla **Enter** no teclado. John resolveu inserir todas as palavras de interesse antes de construir a string, para garantir que não iria se esquecer de nenhuma. Ele as inseriu em inglês para achar os artigos internacionais no tema. Assim, após inserir as palavras, John viu que elas apareceram do lado esquerda da tela (Figura 5.8).

A medida que termos são inseridos, eles passam a ser exibidos na lateral esquerda da área de edição visual da string de busca. Ressalta-se que apenas a inserção de termos na interface não significa que o termo compõe a busca, isso só ocorrerá no momento

em que esse termo passar a fazer parte de um grupo.

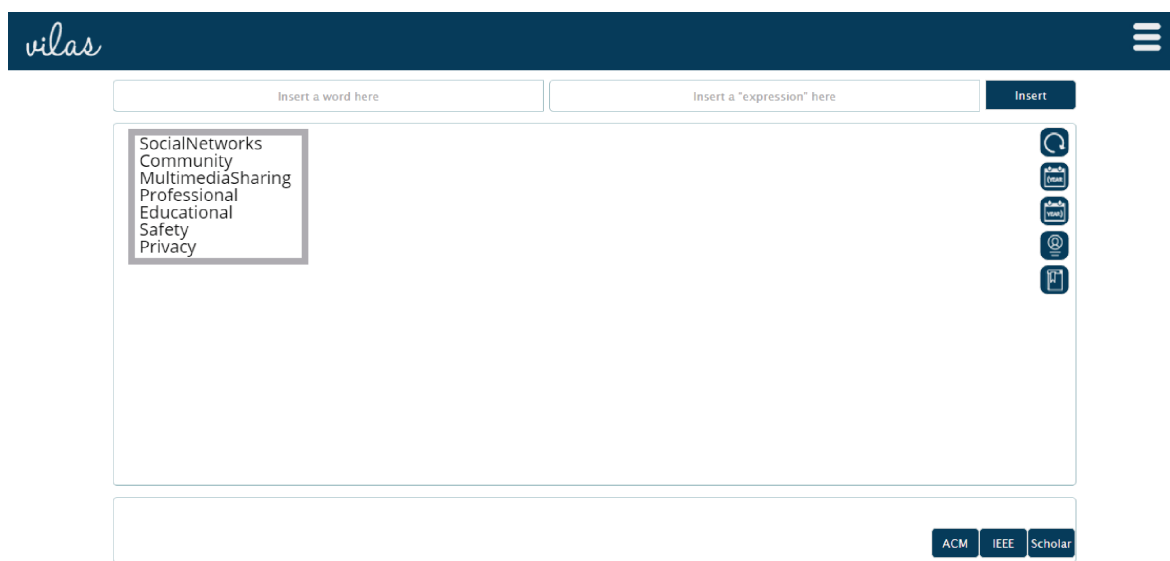


Figura 5.8. Inserção de termos na interface

Continuando a definir a string de busca, John deseja que os resultados tratem sobre **segurança e privacidade**, sendo que os documentos recuperados podem ter os dois termos ou apenas um. Assim, John conclui que ele precisa fazer um agrupamento utilizando o operador lógico OR. A busca pelos quatro tipos de rede social selecionadas pelo usuário também refletem um agrupamento OR.

A ação correspondente à intenção do usuário, exemplificada na Figura 5.9, é a de clicar e arrastar um dos termos até que ele colida à esquerda do outro termo. Os dois termos passam então a configurar um grupo, cujo conector é o operador OR e a representação na linguagem VILAS são os elementos dispostos horizontalmente e nas mesmas cores. A representação do agrupamento dos dois conjuntos de palavras pode ser visualizada na Figura 5.10, bem como a *string* de busca textual gerada.



Figura 5.9. Agrupamento de dois termos com o operador lógico OR

Agora, John quer que os resultados obrigatoriamente sejam sobre **redes sociais de comunidade, compartilhamento multimídia, profissional ou educacional**, e

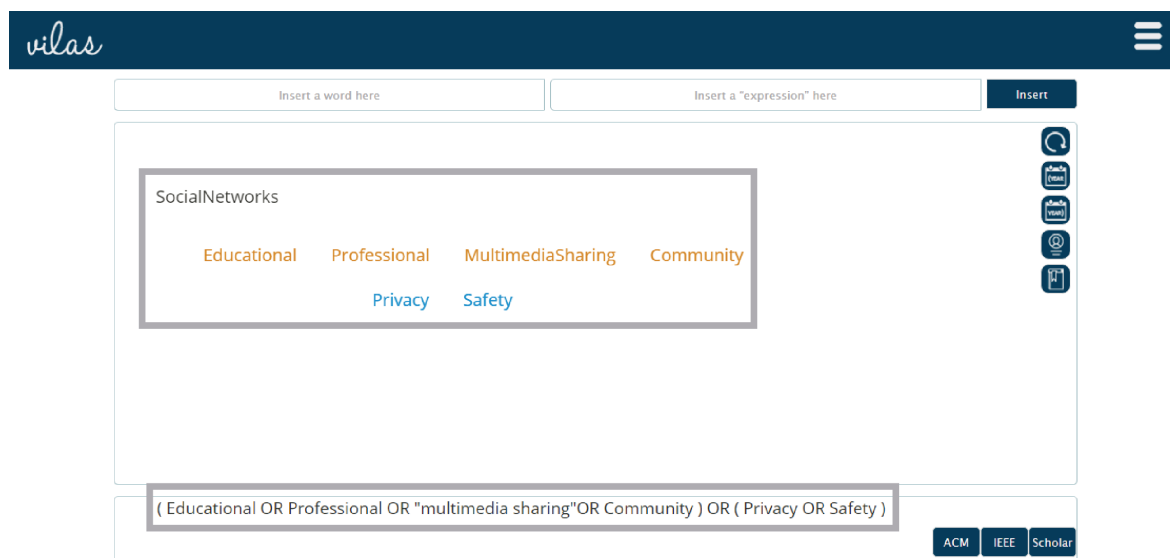


Figura 5.10. Representação da *string* de busca depois da ação de agrupamento OR

que também tratem sobre *segurança* ou *privacidade*. Ele conclui que precisa fazer um agrupamento AND com estes termos. Para isso, ele primeiro arrastou a expressão até o grupo com os tipos de redes. Depois, ele arrastou o outro grupo até o novo grupo formado.

A ação correspondente na Figura 5.11 é semelhante à anterior, com a diferença de que um dos termos deve ser arrastado até colidir no lado direito do outro termo. As representações visual e textual da *string* de busca podem ser visualizadas na Figura 5.12. Um agrupamento AND na linguagem VILAS dispõe os termos alinhados verticalmente e com cores diferentes.



Figura 5.11. Agrupamento de dois termos com o operador lógico AND

Até esse momento John já tem uma *string* de busca relativamente complexa definida e pronta para realizar buscas. Contudo, ele vê que ainda podem ser realizadas alterações na composição da *string*, como por exemplo a definição de um novo rela-

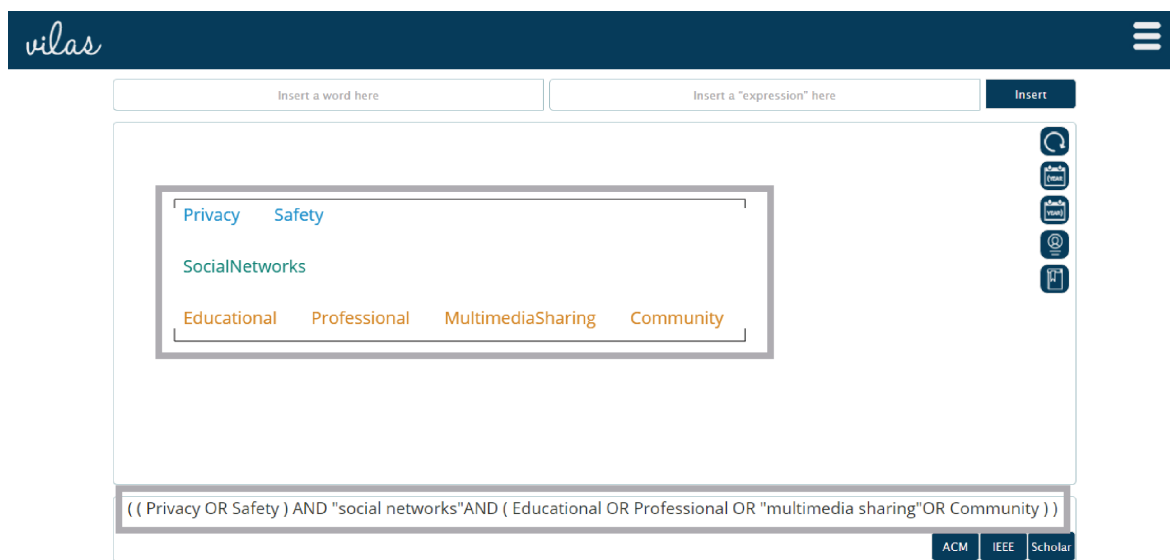


Figura 5.12. Representação da *string* de busca depois do ação de agrupamento AND

cionamento entre os termos, exclusão de um termo pertencente a string de busca ou adição de novos termos. Analisando a string, John decide excluir a expressão sobre compartilhamento multimídia.

Na Figura 5.13 é ilustrada a ação de exclusão de termo da *string* de busca. A ação a ser executada é um clique duplo no termo correspondente, excluindo-o da *string* e um novo clique duplo, excluindo-o do espaço de definição. As representações visual e textual após a exclusão do termo são apresentadas na Figura 5.14.



Figura 5.13. Exclusão de um termo da *string* de busca

John não quer que os artigos recuperados em sua busca sejam muito antigos. Por isso, ele decidiu limitar o ano inicial de publicação dos artigos. Deixando o ano final sem limitação, porque ele quer os artigos mais atuais. Ele observa que é possível também definir um nome de autor, mas no momento ele não quer trabalhos de um autor específico. A ferramenta permite também inserir uma fonte de publicação, buscando por exemplo todos os documentos relacionados à sua busca e que foram publicados em

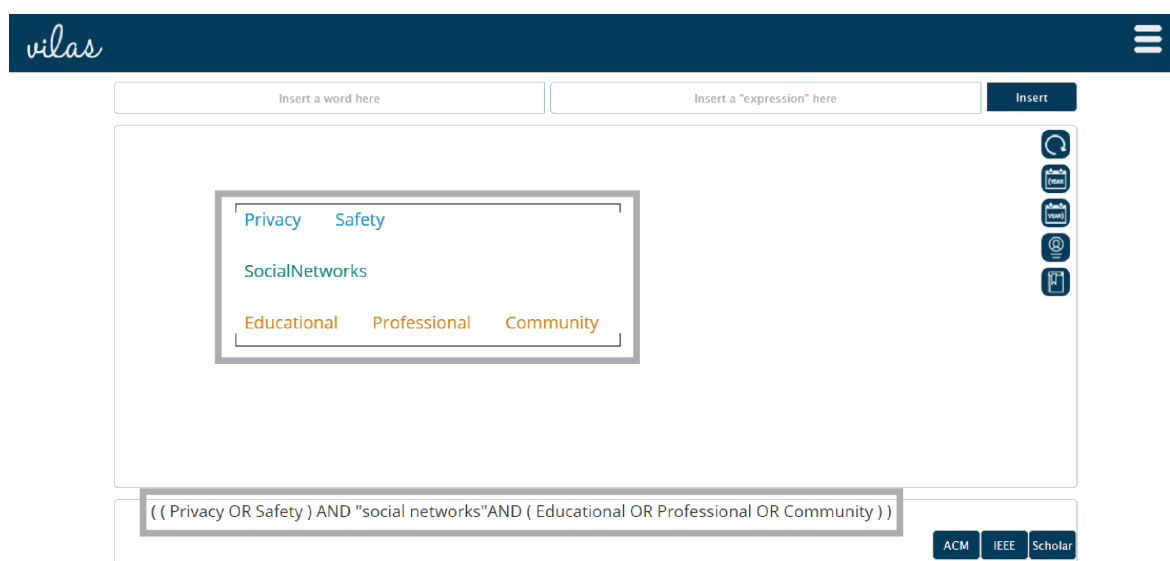


Figura 5.14. Representação da *string* de busca depois da ação de exclusão de um termo

um determinado evento ou por uma determinada instituição. Mas John preferiu, pelo menos inicialmente, buscar por artigos de qualquer publicação.

