Descoberta de Conhecimento em Bases de Dados aplicada ao processo de extração de informações da *Internet* por meio de Mecanismos de Busca

João Vítor Antunes Ribeiro[1], **Danilo Medeiros Eler**[1]

[1]Faculdade de Ciências e Tecnologia
– Universidade Estadual Paulista "Júlio de Mesquita Filho" (FCT/UNESP)
Departamento de Matemática e Computação
Caixa Postal 468 – CEP 19060-900 – Presidente Prudente – SP – Brasil

{j.antunes.cc,daniloelerunesp}@gmail.com

Abstract. *Search engines allow that pages or websites are found through keywords provided by users, returning lists of related research in order of decreasing relevance. These pages are represented by most of these mechanisms as snippets, which synthesize verbatim. The problem with this search results being represented technique is that in many cases users need to mentally group related snippets, which demands effort and time to discover relationships between the results. This text presents a tool with processing based on Knowledge Discovery in Databases, used in coordinated views of the results grouped by subject and user support to solve this problem of mental grouping.*

Resumo. *Mecanismos de busca permitem que páginas ou sites da Web sejam encontrados por meio de palavras-chave fornecidas por usuários, retornando listas de páginas associadas à pesquisa em ordem decrescente de relevância. Essas páginas são representadas pela maioria desses mecanismos como snippets, que as sintetizam textualmente. O problema dessa técnica de representação dos resultados de busca é que em muitos casos os usuários precisam agrupar mentalmente snippets relacionados, o que demanda esforço e tempo para descobrir relacionamentos entre os resultados. Neste trabalho é apresentada uma ferramenta com processamento baseado na Descoberta de Conhecimento em Bases de Dados, usada em visualizações coordenadas dos resultados agrupados por assunto como apoio ao usuário para solucionar este problema de agrupamento mental.*

1. Introdução

Este trabalho trata essencialmente do estudo e aplicação de técnicas de apoio para exploração de páginas da *Web* por meio de mecanismos de buscas, que atualmente tem ditado como navegamos na *Internet*, tornando a experiência de acesso a seus conteúdos ampla, rápida e simples.

A maioria dos mecanismos de busca na *Web* são baseados nos mesmos princípios básicos de busca, desde *web crawling* (varredura na *Web* para descoberta de *sites* existentes) e indexação, até a filtragem e busca de páginas baseada em interações com o usuário, que recebe um conjunto de *sites* relevantes à sua pesquisa, comumente exibidos como uma lista ordenada por ordem decrescente de relevância. Como tratado em [Chakrabarti 2003],

inicialmente os resultados eram exibidos para os usuários como listas de *links* aparentemente sem sentido. Esta técnica de representação dos *sites* sofreu diversas modificações, até passar a ser representada por *snippets*.

Snippet (do português, fragmento) tornou-se a técnica mais popular de representação de resultados de busca de conteúdo da *Internet*, isso porque consegue representar a essência de páginas ou *sites* inteiros em um espaço reduzido, contendo apenas os dados mais relevantes que uma página tem a oferecer de acordo com o interesse informado pelo usuário. Em resposta às pesquisas feitas pelos usuários, os mecanismos de busca retornam listas de *snippets* relacionados à pesquisa. Cada *snippet* possui basicamente um título, um *link* e um breve texto que resume o conteúdo do *site* sintetizado. O problema deste método de representação é a necessidade de, para volumes muito grandes de *snippets* retornados, não haver como agrupar automaticamente os resultados mais relevantes à pesquisa do usuário. Uma busca pelo termo "manga", por exemplo, poderia retornar uma lista com conteúdos relacionados às revistas mangá, às mangas de camisa, ou à fruta de mesmo nome, resultados estes dispersos em localizações diferentes das páginas de resultados de acordo com seu grau de relevância. Isto obriga os usuários a armazenar mentalmente a localização dos *snippets* que acham mais relevantes ou que estão mais relacionados semanticamente. Neste trabalho, é apresentada uma solução para este problema por intermédio da aplicação da Descoberta de Conhecimento em Bases de Dados (do inglês, *Knowledge Discovery in Databases* - KDD), agrupando resultados relacionados semanticamente utilizando Mineração de Dados e exibindo-os aos usuários por meio de representações visuais existentes na área de Visualização de Dados. A perspectiva é que isto proporcione uma experiência mais completa ao usuário ao procurar conteúdos na *Web*, possibilitando filtrar resultados similares mais rapidamente e com menor esforço mental, utilizando representações visuais para apoiar as representações textuais dos *snippets*.