O processo de interação com qualquer uma das opções de refinamento é semelhante, sendo apresentado na Figura 5.15 o processo para limitação do ano inicial de publicação dos documentos. A ação é caracterizada pelo clique no botão de ano inicial, inserção do ano desejado e fechamento do campo aberto.

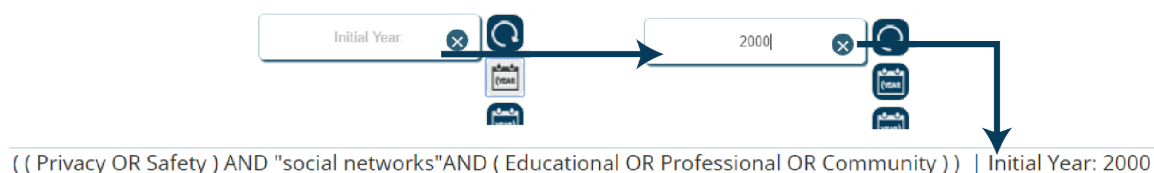


Figura 5.15. Adição do filtro de ano inicial na *string* de busca

Por fim, tendo definido a string de busca (Figura 5.16 e compreendido o relacionamento entre cada termo, ou seja, qual será a configuração dos resultados recuperados perante à busca definida, John decidiu realizar a busca em sistemas de especializados.

Logo em frente à string de busca textual, John viu três botões com os nomes de sistemas de busca acadêmica. Primeiro, ele clicou no botão da ACM, sendo redirecionado para uma página de busca da biblioteca digital da ACM com uma busca já definida e com resultados recuperados (Figura 5.17). Ele observou que a string de busca apresentada era um pouco diferente da que estava apresentada na ferramenta VILAS-Web,

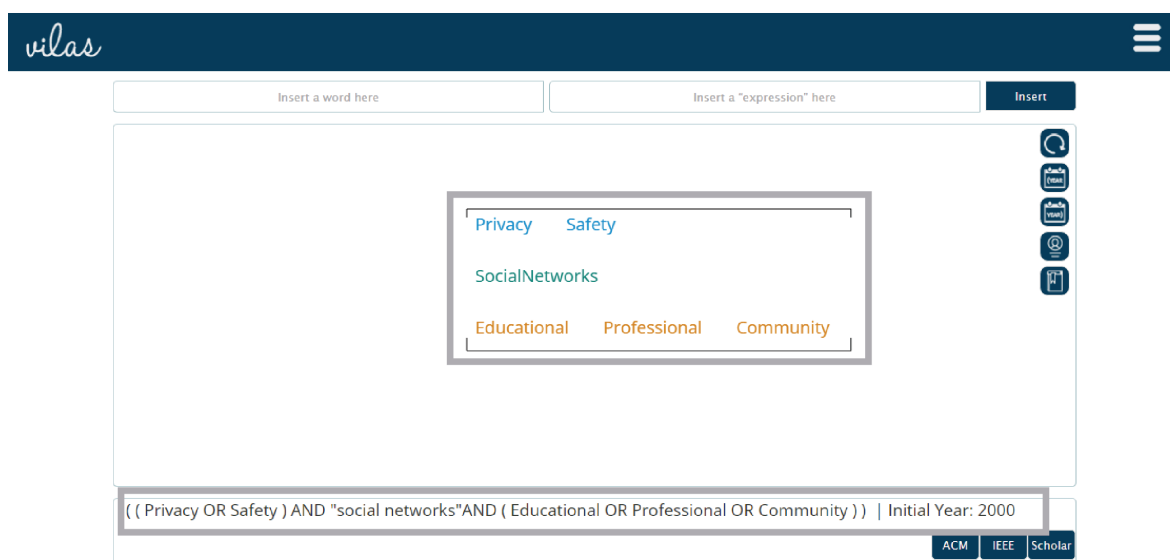


Figura 5.16. Representação final da *string* de busca

o que significava que a *string* tinha sido traduzida para se adequar ao sistema escolhido. Depois, John clicou no botão para a busca na biblioteca digital da IEEE. Novamente, uma nova página foi aberta com uma busca definida naquele sistema (Figura 5.18). Por fim, John clicou no botão Scholar, e se deparou com uma caixa popup. Ao analisar, ele observou que a caixa continha várias *strings* de busca (Figura 5.19), que correspondiam à *string* de busca que ele definiu, mas na sintaxe do Google Scholar (Figura 5.20). John então se concentrou em analisar os documentos recuperados para lê-los e escrever seu trabalho.

Algumas estruturas de agrupamento não são possíveis ou não ocorrem da mesma forma em todos os três sistemas para os quais a ferramenta VILAS-Web fornece suporte. Assim, a *string* de busca definida visualmente e apresentada em formato textual na ferramenta é traduzida e escrita de acordo com a sintaxe específica de cada sistema. No caso do Google Scholar, um nível de agrupamento maior, requer a fatoração da *string* em várias partes, de modo a contemplar todas as possibilidades de resultados.

5.3 Limitações da Ferramenta

Considerando a proposta completa da linguagem VILAS e as variadas bibliotecas digitais que não foram contempladas nesta pesquisa, observa-se que a ferramenta não se apresenta completa, no sentido de que diversas funcionalidades e suporte a bibliotecas podem ser incluídas. Contudo, tem-se aqui uma primeira versão funcional da

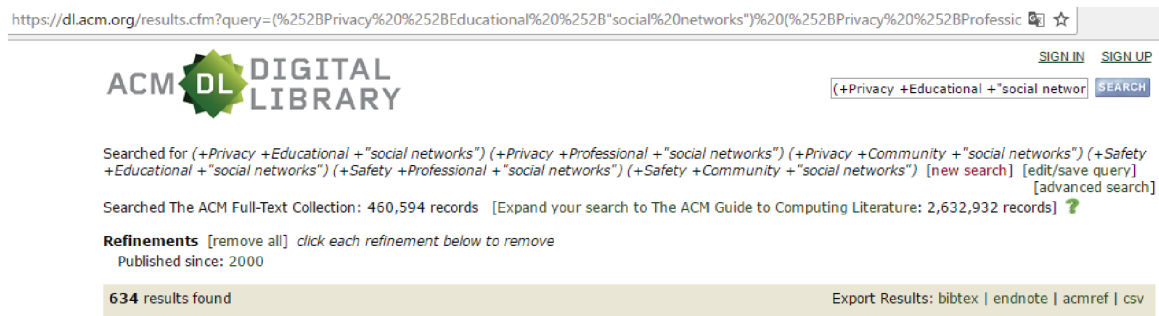


Figura 5.17. *String* de busca traduzida para a biblioteca da ACM

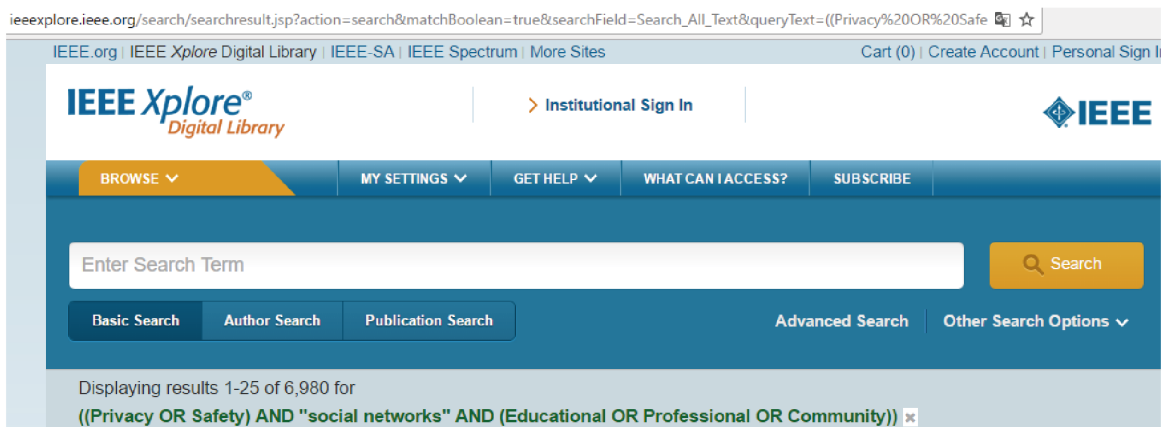


Figura 5.18. *String* de busca traduzida para a biblioteca da IEEE

ferramenta, na qual os usuários podem definir *strings* e obtê-las traduzidas para três diferentes sistemas de busca.

Destaca-se também que a definição do local (ex.: título, resumo) onde a busca por determinado termo ocorrerá não foi implementada. Conforme observado nos sistemas de busca, na ACM e na IEEE essa definição ocorre para cada palavra ou expressão na *string* de busca, de modo que em uma mesma busca, uma palavra possa ser buscada apenas no título, outra apenas no resumo e as demais em qualquer parte do documento. Já no Google Scholar, essa definição ocorre uma vez para toda a *string*. Dessa forma, dada a complexidade necessária na tradução das *strings* levando-se em consideração estes aspectos de cada sistema, essa funcionalidade não foi adicionada nesta versão da ferramenta VILAS-Web, podendo ser contemplada em futuras versões da ferramenta.

Por razões de compatibilidade com a sintaxe de busca de diferentes sistemas, algumas restrições foram aplicadas quanto a liberdade para inserção de termos, agrupamento por meio de operadores lógicos e aplicação de filtros. Vale ressaltar que a

```

Strings generated for use in Scholar:
https://scholar.google.com/scholar?
as_q=Privacy+Safety+SocialNetworks+Educational+Professional+Community&as_ylo=2000
*https://scholar.google.com/scholar?as_q=Privacy+Educational&as_epq=Social+Networks&as_ylo=2000
*https://scholar.google.com/scholar?as_q=Safety+Educational&as_epq=Social+Networks&as_ylo=2000
*https://scholar.google.com/scholar?as_q=Privacy+Professional&as_epq=Social+Networks&as_ylo=2000
*https://scholar.google.com/scholar?as_q=Safety+Professional&as_epq=Social+Networks&as_ylo=2000
*https://scholar.google.com/scholar?as_q=Privacy+Community&as_epq=Social+Networks&as_ylo=2000
*https://scholar.google.com/scholar?as_q=Safety+Community&as_epq=Social+Networks&as_ylo=2000

```

Figura 5.19. Listagem das *strings* de busca traduzidas para o Google Scholar

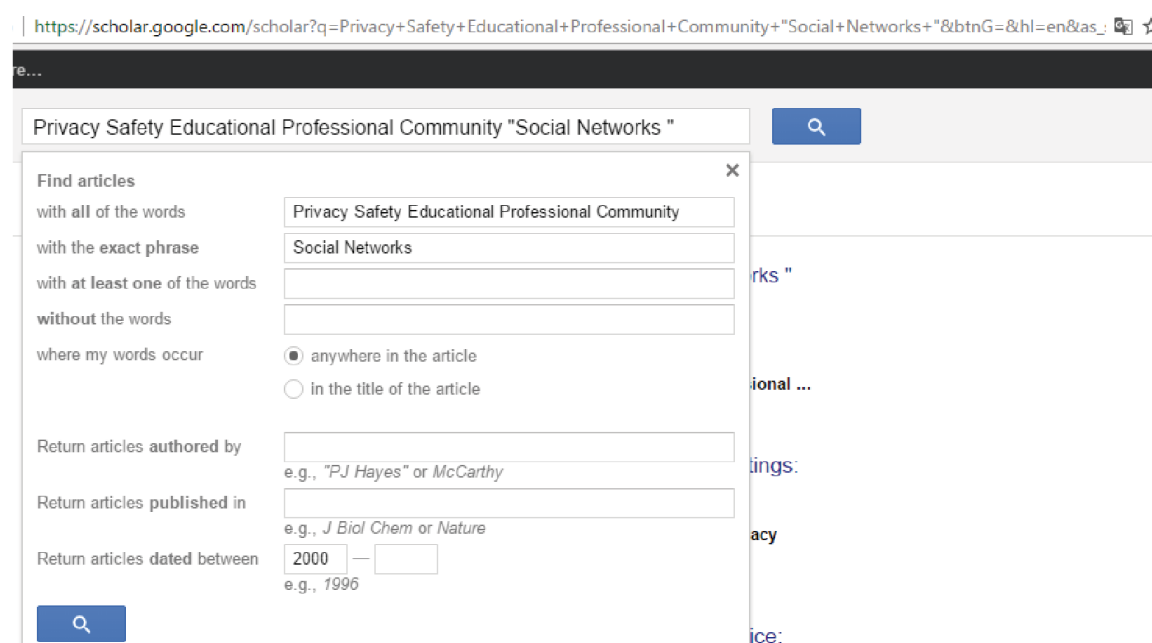


Figura 5.20. Representação de uma das *strings* de busca traduzida para o Google Scholar

limitação na quantidade de termos tem também o intuito de levar o usuário a inserir em sua busca apenas as palavras-chave que considera mais importantes para o resultado que deseja. As restrições definidas na versão atual da ferramenta VILAS-Web são descritas na Tabela 5.1.

Tabela 5.1. Restrições para definição da *string*

Restrição	Valor	Descrição
Quantidade de termos	15	Não podem ser inseridos mais do que 15 termos (palavras) na <i>string</i> de buscas.
Nível de hierarquia	3	Um grupo pode estar dentro de outro grupo concatenado por um operador lógico diferente. Como por exemplo ((term1 OR term2) AND term3). Ou ainda, (((term1 OR term2) AND term3) OR term4). Contudo, o mesmo não é válido para um quarto grupo. Como por exemplo (((term1 OR term2) AND term3) OR term4) AND term5)
Refinamento <i>Author</i>	1	Pode ser usado apenas uma vez e é aplicado em toda a <i>string</i> . Contudo, o usuário pode inserir mais de um nome de autor, desde que em um mesmo termo.
Refinamento <i>Publication</i>	1	Pode ser usado apenas uma vez e é aplicado em toda a <i>string</i> .
Refinamento <i>Year</i>	1	Pode ser usado apenas uma vez e é aplicado em toda a <i>string</i> . Contudo, o usuário pode inserir um ano inicial e um ano final.

Capítulo 6

Avaliação da Linguagem VILAS

A avaliação do linguagem visual se deu através de duas perspectivas diferentes: avaliação da representação visual escolhida (signos estáticos) e avaliação através da ferramenta. Desta forma, foi possível identificar não somente questões relativas à usabilidade da interface desenvolvida para uso da linguagem, como também foi possível avaliar a compreensão da linguagem por parte dos usuários. Assim, nas Seções 6.1 e 6.2 são discutidas, respectivamente, as avaliações da representação de VILAS e da ferramenta para construção da representação visual a partir da linguagem.

6.1 Avaliação da Representação Visual

O objetivo da avaliação da representação de VILAS foi o de identificar a compreensibilidade da representação visual de uma *string* de busca por parte dos usuários. A representação da linguagem reflete a linguagem em si, onde os elementos e os relacionamentos entre eles são definidos. Em se tratando de uma linguagem visual, é necessário não apenas se apoiar em princípios de *design* para construção de representações visuais minimalistas e adequadas, como também é essencial investigar se o público para quem essa visualização se destina é capaz de perceber os estímulos enviados pela representação, assimilando-os e como resultado compreendendo o que o *designer* quis exprimir por meio deles.

A avaliação consistiu em uma discussão informal e individual com quatro usuários identificados como pertencentes ao público a que a linguagem se destina. Uma vez sendo uma avaliação informal, não houve preocupação em realizar os testes em ambiente controlado, preocupando-se apenas em encontrar um ambiente sem barulhos ou interrupção. Os testes com cada participante ocorreram no local mais apropriado para o usuário, contando com o uso de um computador para apresentação das imagens e gra-

vação de áudio da discussão. Todos os participantes possuem conhecimentos em lógica booleana e são estudantes de pós-graduação de diferentes campos da Ciência da Computação (Banco de Dados, Engenharia de Software, Interação Humano-Computador e Mineração de Dados). Foi solicitado aos participantes da pesquisa que comentassem a respeito do seu entendimento a partir da observação de uma série de imagens de representações visuais de *strings* de busca, com o uso da linguagem VILAS. Durante a realização do teste com cada um dos participantes foi seguido um roteiro de perguntas, incluindo diferentes representações e objetivos, deixando claro ao participante que quaisquer interpretações que ele obtivesse estariam corretas para o contexto da pesquisa, e instigando-o a expor mais sua opinião nos momentos em que se percebia que o mesmo não compreendia a questão ou a visualização, ou se mostrava em dúvida quanto a algum dos estímulos visuais recebidos. O roteiro completo da avaliação e o termo de consentimento para participação da pesquisa, encontram-se no Apêndice A. Posteriormente aos testes com os quatro participantes, os dados obtidos foram analisados a fim de identificar os pontos onde a compreensão dos usuários foi contrária ou semelhante ao exposto na representação da linguagem, e também as similaridades e dissimilaridades na percepção dos diferentes usuários.

Inicialmente foi brevemente explicado aos participantes o que é uma *string* de busca e como ela é formada. Assim, no primeiro contato com a representação visual os participantes já sabiam que se tratava de uma *string*, e a partir daí foi discutido o significado dos estímulos visuais de cores, posicionamento e agrupamento de palavras. Foi consenso entre os participantes a aleatoriedade dos estímulos enviados quanto a representação de termos ligados com o operador OR e termos ligados com o operador AND. Um dos participantes disse que “*precisava ver mais imagens para encontrar um padrão do que é AND e OR*”. De forma geral, nos primeiros contatos com a representação, todos os participantes se mostraram confusos e tiveram interpretações inconsistentes quanto aos dois tipos de agrupamento. Contudo, era esperado que a representação dos agrupamentos gerasse dúvidas, considerando que essas representações fora do contexto de interação são arbitrárias e ambas podem representar qualquer um dos operadores. Além disso, as palavras utilizadas na representação chamaram mais a atenção dos participantes do que o previsto, de modo que muitas vezes a interpretação dos operadores lógicos da *string* de busca foi feita mais em função do significado da palavra e da expectativa do participante sobre a *string* que faria sentido, do que na própria representação visual. Assim, em alguns casos, um determinado posicionamento era ora interpretado como OR, e pouco depois como AND pelo mesmo participante.

As diferentes cores e tonalidades de cores entre os termos causou dificuldades de compreensão para todos os participantes em algum momento. Um dos participantes

demonstrou incerteza quanto ao significado das cores, afirmando que “*uma palavra está mais clara que a outra, mas não consigo entender porquê*”. Enquanto outros comentaram que as cores e tonalidades estavam confundindo muito e que não conseguia identificar nada nas cores. Outro participante buscou relacionar termos da mesma cor e mesma tonalidade como parte de um sub-grupo, dentre outros termos com a mesma cor, mas tonalidades diferentes. Ao final da discussão, um dos participantes que anteriormente não havia compreendido as cores, comentou que “*cada grupo de cores está dentro um parênteses*”, contudo ainda sem entender exatamente o que isso representava.

O colchete horizontal usado para enfatizar a existência de um operador AND entre os termos, foi percebido por um dos participantes como um grupo de palavras de maior importância na busca, ou seja, que possuía prioridade sobre as demais palavras naquela mesma busca. Outro participante entendeu que o colchete indicava a separação de *strings*. Enquanto que dentre os outros dois participantes, um não compreendeu o propósito da linha colchete e o outro o compreendeu como sendo a indicação do operador AND. A representação de uma expressão, que consiste na junção das palavras que a compõem, foi compreendida pela maioria dos participantes. As explicações que deram para a representação foi de que “*porque quero que venha exatamente assim; como se fosse aspas*” ou “*porque é um nome composto*”. Ao final do teste, cada participante foi capaz de estruturar a representação visual da *string* em uma representação *textual*, eliminando inconsistências encontradas no decorrer da discussão e desconsiderando os estímulos que não havia compreendido, como no caso das diferenças nas cores e tonalidades.

6.1.1 Discussão

Como resultado desta avaliação, algumas representações foram mantidas, enquanto outras foram repensadas. No caso da diferenciação de tonalidades de cores, até este momento eram sete diferentes tonalidades, definidas de forma aleatória, sendo que na versão atual da linguagem, todos os termos ligados com um operador OR possuem a mesma tonalidade e cor, havendo um espaço um pouco maior entre eles do que anteriormente. Ambas as representações podem ser vistas no conjunto de Figuras 6.1. Ainda dentro do campo das cores, a troca de cores entre elementos conectados por AND (Figuras 6.2), antes mais sutil tornou-se um pouco mais expressiva, de modo a enfatizar mais essa troca.

Outra alteração realizada na representação da *string* foi quanto ao colchete horizontal que enfatiza a existência de um operador AND conectando os elementos. Uma



Figura 6.1. Representações do agrupamento OR

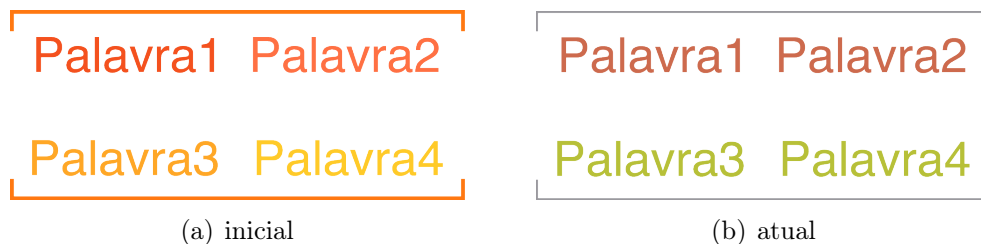


Figura 6.2. Representações do agrupamento AND

vez que foi observado que esses colchetes chamavam demasiadamente a atenção dos participantes, a cor deles foi alterada para uma pigmentação mais neutra, ao invés de uma cor definida pela combinação das cores dos elementos internos, como era antes. A intenção é apenas de auxiliar o usuário da linguagem a identificar por meio da completude da representação, a existência de uma estrutura ali dentro, e que possui grande importância dentro da *string* de busca sendo definida, onde essa importância se exprime em que todos aqueles termos unitários ou grupos de termos estarão presentes no resultado da busca.

Quanto à representação visual dos operadores AND e OR, vale ressaltar que os participantes tiveram que interpretá-la fora de um contexto de uso, ou do apoio de uma ferramenta. Assim, a expectativa é de que quando um usuário utilizar a ferramenta para construir uma *string*, ele terá acesso também aos signos dinâmicos, ou seja, interagirá com a ferramenta percebendo o comportamento de cada ação tomada na representação visual, podendo então concluir o que cada organização representa. A partir desta avaliação e dos ajustes realizados na representação de VILAS, observa-se que embora a representação permita ao usuário expressar *strings* de busca acadêmica, ela envolve um custo de aprendizado das representações da estrutura da *string* por parte dos usuários. No entanto, como são poucas representações a serem aprendidas, e a interação com a ferramenta pode facilitar este aprendizado, acredita-se que o custo para o usuário não será muito alto, tornando ainda viável o uso da linguagem visual.

6.2 Avaliação com uso do Protótipo

Após a revisão da linguagem com base nos resultados da avaliação da representação dos elementos, foi feita a avaliação da linguagem visual utilizando o protótipo desenvolvido. O objetivo desta avaliação foi de investigar o uso da linguagem visual proposta em uma interface interativa. Permitindo assim analisar a percepção dos usuários quanto a representação visual e o comportamento deles durante a interação, de modo a identificar o quanto a interação auxilia no entendimento e uso da linguagem.

Observando o usuário final da linguagem, foram selecionados participantes de diferentes campos da Ciência da Computação, que possuem conhecimentos em lógica booleana. Além disso, nenhum dos participantes tinha um conhecimento prévio sobre a linguagem, sendo sua primeira interação com a linguagem no momento da avaliação. Participaram da avaliação seis estudantes de pós-graduação em Ciência da Computação, com pesquisas nas áreas de Aprendizado de Máquina, Interação Humano-Computador e Recuperação de Informação. No Apêndice B é apresentado o roteiro utilizado na condução desta avaliação e o modelo do termo de consentimento para realização deste teste. Os testes foram conduzidos no laboratório de testes com usuários do DCC/UFMG. O laboratório conta com dois ambientes, em que um é equipado com um computador com câmera integrada para a interação do usuário, tendo sido instalada uma câmera atrás do usuário para captar seus movimentos em direção a tela. O outro ambiente da sala é isolado do primeiro e equipado com duas outras telas para observação. Além disso, foi utilizado o software Morae para gravação da interface da ferramenta no momento da interação do usuário com a mesma. Participaram dois avaliadores, a autora que conduziu a avaliação e um aluno de iniciação científica que auxiliou-a na avaliação, no papel de observador.

Os testes ocorreram individualmente com cada um dos participantes. Antes de iniciar a interação com o sistema, foi feita uma breve entrevista com o participante, a respeito de seu conhecimento quanto à buscas acadêmicas, a frequência com que é feita e qual o comportamento para a construção de uma *string*. Quatro dos participantes disseram fazer buscas por artigos acadêmicos pelo menos uma vez por semana, um deles realiza buscas quase todos os dias e o outro disse realizar buscas pelo menos uma vez por mês. Foi também perguntado aos participantes se eles utilizam o modo de busca avançada quando realizam buscas acadêmicas, sendo que apenas um respondeu que não. Os outros, que utilizam o modo de busca avançada disseram que utilizam apenas depois de realizar uma busca simples e desejam refinar a busca, definindo melhor os termos e combinações entre eles. Além disso, as ferramentas/repositórios utilizados pelos usuários são as mesmas que serviram de objeto de estudo neste trabalho (ACM,

IEEE e Google Scholar), sendo que além dessas, os participantes citaram a DBLP, HCIBib, Springer e Portal de Periódicos da Capes.

Depois foi solicitado ao participante que interagisse com a ferramenta para a definição de uma *string* de busca, sendo que para isso deveria utilizar os operadores lógicos OR e AND. Para tanto, explicou-se ao participante quais os elementos compõem uma *string* de busca acadêmica e deixou-se que ele utilizasse o sistema. Algumas das *strings* de busca definidas pelos participantes da avaliação são apresentadas no conjunto de Figuras 6.3. Um pesquisador acompanhou a interação do participante com a ferramenta o tempo todo, respondendo a dúvidas que não foram contempladas na seção de ajuda e instigando o participante a pensar no que ele desejava fazer quando se percebia que o mesmo estava perdido, enquanto outro pesquisador ficou do outro lado da sala observando a interação do usuário e fazendo anotações. Durante a interação do usuário seguiu-se o protocolo de testes *Think-Aloud*, no qual o usuário é estimulado a falar a respeito das ações que executava, qual o resultado era esperado e os motivos que o levavam a chegar a determinada ação, comentando também quando gostaria de fazer alguma coisa mas percebia não ser possível. Ao final, foi feita uma entrevista com cada participante, investigando a compreensão do participante quanto à representação da *string* de buscas e também sobre sua opinião a respeito da interação com a ferramenta.

Ao começar a interagir com a ferramenta VILAS-Web, os usuários primeiro observaram a interface a fim de ter uma visão geral dos signos que a compõem. Três dos participantes foram primeiro ao menu de ajuda para descobrir o que era a linguagem e como poderia definir uma *string* de buscas, contudo um desses participantes começou a ler e resolveu já definir sua busca. Os demais participantes foram diretamente para os campos de inserção de texto. Contudo, ao se depararem com um campo para inserir uma palavra e outro para inserir uma expressão, os participantes ficaram confusos sobre o que inserir e onde inserir. Por exemplo, um dos participantes digitou uma *string* composta de duas palavras e um operador lógico no campo de expressões. Outro participante, entendeu a expressão como sendo o operador lógico AND ou OR. Nestes dois casos, os usuários perceberam que a ação não era possível, de modo que um deles procurou por mais informações nas páginas de ajuda e o outro perguntou diretamente ao pesquisador que acompanhava a avaliação. Além disso, ainda no contexto da expressão, no momento em que inseriam uma expressão e viam as palavras se juntarem, dois participantes pensaram ser um erro, sendo que um deles pensou ter digitado sem o espaço e o outro pensou ser um erro da ferramenta.

Considerando que o caminho mais natural ao interagir com uma nova interface é primeiro explorá-la, a atenção dada inicialmente ao tutorial foi mínima. Assim, em especial os participantes que não acessaram o tutorial, ao começar a interagir com

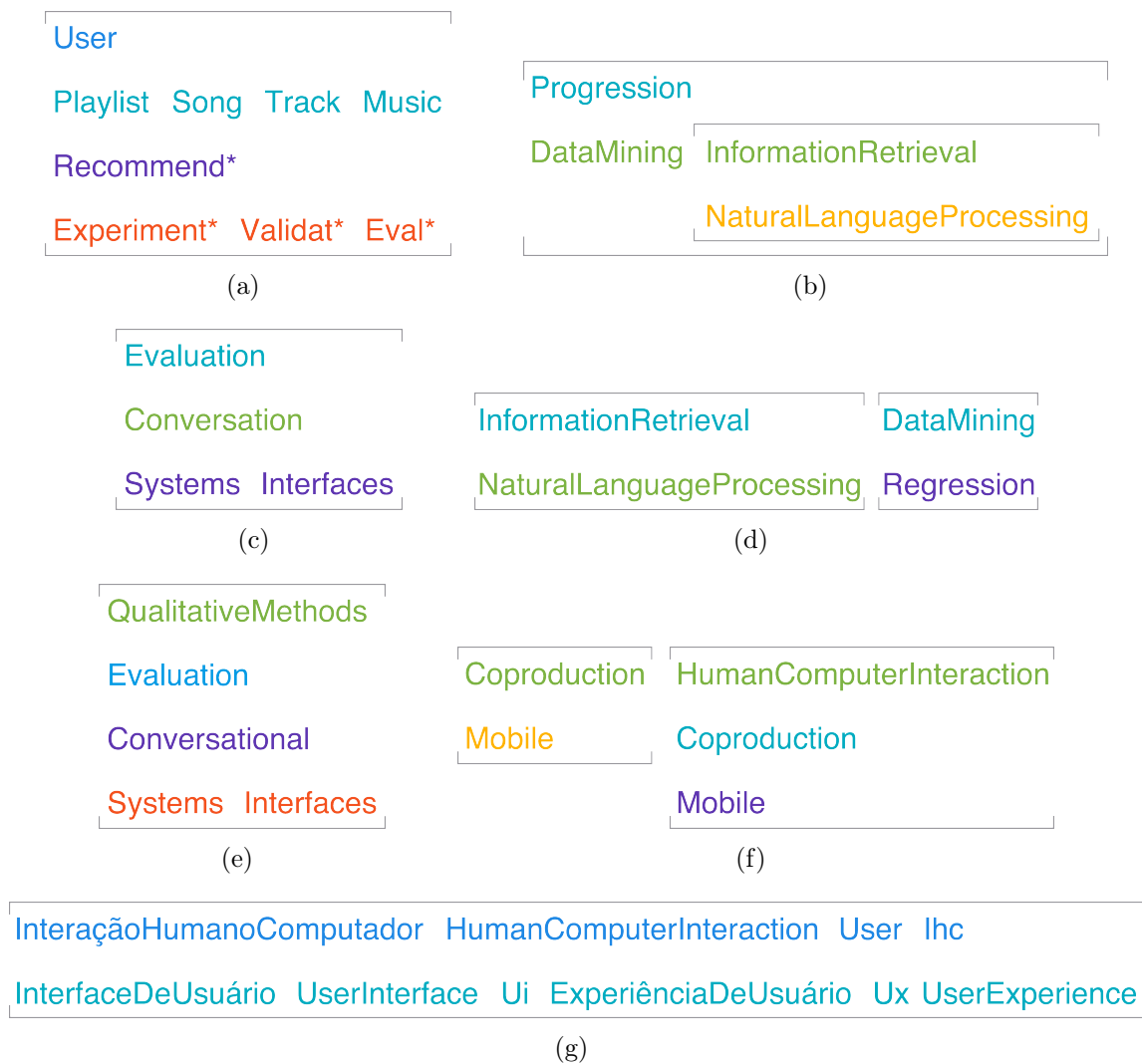


Figura 6.3. *Strings* de busca definidas pelos participantes da avaliação

a ferramenta, ficaram surpresos ao descobrir que as palavras e expressões inseridas podiam ser arrastadas na tela. Dessa forma, foi usada a estratégia de tentativa e erro para descobrir como construir grupos OR e AND. Um dos usuários tentou selecionar os termos exibidos na tela, clicando em um ponto da tela e arrastando. Outro usuário deu vários cliques no termos, tentando descobrir alguma coisa. Por fim, por conta própria, leitura do tutorial ou explicação do pesquisador, todos os participantes descobriram a possibilidade de clicar e arrastar termos na tela. O agrupamento de palavras e expressões com os operados lógicos foi o que gerou mais dúvidas nos participantes. Mais uma vez, alguns utilizaram da estratégia de tentativa e erro, arrastando palavras e observando que quando haviam colisão com outro termo, ocorria um agrupamento. Entretanto, todos os participantes recorreram ao tutorial para confirmar ou descobrir qual o movimento correspondia a ação de adicionar um operador lógico OR e qual

adicionava o operador lógico AND. Três participantes consultaram o tutorial mais de uma vez, por terem dificuldades em lembrar qual ação era de qual operador. Outro participante removia os termos da *string* quando percebia, através da indicação do sistema quanto ao tipo de agrupamento, que tinha feito o agrupamento contrário ao que desejava e fazia nova tentativa.

As opções para refinamento da busca (limitação de ano, nome de autor e fonte de publicação) são comuns em sistemas de busca, o que levou os participantes a compreenderem imediatamente a função desses operadores para a construção da *string* de busca. Contudo, a maioria dos participantes se mostrou confusa quanto ao que significava *publisher*, o que foi explicado pelo pesquisador que acompanhava os testes, uma vez que a explicação sobre essas opções não existia no tutorial de uso da ferramenta. Um dos participantes quis remover um dos filtros que tinha inserido e a princípio não soube como fazer, consultando o tutorial para descobrir, já que ele se lembrou de ter lido algo sobre isso.

No início da avaliação foi explicado aos participantes as finalidades da ferramenta VILAS-Web, que iam desde o suporte à construção da *string* de busca, à tradução desta *string* para os sistemas de busca acadêmica da ACM, IEEE e Google Scholar. Assim, quando se sentiram satisfeitos com a *string* de busca definida, todos os usuários clicaram em um dos botões para ver como ocorreria essa tradução. Observou-se que os participantes se sentiram satisfeitos ao perceberem que a *string* de busca que definiram de modo visual foi representada no sistema de busca, em especial na ACM, onde a *string* foi modificada para se adequar à sintaxe daquele sistema. A tradução da *string* de busca para o Google Scholar não foi trivial para a maioria dos participantes, pois apenas um demonstrou ter consciência de que aquela *string* que havia definido não se encaixava na sintaxe do sistema. Contudo, depois de entenderem o motivo de uma única *string* gerar várias *strings* no Google Scholar, os participantes ficaram satisfeitos e acharam curioso o fato de que caso quisessem buscar aquela *string* exata no sistema, e se tivessem que fazer isso sem a ferramenta, teriam que construir todas as alternativas possíveis da *string*.

Depois da interação com a ferramenta, foram feitas algumas perguntas sobre essa interação. Inicialmente, procurou-se identificar se os participantes conseguiam visualizar a *string* de busca visual como sendo a representação da mesma busca na *string* de busca textual. Os participantes disseram ter compreendido isso, mas não em um primeiro momento. A apresentação da *string* de busca em formato textual auxiliou na percepção do que significava a representação, uma vez que, segundo um dos participantes “*se eu mudar aqui (área de exibição da string de busca) vai mudar aqui embaixo. Consigo entender que é a mesma coisa, mas eu preciso interpretar*”. Outro

participante destacou a necessidade de aprender um conceito novo, “*quando comecei a usar fiquei procurando cadê o OR e o AND. Só que aí depois eu vi que era tudo visual. Eu gostei, só que não foi trivial para mim entender*”. Quanto à representação visual de uma *string* de busca, os participantes consideram uma boa representação e que ajudaria as pessoas a entenderem a estrutura da busca. Um dos participantes acredita que a linguagem deve ser focada em outras áreas, “*não focado na área da computação, mas em outras áreas, facilitaria sim*”. Outro participante comentou que ajudaria se tivesse uma indicação escrita dos operadores lógicos AND e OR, “*acho que só com as cores e a ordem não é tão fácil entender*”.

Quanto à ferramenta VILAS-Web e considerando que a versão utilizada para testes era uma versão beta, os participantes demonstraram terem gostado. Além disso, sentiram-se bem confortáveis utilizando a ferramenta e familiarizados com os signos apresentados. Destaca-se aqui que os usuários compreenderam o signo de reinicializar a *string* como uma forma de atualizar a *string* visual, ou de atualizar a *string* textual. Apesar disso, a ação realizada pelo botão é a de limpar toda a tela, o que, segundo um dos participantes o deixaria bem frustrado se ele tivesse realizado a ação quando estava no processo de construção de uma *string* esperando por outro resultado. Dado isso, considera-se que em uma nova versão da ferramenta, este signo deva ser alterado para facilitar a sua interpretação como limpeza da tela, do que de atualização. As maiores dificuldades que os participantes apresentaram foram relacionadas à montagem da *string* de buscas. A manipulação direta dos termos na tela para gerar grupos OR e AND gerou muitas dúvidas, levando alguns dos usuários a recorrerem ao tutorial mais de uma vez, de modo a lembrar como fazer cada combinação.

Por fim, dada a experiência de interação com a ferramenta, foi perguntando aos participantes quais sugestões eles teriam para melhorar a interação e também a linguagem visual. Dentre as sugestões dos usuários, a mais recorrente foi a necessidade de editar a *string* textual, seja na área de exibição da *string* genérica ou no momento da inserção de termos, podendo inserir diretamente uma *string* naquele espaço. Além disso, os participantes demonstram desejar mais poder de manipulação, podendo editar palavras diretamente na *string* visual, ou organizar grupos apenas clicando e arrastando um subgrupo completo. Algumas das falas dos usuários referentes a essa sugestão foram: “*acho que inserir a string completa, podendo ou inserir as palavras e o OR e o AND, ou ir inserindo só as palavras*”; “*poder pré-montar o que eu quero e se eu quiser mudar, só clicar e arrastar*”; “*talvez ter um lugar aqui (na interface) com um botõzinho com um OR e um AND, e aí você arrasta*”; “*ter alguma informação antes de fazer o AND e o OR, talvez ter uma caixinha que aparece e aí pergunta AND ou OR*”.

6.2.1 Discussão

A avaliação com o uso do protótipo permitiu a identificação de importantes pontos a serem revistos nas próximas versões da linguagem e da ferramenta. Onde se destaca a necessidade de procurar outras formas de realizar o agrupamento com os operadores AND e OR. A organização visual de um grupo AND e de um grupo OR fica clara para os usuários quando eles estão interagindo com a ferramenta, e ao entenderem isso é observado que os usuários não apresentam dificuldades em aceitar e compreender o sentido dessa organização. Entretanto, a manipulação dos termos para formar esses agrupamentos, em que para um deve haver uma colisão à direita e para o outro uma colisão à esquerda, é completamente arbitrária. E possivelmente, em razão de não haver um significado por trás dessa manipulação, os usuários não conseguiram internalizá-la, apresentando dificuldades para se lembrar mesmo depois de já ter o conhecimento sobre. Além disso, apesar de no momento que o agrupamento é feito ser exibida uma mensagem com o tipo e a *string* de busca textual atualizada, é preciso encontrar uma maneira de indicar ao usuário o tipo de agrupamento antes dele ser realizado. Assim, consideram-se inúmeras outras possibilidades de definir essa manipulação, onde o modo de manipulação atual ainda possa ser usado, como uma forma mais rápida, mas existindo outras formas de manipulação, em especial para usuários iniciantes.

Outros pontos a serem investigados na ferramenta são relativos ao campo para inserção de termos. A partir da avaliação, observou-se que os usuários desejavam inserir a *string* de busca completa ou parcial, e não apenas um termo de cada vez. Além disso, a existência de um campo separado para expressão deixou os usuários confusos, dado que para eles uma expressão podia tanto ser as palavras que normalmente ele colocaria entre aspas, quanto a *string* de busca em si. Logo, pretende-se implementar na ferramenta mecanismos para o reconhecimento de uma *string* de busca, gerando em consequência a representação visual específica, de modo que o usuário não precise construir os relacionamentos de forma totalmente visual. Tendo também a possibilidade de edição da *string* textual, onde a edição é refletida diretamente na *string* visual. Em adicional, pretende-se também permitir maior controle na *string* visual, criando maneiras de edição dos termos e sub-grupos. Buscando assim dar mais poder de manipulação da *string* ao usuário, de modo que essa manipulação ocorra em qualquer momento da construção ou da representação da *string* de busca.

As opções de refinamento disponíveis até o momento são apenas três. Contudo, foi observado nas bibliotecas digitais analisadas neste trabalho que existem várias outras opções que podem ser implementadas. Porém, a implementação dessas outras opções vai de encontro a falta de uma linguagem de busca acadêmica ampla e generalizada,

em que as opções de refinamento não são as mesmas em todos os sistemas. Assim, este é um ponto que necessita de uma análise profunda, buscando encontrar um caminho através do qual o usuário possa ter acesso a maior quantidade de opções de refinamento.

Além de aspectos a serem revistos e melhorados na linguagem e na ferramenta, tem-se também a possibilidade de tradução da *string* de buscas para uma quantidade maior de sistemas de busca acadêmica. De modo que, ao oferecer mais opções de sistemas de busca ao usuário, ele certamente encontre os sistemas que costuma utilizar, e o ganho ao utilizar a ferramenta VILAS-Web seja maior, no sentido de não precisar escrever *strings* para cada um desses outros sistemas.

Enfim, todos os pontos que foram identificados nesta avaliação e que possam ser identificados em avaliações futuras, devem ser considerados para construção de uma versão mais robusta da ferramenta VILAS-Web. Possibilitando assim avaliações mais extensas da linguagem VILAS, investigando o uso da linguagem visual em detrimento da linguagem textual.

6.3 Avaliação das Dimensões Cognitivas

Uma dimensão cognitiva de uma notação é “uma característica da maneira como a informação é estruturada e representada, compartilhada por muitas notações de diferentes tipos e por sua interação com a arquitetura cognitiva humana” [18]. O *framework* de Dimensões Cognitivas para Notação (CDN) foi utilizado para avaliar a notação da linguagem VILAS, buscando evidenciar como as pessoas usam a notação visual definida. Tendo nas CDN’s uma ferramenta para avaliação de interfaces de usuário que consistem de elementos gráficos completamente novos [Blackwell & Green], possibilitando a descoberta de problemas de usabilidade dentro dessas interfaces.

A aplicação do *framework* CDN é baseada na execução dos seguintes passos: descrição do sistema; definição do que o usuário pode fazer com a notação; definição de tarefas representativas a serem realizadas; investigação da aplicação das dimensões cognitivas para cada uma das tarefas; comparação do perfil observado com o perfil ideal para aquele tipo de atividade. Sendo que a avaliação consiste na observação de variadas dimensões, listadas a seguir de acordo com o apresentado em [18].

- Viscosidade: indica resistência a mudanças, procurando identificar qual a quantidade de trabalho que o usuário deve fazer para efetuar uma pequena alteração.
- Visibilidade: facilidade em visualizar componentes. Quanto mais longa é a pesquisa necessária para encontrar algo, menos visível e satisfatória é a experiência.

- Compromisso Precoce: restrições na ordem de fazer as coisas. Uma vez que o usuário não deve tomar decisões antes da informação necessária para ela, deve-se garantir que a ordem das coisas é adequada.
- Dependências Ocultas: links importantes entre entidades devem ser visíveis, de modo que o usuário consiga acessá-los e saiba de sua existência.
- Expressividade: a finalidade de uma entidade deve ser facilmente deduzida. Além disso, dois elementos com diferentes finalidades não devem parecer os mesmos, a fim de garantir que os usuários não tenham interpretações errôneas ao identificar similaridades.
- Propensão a Erro: a notação deve ajudar ao usuário a evitar erros ou deslizes, e o sistema deve fornecer proteção para evitar erros.
- Abstração: indica o número de abstrações que um usuário precisa dominar antes que ele seja capaz de usar o sistema.
- Notação Secundária: são informações adicionais à sintaxe formal, utilizadas para complementar um elemento (ex.: cores), mas que não mudam o significado do elemento.
- Proximidade de Mapeamento: identifica o quão próxima é a notação do resultado que ela descreve.
- Consistência: a notação deve utilizar padrões consistentes, permitindo que o usuário reconheça símbolos familiares.
- Dispersividade: indica a verbosidade da linguagem, ou seja, o número de símbolos ou entidades requeridas para expressar um significado. Quanto menor o espaço usado para transmitir uma mensagem, melhor.
- Operações Mentais Difíceis: identifica a demanda de recursos cognitivos, julgando a complexidade da notação para o usuário.
- Provisoriedade: define o grau de comprometimento para as ações. Indicando a possibilidade do usuário fazer seleções indicativas antes de fazer escolhas definitivas.
- Avaliação Progressiva: permite aos usuários parar no meio do trabalho para verificar o que está sendo produzido.

6.3.1 Discussão

Como objeto de avaliação a partir da CDN, tem-se a notação VILAS e a ferramenta VILAS-Web. VILAS é descrita como uma linguagem visual para construção de *strings* de busca acadêmica. VILAS-Web, por sua vez, é descrita como uma ferramenta que implementa VILAS e fornece um ambiente interativo para construção de uma *string* de busca a partir da notação apresentada. Tendo em VILAS o artefato de estudo, foi feita uma análise embasada nas dimensões cognitivas apresentadas em [18], a fim de identificar se essas são adequadamente apoiadas pelo artefato de informação. Sendo a tarefa de análise o uso de VILAS para a definição de uma *string* de busca. Para cada uma das dimensões cognitivas mencionadas anteriormente, investigou-se aspectos relacionados à notação VILAS e à interação com VILAS-Web, concluindo com uma discussão sobre o que seria a interação desejada para tal dimensão.

- Viscosidade:

Notação VILAS: a construção de uma *string* a partir da notação definida em VILAS é feita através de combinações de elementos básicos – termos e operadores lógicos. Sendo a reconstrução de uma *string* feita pelo rearranjar desses elementos dentro da *string*, seja alterando ou excluindo combinações.

Interação com VILAS-Web: sempre que se insere uma nova palavra ou expressão, bem como se exclui algo da *string* de busca, a representação visual é reestruturada, de forma a apresentar a nova configuração da *string*. Em VILAS-Web, a adição de termos em uma *string* ocorre através da manipulação direta, a partir de onde são definidas regras para inserção destes termos dentro da *string*. Entretanto, é permitida apenas a exclusão de um termo por vez da *string* e a reestruturação ocorre somente no sentido de se adequar a ausência deste termo. Casos em que o usuário deseja excluir um grupo inteiro da *string* exigirão uma quantidade de trabalho proporcional a quantidade de termos a serem excluídos.

Cenário ideal: deve ser possível ao usuário acrescentar ou remover termos, expressões e/ou grupos em uma *string*, sempre que ele deseja. Além disso, a reestruturação de uma *string* deve ser tão simples quanto arrastar e combinar termos.

- Visibilidade:

Notação VILAS: uma *string* de busca completa pode ter de 2 a n termos, dentre os quais podem haver uma variedade de subgrupos. Assim, por mais que a notação utilize de cores e posicionamento de elementos para representar as

combinações existentes, não é possível garantir que o usuário visualize facilmente a composição da *string*. Isto é, como os termos da *string* se relacionam entre si através de combinações lógicas, e como é definida a estruturação hierárquica da *string*.

Interação com VILAS-Web: tendo uma *string* de busca com cerca de quinze termos, e uma estrutura composta de até três níveis aninhados, considera-se que a representação seja clara e compreensível, sendo possível identificar grupos e relacionamentos entre os elementos da *string*.

Cenário ideal: a construção de uma *string* com quantos termos o usuário deseja, tem como consequência um custo mais alto para o usuário perceber o que a *string* representa. Sendo que a compreensão da *string* se torna mais custosa não apenas através da notação VILAS, mas também em sua forma textual. Dessa forma, a limitação da quantidade de termos procura garantir que a *string* seja facilmente compreendida pelo usuário.

- Compromisso Precoce:

Notação VILAS: a notação fornece um conjunto de elementos através dos quais o usuário é livre para construir *strings* de busca com os termos desejados, realizando combinações entre eles.

Interação com VILAS-Web: é fácil tirar ou colocar termos em estruturas que estejam no mais alto nível de hierarquia, mas não é possível acrescentar termos em estruturas aninhadas. Tendo, portanto, a construção da *string* visual a partir de sua parte mais interna, até sua parte mais externa. O que leva a necessidade do usuário definir todos os termos e combinações mais internas da sua *string*, antes de definir os mais externos.

Cenário ideal: o usuário deve ser capaz de inserir, editar ou excluir termos da *string* de busca sempre que desejar. Além disso, é desejado que o usuário tenha liberdade para manipular a *string* de busca como um todo, definindo combinações ou reajustando elas com facilidade.

- Dependências Ocultas:

Notação VILAS: os subgrupos que compõem uma *string* não apresentam um fechamento completo. Isto é, não existe um elemento como uma contêiner visível que englobe todos os elementos de um subgrupo. Contudo, através de fatores como cores e organização espacial, é possível identificar subgrupos dentro da *string*.

Interação com VILAS-Web: em adicional à apresentação da notação, a ferramenta fornece ao usuário a possibilidade de definir filtros e opções de refinamento da busca. Sendo que estes, apesar de não serem representados visualmente na notação, são facilmente encontrados na interface de VILAS-Web e apresentados na *string* textual.

Cenário ideal: é desejado que a notação contemple também elementos para representação de filtros e opções de refinamento de busca. De modo que o usuário não dependa de uma ferramenta interativa para construir uma *string* de busca que utilize filtros e refinamento.

- Expressividade:

Notação VILAS: o principal objetivo da representação visual definida em VILAS, é o de fornecer uma visão mais perceptível da estrutura de uma *string* de busca. Tendo nos elementos de combinação, os elementos que exercem maior influência sobre a busca a ser realizada, a representação deles foi definida de forma a ressaltar as diferenças e possibilitar a identificação de cada representação rapidamente. No entanto, a organização proposta para OR (elementos de uma mesma cor em uma mesma linha) e para o AND (elementos de cor diferentes em linhas diferentes), embora tenha um *rationale* por trás, nem sempre é facilmente percebida pelos usuários.

Interação com VILAS-Web: em adicional à apresentação da notação, a ferramenta implementa o processo de interação para construção da *string*. Onde a interação ocorre através da manipulação direta dos termos, realizando combinações. Entretanto, por mais que as regras para construção dos agrupamentos OR (arrastar um termo à direita do outro) e AND (arrastar um termo à esquerda do outro) sejam diferentes, elas são semelhantes e arbitrárias, sem possuir um significado intrínseco.

Cenário ideal: no que diz respeito à notação, deve-se apenas atentar para o uso de cores contrastantes e que destaquem as diferenças a que se propõem destacar. Tendo como desejado na manipulação direta, regras mais relacionadas à forma como os termos se organizam em cada tipo de agrupamento. Onde agrupamentos OR sejam definidos a partir de colisões horizontais, enquanto que colisões verticais definiriam agrupamentos AND.

- Propensão a Erro:

Notação VILAS: ainda que a notação tente ressaltar com o uso de cores e organização espacial o significado dos elementos de combinação da linguagem,

ela não previne erros ou deslizes do usuário, no sentido de que o usuário pode ter interpretações errôneas a respeito do significado de alguma representação.

Interação com VILAS-Web: através da redundância de representações (visual e textual) a ferramenta se propõe a evitar interpretações errôneas. Contudo, combinações são realizadas imediatamente a identificação de uma colisão, levando o usuário a executar ações extras caso tenha cometido um erro.

Cenário ideal: além da representação textual da linguagem da *string* sendo construída, é desejado que o usuário receba um *feedback* associado à manipulação direta. Onde a combinação não ocorra tão logo haja colisão, mas que antes seja informado ao usuário qual combinação ele está prestes a fazer, e dê tempo ao usuário para decidir se ele deseja mesmo realizar essa ação.

- Abstração:

Notação VILAS: considerando o contexto de uma *string* de busca acadêmica, a compreensão, mesmo que superficial, de como funciona a lógica *booleana* é requerida para a compreensão de VILAS. Onde os recursos visuais buscam enviar estímulos ao olhar do usuário, para que ele possa perceber com mais clareza o que a sintaxe da *string* está representando. Por exemplo, uma combinação OR indica que pelo menos um dos elementos do grupo deve ser retornado na busca. Mas quando se vê isso sob a ótica da representação visual, tem-se um grupo de elementos organizados em uma mesma linha e com uma mesma cor. Tendo aqui, como abstração pretendida, a de que, sendo esta a representação de uma combinação OR, a busca deve retornar pelo menos um dos elementos dessa linha. Assim, o usuário deve compreender as representações propostas para os elementos de combinação para usar VILAS e ferramentas que implementem seu uso.

Interação com VILAS-Web: idem à Notação VILAS.

Cenário ideal: espera-se que a representação atual da *string* de busca em VILAS seja facilmente interpretada pelo usuário. Porém, o ideal é não exigir ou exigir o mínimo de conhecimento de lógica *booleana*. Sendo assim, é desejado que existam exemplos e *feedbacks* que auxiliem o usuário a construir uma *string* de busca e compreender o significado e composição da *string*.

- Notação Secundária:

Notação VILAS: a representação das combinações lógicas OR e AND em uma *string* é feita tanto pela organização espacial dos termos, quanto pelo uso de

cores, de modo que a partir de apenas uma dessas informações já seja possível identificar o significado da representação.

Interação com VILAS-Web: além do uso de cores e diferentes organizações espaciais na representação visual da *string*, a ferramenta VILAS-Web exhibe a representação textual da *string* de busca. A *string* textual é construída automaticamente a partir da representação visual, sem perdas de sentido ou significado. Contudo a notação secundária é apenas de saída, isto é, não é possível definir ou editar uma *string* através da representação textual.

Cenário ideal: tendo como notação secundária a representação textual da *string*, é ideal que seja possível ao usuário a construção de uma *string* por este caminho. De modo que a representação textual reflita e visual, e vice-versa, permitindo que o usuário escolha o caminho pelo qual ele deseja definir sua *string* de busca.

- Proximidade de Mapeamento:

Notação VILAS: atualmente VILAS oferece representações para os elementos de conteúdo e combinação de uma linguagem de busca. Os elementos de conteúdo tem uma representação bem próxima do que representam, uma vez que os termos e expressões de busca são representados por eles mesmos (coloridos e sem espaço no caso da expressão). Entretanto, nos elementos de combinação, o mapeamento não é tão próximo, já que as combinações são representadas visualmente por uma notação definida especificamente no contexto de VILAS.

Interação com VILAS-Web: idem à Notação VILAS.

Cenário ideal: o mapeamento de uma *string* textual para uma *string* visual em VILAS depende de características exclusivas da notação. O que leva a necessidade de avaliações a fim de identificar se a notação de VILAS é tão próxima quanto o esperado do domínio que ela representa.

- Consistência:

Notação VILAS: os termos e expressões sempre são expressos em forma textual e definidos pelo usuário. Um termo representa exatamente uma palavra, enquanto que uma expressão é o equivalente a um conjunto de palavras do mesmo contexto (ex.: interação humano-computador, recuperação de informação), representado na notação como um conjunto de termos agrupados sem espaço e iniciais maiúsculas. No caso dos elementos de combinação, a seleção de cores e a representação

espacial são utilizadas de forma única para cada tipo de combinação (OR ou AND).

Interação com VILAS-Web: idem à Notação VILAS.

Cenário ideal: a notação VILAS apresenta-se consistente ao representar uma *string* de busca.

- Dispersividade:

Notação VILAS: a representação visual da *string* é contida em um conjunto de palavras e expressões, organizados espacialmente a partir de regras pré-definidas pela notação. Assim, por mais que sejam aplicadas diferentes cores e posicionamentos de elementos de acordo com o tipo de combinação, as regras de seleção e posicionamento relativo dos termos são poucas e bem definidas para cada um dos dois grupos – AND e OR.

Interação com VILAS-Web: em adicional à apresentação da notação, a ferramenta permite a definição de filtros e opções de refinamento da busca. Onde esses filtros e opções são aplicados em toda a *string* e necessitam ser definidos apenas uma vez.

Cenário ideal: não sendo possível e nem desejável limitar o tamanho dos termos que o usuário insere em uma *string*, a limitação da quantidade de termos pode ser vista como uma forma de manter a *string* clara e contida em um espaço visível.

- Operações Mentais Difíceis:

Notação VILAS: para *strings* de busca simples (poucos termos e aninhamento de elementos de combinação) acredita-se que o custo cognitivo para entendê-la seja baixo, considerando-se que o usuário esteja familiarizado com a representação visual. Porém, a medida que a *string* de busca se torna maior e mais complexa, a compreensão do que ela representa passa a exigir mais esforço quando observada sob a visão do todo.

Interação com VILAS-Web: em adicional à apresentação da notação, a limitação da quantidade de termos em uma *string* visa garantir que o usuário consiga interpretar o que a *string* representa sem que para isso haja um alto custo cognitivo.

Cenário ideal: além da representação estática da *string* textual, é desejado que o usuário consiga construir uma *string* através deste caminho. Assim, o usuário é livre para escolher o caminho que ele considera mais fácil para definir sua *string* e compreender com mais clareza e facilidade o que essa *string* representa.

- Provisoriedade:

Notação VILAS: a linguagem VILAS busca apoiar a construção de uma *string* de busca, de modo que o usuário possa fazer quantas modificações ele desejar na *string*. Assim, não se tem uma representação visual definitiva de uma *string* de busca em VILAS até que o usuário se decida por isso.

Interação com VILAS-Web: idem à Notação VILAS.

Cenário ideal: em um cenário ideal, mesmo depois de um tempo, o usuário deve ser capaz de reescrever a *string* de busca definida. Ou seja, deve ser possível ao usuário salvar uma *string* de busca para edição futura.

- Avaliação Progressiva:

Notação VILAS: como VILAS representa graficamente os elementos de uma linguagem de busca, a partir do momento que começa a definir uma *string* de busca, o usuário é capaz de visualizar o que está definindo.

Interação com VILAS-Web: o usuário pode gerar suas buscas, ainda que a *string* não esteja finalizada, avaliando o resultado gerado nos mecanismos de busca associados. A partir da sua avaliação dos resultados, pode alterar a *string* sendo montada.

Cenário ideal: permitir a edição futura de uma *string* é desejado para que, a qualquer momento depois de definir uma *string* e utilizá-la, o usuário possa refinar o que definiu e assim obter resultados melhores.

Conclui-se portanto, que a notação definida em VILAS atende às várias dimensões cognitivas listadas anteriormente, estruturando e representando a informação a que se propõem de forma significativa. Contudo, observa-se que apenas a notação estática é insuficiente para a compreensão do usuário perante ao que ela representa, sendo necessário o apoio de um sistema interativo, que em um cenário ideal, atenda a todas as necessidades de estruturação da notação e interação do usuário. Portanto, tendo em vista a ferramenta VILAS-Web, compreende-se que ela ainda não apresenta-se como um cenário ideal para interação com a linguagem VILAS, gerando um esforço cognitivo maior do que o esperado em alguns momentos. Tendo nas dimensões cognitivas avaliadas uma série de aspectos a serem implementados ou melhorados na ferramenta para que ela se aproxime mais de um cenário ideal para interação com linguagem VILAS.

Capítulo 7

Conclusão

Neste trabalho foi apresentada a linguagem visual para buscas acadêmicas VILAS, cujo objetivo é prover uma representação de *strings* de buscas definidas com o uso de operadores lógicos e hierarquia de elementos. Além disso, foi apresentado um protótipo funcional da ferramenta *web* que implementa esta linguagem e que permite não apenas a definição da *string* de busca, como também a tradução desta *string* para os sistemas de busca acadêmica da ACM, IEEE e Google Scholar.

Através da revisão da literatura, foi observado que a maioria dos trabalhos que tratam sobre linguagens visuais para construção de *strings* são antigos e são definidos para um domínio muito específico ou muito abrangente, o que deixa a representação, em geral, mais complexa. Em adicional, estes trabalhos apresentam foco maior na organização da apresentação dos resultados da busca, onde o processo de busca se restringe inicialmente a definição de uma *string* de busca textual. Assim, o presente trabalho oferece contribuição científica, no sentido de avançar a pesquisa tanto na parte de visualização de *strings* de busca, quanto revisitando o uso de uma linguagem visual em interfaces de busca. Ou seja, trazendo uma abordagem de linguagens visuais mais atualizada ao contexto computacional atual e uma proposta para melhorar a compreensão da estruturação de uma *string* de busca através da interação com representações visuais.

A linguagem proposta é uma representação visual de uma *string* de busca, onde as combinações lógicas (OR e AND) entre as palavras inseridas na busca são representadas através da organização espacial dos elementos da busca e cores dos elementos. Provendo assim uma alternativa à visualização da *string* textual, em que os operadores lógicos são inseridos entre as palavras e a hierarquia é representada com o uso de parênteses.

A representação visual definida para VILAS foi resultado de um processo envolvendo várias tentativas de representar uma *string* de busca. Tendo por base os

princípios da Gestalt para construção dessa representação visual, procurou-se manter apenas os elementos essenciais para abstração da *string* de busca, fazendo uso de cores e organização espacial para definição de grupos e similaridades/dissimilaridades entre os elementos representados. Logo, existem outras formas de representação e organização dos elementos que podem também expressar a estrutura de uma *string* de buscas, sendo a apresentada neste trabalho apenas uma delas. Portanto, pretende-se posteriormente investigar a aplicação de variações na representação apresentada, de modo a encontrar aquela que melhor responde as expectativas dos usuários quanto ao modo como a *string* de busca é apresentada, tornando a representação visual o mais intuitiva possível para a maioria dos usuários.

Para implementação da linguagem visual proposta, foi desenvolvida uma ferramenta *web*. Com o uso da ferramenta, é possível ao usuário a definição de uma *string* de busca utilizando a linguagem VILAS e a tradução desta para outros sistemas. Através das avaliações com usuários, observou-se que alguns aspectos relacionados à interação com a ferramenta do sistema também precisam ser repensados, garantindo que os usuários não apresentem dificuldades para utilizar a ferramenta. Nesse ponto, destaca-se a necessidade de alternativas a manipulação direta dos termos da *string* de busca e geração de grupos, interação esta que gerou muitas dúvidas entre os participantes da avaliação. Sendo uma possível solução a criação de um caminho alternativo para realizar o agrupamento, em que ao detectar a colisão de dois termos, o sistema exhibe ao usuário as duas opções de agrupamento a serem realizadas. Outra melhoria a ser realizada é a possibilidade de inserção de uma *string* de busca completa ou parcial em formato textual, de modo que o sistema gere a representação da mesma sem que para isso o usuário precise arrastar elementos e construir grupos. Além de fornecer mais oportunidades de edição da *string* visual, permitindo ao usuário isolar sub-grupos e editar apenas esses elementos internos, ao invés de apenas movimentar o grupo principal. Dando ao usuário maior poder de manipulação, de modo que o usuário possa decidir quando usar o formato textual e quando usar o formato visual.

A tradução da *string* de busca textual gerada a partir da representação visual em VILAS, apresenta-se como uma das vantagens do uso da ferramenta VILAS-Web. Assim, pretende-se expandir essa tradução para uma quantidade maior de sistemas de busca acadêmica, tornando ainda mais vantajoso ao usuário o uso da ferramenta. Fornecendo suporte não somente a bibliotecas digitais relacionadas a Ciência da Computação, mas também a bibliotecas de outras áreas. Possibilitando, tanto o uso da ferramenta e da linguagem por uma grupo maior de usuários, quanto a realização de testes com usuários que não tem um conhecimento prévio de lógica booleana, como no caso de alunos da computação.

Em adicional, outras possibilidades de incremento da ferramenta VILAS-Web se encontram na recomendação de termos de busca e possibilidades de salvar a *string* de busca. De modo que ao receber recomendação de termos, o usuário tenha uma visão mais ampla do que ele deseja buscar e com isso gere uma *string* de busca mais adequada ao tipo de informação que ele deseja. Tendo na possibilidade de salvar a *string* de busca uma forma de alterar a *string* de busca depois de realizar buscas ou de reaproveitá-la para uma nova busca. Além de gerar a partir daí, dados sobre o comportamento de busca do usuário, que podem ser analisados a fim de gerar recomendações mais específicas.

Pretende-se, tanto com a linguagem, quanto com a ferramenta *web*, uma alternativa para a definição de buscas acadêmicas. Apesar do tempo requerido para o aprendizado de uma nova linguagem e representação, a ideia é que o usuário necessite aprender apenas este sistema, eliminando-se a necessidade de entender as particularidades de cada sistema de busca quando desejar definir uma *string* de busca mais complexa. Tornando o processo de definição da *string* de busca mais fácil e menos trabalhoso, e aumentando a capacidade de entendimento do usuário quanto a estrutura da *string* de busca de modo a minimizar interpretações errôneas da compreensão do sistema de busca. Tendo-se por expectativa que a linguagem proposta seja mais simples de aprender e dê ao usuário maior capacidade para expressar seu desejo de informação.

Referências Bibliográficas

- [1] Baeza-Yates, R. & Ribeiro-Neto, B. (2011). *Modern Information Retrieval: The Concepts and Technology Behind Search*. Addison Wesley. ISBN 9780321416919.
- [2] Baeza-Yates, R.; Ribeiro-Neto, B. et al. (1999). *Modern information retrieval*, volume 463. ACM press New York.
- [3] Balkir, N. H.; Sukan, E.; Ozsoyoglu, G. & Ozsoyoglu, Z. M. (1996). Visual: A graphical icon-based query language. Em *Data Engineering, 1996. Proceedings of the Twelfth International Conference on*, pp. 524--533. IEEE.
- [Blackwell & Green] Blackwell, A. & Green, T. Notational systems—the cognitive dimensions of notations framework. Em Carroll, J. M., editor, *HCI models, theories, and frameworks: Toward a multidisciplinary science*. Morgan Kaufmann.
- [5] Blackwell, A. F. (1996). Metacognitive theories of visual programming: what do we think we are doing? Em *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pp. 240--246. IEEE.
- [6] Boshernitsan, M. & Downes, M. S. (2004). *Visual programming languages: A survey*. Citeseer.
- [7] Choi, R. H. & Wong, R. K. (2015). Vxq: A visual query language for xml data. *Information Systems Frontiers*, 17(4):961--981.
- [8] Chomsky, N. (1956). Three models for the description of language. *IRE Transactions on information theory*, 2(3):113--124.
- [9] de Souza, C. S. (2005). *The semiotic engineering of human-computer interaction*. MIT press.
- [10] de Souza, C. S. & Leitão, C. F. (2009). Semiotic engineering methods for scientific research in hci. *Synthesis Lectures on Human-Centered Informatics*, 2(1):1--122.

- [11] de Souza, C. S.; Leitão, C. F.; Prates, R. O.; Bim, S. A. & da Silva, E. J. (2010). Can inspection methods generate valid new knowledge in hci? the case of semiotic inspection. *International Journal of Human-Computer Studies*, 68(1):22--40.
- [12] de Souza, C. S.; Leitão, C. F.; Prates, R. O. & da Silva, E. J. (2006). The semiotic inspection method. Em *Proceedings of VII Brazilian symposium on Human factors in computing systems*, pp. 148--157. ACM.
- [13] Fagan, J. C. (2006). Usability testing of a large, multidisciplinary library database: Basic search and visual search. *Information technology and libraries*, 25(3):140.
- [14] Gattegno, C. (2010). *Towards a Visual Culture*. EDUC SOLUTIONS Incorporated. ISBN 9780878251971.
- [15] Gomes, G. M. R.; Cendón, B. V. & Prates, R. O. (2015). Estudo da comunicabilidade da interface e do comportamento de busca da informação no portal de periódicos da capes. Spain and Portugal International Society for Knowledge Organization.
- [16] Gomes Filho, J. (2000). Gestalt do objeto. *Sistema de leitura Visual da Forma*. São Paulo: Escrituras Editora.
- [17] Gossen, T.; Nitsche, M. & Nürnberger, A. (2012). Knowledge journey: A web search interface for young users. Em *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval*, p. 1. ACM.
- [18] Green, T. R. (1989). Cognitive dimensions of notations. *People and computers V*, pp. 443--460.
- [19] Haag, F.; Lohmann, S.; Siek, S. & Ertl, T. (2015a). Queryvowl: Visual composition of sparql queries. Em *The Semantic Web: ESWC 2015 Satellite Events*, pp. 62--66. Springer.
- [20] Haag, F.; Lohmann, S.; Siek, S. & Ertl, T. (2015b). Visual querying of linked data with queryvowl. *Joint Proceedings of SumPre*, pp. 2014--15.
- [21] Hils, D. D. (1992). Visual languages and computing survey: Data flow visual programming languages. *Journal of Visual Languages & Computing*, 3(1):69--101.
- [22] Hoeber, O.; Yang, X.-D. & Yao, Y. (2005). Visualization support for interactive query refinement. Em *Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on*, pp. 657--665. IEEE.

- [23] Jones, S. (1998). Graphical query specification and dynamic result previews for a digital library. Em *Proceedings of the 11th annual ACM symposium on User interface software and technology*, pp. 143--151. ACM.
- [24] Kobsa, A. (2004). User experiments with tree visualization systems. Em *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pp. 9--16. IEEE.
- [25] Krusec, R.; Lenic, M.; Mernik, M. & Zumer, V. (2001). Language development in a visual manner. Em *Human-Centric Computing Languages and Environments, 2001. Proceedings IEEE Symposia on*, pp. 396--403. IEEE.
- [26] Larkin, J. H. & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive science*, 11(1):65--100.
- [27] Maletic, J. I.; Marcus, A. & Collard, M. L. (2002). A task oriented view of software visualization. Em *Visualizing Software for Understanding and Analysis, 2002. Proceedings. First International Workshop on*, pp. 32--40. IEEE.
- [28] Manning, C. D.; Raghavan, P.; Schütze, H. et al. (2008). *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.
- [29] Marriott, K. & Meyer, B. (2012). *Visual Language Theory*. Springer New York. ISBN 9781461216766.
- [30] Marriott, K.; Meyer, B. & Wittenburg, K. B. (1998). A survey of visual language specification and recognition. Em *Visual language theory*, pp. 5--85. Springer.
- [31] Myers, B. A. (1990). Taxonomies of visual programming and program visualization. *Journal of Visual Languages & Computing*, 1(1):97--123.
- [32] Navarro-Prieto, R. & Cañas, J. J. (2001). Are visual programming languages better? the role of imagery in program comprehension. *International Journal of Human-Computer Studies*, 54(6):799--829.
- [33] Prates, R. O. & Barbosa, S. D. J. (2007). Introdução à teoria e prática da interação humano computador fundamentada na engenharia semiótica. *Atualizações em informática*, pp. 263--326.
- [34] Sandhu, A. K. & Liu, T. (2014). Wikipedia search engine: Interactive information retrieval interface design. Em *User Science and Engineering (i-USER), 2014 3rd International Conference on*, pp. 18--23. IEEE.

- [35] Spoerri, A. (1995). *InfoCrystal, a visual tool for information retrieval*. Tese de doutorado, Massachusetts Institute of Technology.
- [36] Spoerri, A. (2004). How visual query tools can support users searching the internet. Em *Information Visualisation, 2004. IV 2004. Proceedings. Eighth International Conference on*, pp. 329--334. IEEE.
- [37] Stenning, K. & Oberlander, J. (1995). A cognitive theory of graphical and linguistic reasoning: Logic and implementation. *Cognitive science*, 19(1):97--140.
- [38] Stojanovic, N.; Studer, R. & Stojanovic, L. (2004). An approach for step-by-step query refinement in the ontology-based information retrieval. Em *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 36--43. IEEE Computer Society.
- [39] Zhang, J. & Marchionini, G. (2005). Evaluation and evolution of a browse and search interface: Relation browser++. Em *Proceedings of the 2005 national conference on Digital government research*, pp. 179--188. Digital Government Society of North America.

Apêndice A

Avaliação da Representação Visual

Roteiro para Avaliação da Representação Visual

Título:	Roteiro de perguntas para avaliação da representação
Instituição:	DCC/UFMG
Pesquisadores:	Pricila Resende Rodrigues (pricilarr@dcc.ufmg.br) Raquel Oliveira Prates (rprates@dcc.ufmg.br)

Apresentação do projeto: Apresentar o termo de consentimento ao participante, explicando o objetivo do projeto e do teste. Além disso, explicar brevemente o que é uma busca acadêmica e como é a composição de uma *string*.

Perfil do participante: Coletar informações básicas sobre o perfil do participante, como gênero, idade e escolaridade. Além de identificar o conhecimento do participante quanto a pesquisas acadêmicas e a frequência com que elas são realizadas.

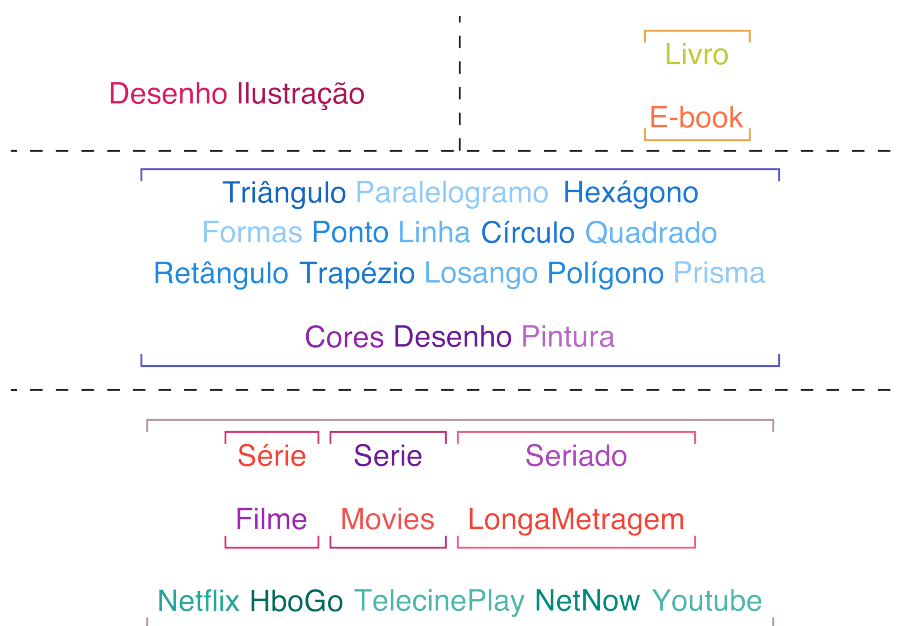
Perguntas para avaliação: Apresentar grupos de *strings* representadas visualmente a partir da gramática definida na linguagem VILAS. As perguntas listadas a seguir servem como um guia para condução da avaliação informal, garantindo que a avaliação contemple todos os pontos identificados como de interesse nesta avaliação.

1. Observando este grupo de imagens, considerando que representam *strings* de busca, o que você entende quanto ao que está sendo representado?

Caderno BlocoDeAnotações Caneta Lápis



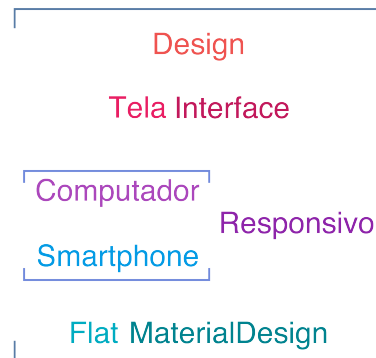
2. Nas imagens, temos duas representações de grupos diferentes, onde uma representa palavras agrupadas por OR, e a outra representa as palavras agrupadas por AND. Para você, o que melhor representa um OR? E um AND? Por que?



3. Nesta imagem tem-se diversas palavras, extraídas de *strings* de busca. Observando essas palavras, qual o sentido você encontra para esta representação?



4. Ao observar as imagens anteriores, você já sabe que a representação da *string* consiste na ligação de palavras com os operadores AND e OR. Você também percebeu que as palavras estão organizadas em diferentes alinhamentos (horizontal e vertical). Além disso, as palavras estão em diferentes cores, variando não somente de cor para cor, mas também em tonalidades da mesma cor. Qual o significado você dá as cores das palavras? Você consegue identificar alguma relação entre as cores e os outros elementos da representação visual?
5. Dado o que já foi visto anteriormente, ao observar esta imagem você já sabe que ela representa uma *string* de busca. Você consegue visualizar a *string* que ela representa?
6. Gostaria de fazer alguma observação? Alguma representação que você ainda está em dúvidas quanto ao significado ou que realmente não entendeu? Alguma sugestão para modificações na representação de algum(s) elemento(s)?



Conclusão: Agradecer ao participante a colaboração na avaliação da representação visual.

Termo de Consentimento

Título:	Avaliação informal da representação de VILAS
Instituição:	DCC/UFMG
Pesquisadores:	Pricila Resende Rodrigues (pricilarr@dcc.ufmg.br) Raquel Oliveira Prates (rprates@dcc.ufmg.br)

Introdução: Este Termo de Consentimento têm o intuito de garantir que você tenha recebido todas as informações sobre a sua participação nesta pesquisa. Para tanto, pedimos que leia todo o texto e assine ao final. Caso tenha alguma dúvida, esteja livre para perguntas.

Objetivo: O objetivo da presente pesquisa é o de avaliar a gramática desenvolvida para a linguagem visual de busca acadêmica, denominada VILAS. Buscando identificar a compreensibilidade da representação visual de uma *string* de busca.

Informações Gerais da Avaliação: Você será solicitado a discutir sobre algumas representações visuais na linguagem VILAS e expressar seu entendimento a partir do que for observado. A discussão será gravada para fins de análise posterior por parte dos pesquisadores envolvidos.

Uso dos Dados: Os dados coletados durante as discussões serão analisados e usados como base para identificação de falhas na comunicação da representação visual e melhorias na linguagem VILAS.

Benefícios: Participando desta pesquisa você estará contribuindo para o desenvolvimento da linguagem VILAS e posteriores pesquisas relacionadas.

Voluntariedade: Você é livre para decidir não participar desta avaliação ou desistir a qualquer momento.

Confidencialidade: Informações que possam identificá-lo serão omitidas.

Dúvidas: Quaisquer dúvidas ou sugestões, estava livre para perguntar durante e posteriormente à realização desta pesquisa, entrando em contato pelo e-mail *pricilarr@dcc.ufmg.br*.

O presente documento descrevendo minha participação na pesquisa Avaliação informal da representação de VILAS foi lido e explicado. Consinto em participar deste estudo e declaro ter recebido uma cópia deste termo de consentimento.

Assinatura do participante

Assinatura do pesquisador

Data: _____

Apêndice B

Avaliação com Uso do Protótipo

Roteiro para Avaliação com Uso do Protótipo

Título:	Roteiro de perguntas para avaliação com uso do protótipo VILAS-web
Instituição:	DCC/UFMG
Pesquisadores:	Pricila Resende Rodrigues (pricilarr@dcc.ufmg.br) Raquel Oliveira Prates (rprates@dcc.ufmg.br)

Apresentação do projeto: Apresentar o termo de consentimento ao participante, explicando o objetivo do projeto e do teste. Além disso, explicar brevemente o que é uma busca acadêmica e como é a composição de uma *string*.

Perfil do participante: Coletar informações básicas sobre o perfil do participante, como gênero, idade e escolaridade. Além de identificar o conhecimento do participante quanto a pesquisas acadêmicas e a frequência com que elas são realizadas.

- Idade: 17-23 24-30 31-37 38+
- Escolaridade: Graduação em curso Graduação
 Mestrado em curso Mestrado
 Doutorado em curso Doutorado
- Qual sua área de pesquisa? _____
- Com qual frequência você procura artigos acadêmicos?
 Quase todos os dias Pelo menos uma vez por semana
 Pelo menos uma vez a cada 15 dias Pelo menos uma vez por mês
 Quase nunca
- Uma string de busca acadêmica são as palavras inseridas na barra de buscas, a fim de obter artigos científicos relevantes para sua pesquisa. Você já definiu uma string de busca acadêmica usando combinações lógicas AND, OR e NOT?
 Sim Não
- Quais ferramentas/repositórios você normalmente utiliza para buscar artigos acadêmicos?

- Você costuma utilizar o modo de busca avançada em sistemas de busca acadêmica? Se sim, em quais situações? Sim Não

Avaliação: Pedir ao participante para utilizar a ferramenta para construir a representação visual de uma *string* de busca. Deixar que o participante explore a ferramenta e auxiliar quando houver dúvidas. Durante o uso da ferramenta, investigar qual a compreensão do participante quanto ao que está sendo representado, solicitando ao participante que fale sobre suas intenções de ação e percepções dos elementos na tela.

Entrevista pós-teste: Investigar a opinião do participante sobre a ferramenta e seu uso.

1. Compare a imagem da *string* que você construiu na ferramenta com a representação textual desta mesma *string*. Você consegue identificar que tanto a representação visual, quanto a representação textual se referem à mesma *string*?
2. O que você acha da representação visual definida?
3. Você ficou em dúvida quanto ao significado de alguma representação?
4. Você tem sugestões em relação à linguagem visual?
5. O que você achou da ferramenta que implementa esta linguagem visual?
6. Você tem sugestões em relação à ferramenta, como novas funcionalidades ou caminhos alternativos para determinada ação?
7. Gostaria de fazer alguma observação? Alguma dúvida ou sugestão?

Conclusão: Agradecer ao participante a colaboração na avaliação da ferramenta web para uso da linguagem VILAS.

Termo de Consentimento

Título:	Avaliação com Uso do Protótipo VILAS-Web
Instituição:	DCC/UFMG
Pesquisadores:	Pricila Resende Rodrigues (pricilarr@dcc.ufmg.br) Raquel Oliveira Prates (rprates@dcc.ufmg.br)

Introdução: Este Termo de Consentimento têm o intuito de garantir que você tenha recebido todas as informações sobre a sua participação nesta pesquisa. Para tanto, pedimos que leia todo o texto e assine ao final. Caso tenha alguma dúvida, esteja livre para perguntas.

Objetivo: O objetivo da presente pesquisa é o de avaliar a ferramenta web desenvolvida para construção de *strings* na linguagem visual de busca acadêmica VILAS.

Informações Gerais da Avaliação: Você será solicitado a utilizar a ferramenta para definir uma *string* de buscas na linguagem visual VILAS. Depois, serão feitas algumas perguntas a respeito da sua experiência ao utilizar a ferramenta, sendo que toda a discussão será gravada para fins de análise posterior por parte dos pesquisadores envolvidos.

Uso dos Dados: Os dados coletados serão analisados e usados como base para identificação de falhas na comunicabilidade da ferramenta e da linguagem.

Benefícios: Participando desta pesquisa você estará contribuindo para o desenvolvimento da linguagem VILAS e posteriores pesquisas relacionadas.

Voluntariedade: Você é livre para decidir não participar desta avaliação ou desistir a qualquer momento.

Confidencialidade: Informações que possam identificá-lo serão omitidas.

Dúvidas: Quaisquer dúvidas ou sugestões, sinta-se livre para perguntar durante e posteriormente à realização desta pesquisa, entrando em contato pelo e-mail *pricilarr@dcc.ufmg.br*.

O presente documento descrevendo minha participação na pesquisa Avaliação do protótipo VILAS-Web foi lido e explicado. Consinto em participar deste estudo e declaro ter recebido uma cópia deste termo de consentimento.

Assinatura do participante

Assinatura do pesquisador

Data: _____