2. Contextualização

Segundo Levene (2010), as primeiras tentativas de se organizar o conteúdo da *Web* vieram por parte dos *sites* Yahoo e Excite, que introduziram o conceito de *search engine* ou mecanismo de busca. Em um cenário onde a *Web* ainda não contava com páginas dinâmicas e mecanismos de interatividade com o usuário, os conteúdos disponíveis eram em grande parte texto simples com formatação rudimentar, onde a minoria apresentava algum recurso multimídia, como imagens e sons. A proposta do Yahoo foi agrupar o maior número de páginas aparentemente relacionadas por meio de classes de assuntos predefinidas, como esportes, notícias e entretenimento, sendo essa classificação de feita manualmente, vasculhando *site* por *site*. Com isso, tornou-se mais prático para os usuários encontrar conteúdos relacionados, bastando acessar esses agrupamentos e navegar por seus *sites*. Esse conceito de acesso a conteúdos da *Internet* revolucionou a maneira pela qual a *Internet* era utilizada, e contribuiu significativamente para sua popularização [Levene 2010].

O Excite, na época o único concorrente à altura do Yahoo, buscava conteúdos na *Web* com base em *keywords* (do portugês, palavras-chave) que os usuários forneciam ao *site* [Levene 2010], um método mais intuitivo e simples do que o sistema proposto pelo Yahoo. Atualmente, este é o modelo mais utilizado pelos mecanismos de busca, inclusive o Google, o Bing e o Yahoo.

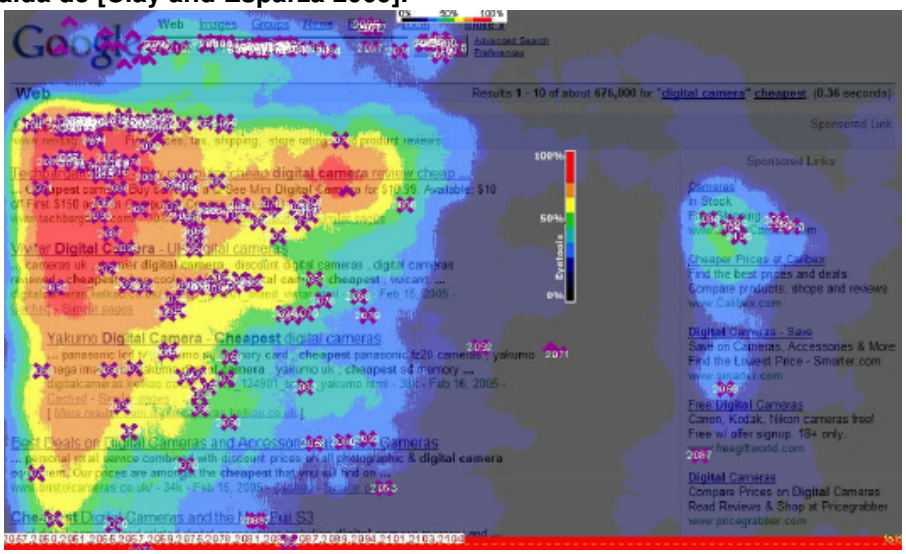
Mesmo oferecendo os melhores mecanismos de busca até o momento, os serviços do Yahoo e Excite eram ineficientes e rudimentares, pois não retornavam resultados muito relevantes às buscas específicas dos usuários [Levene 2010]. Coube ao Google a revolução da forma como hoje acessamos e interagimos com a *Internet*, que se deve em grande parte ao algoritmo utilizado para otimizar buscas de conteúdo nas páginas da *Web*. O *Page Rank*, como é chamado, é um algoritmo com uma ideia simples, que verifica se uma página é mais indicada do que outra em relação à uma pesquisa, por meio de um sistema de "votação" de uma página para outra. Por exemplo, se para duas páginas, que aparentemente tratam do mesmo assunto, for constatado que a página A possui mais referências à seu *link* do que a página B, a página A deve aparecer antes de B na listagem de *sites* retornados, pois A recebeu mais "votos" em relação à B, e por isso é considerada mais relevante. Esta lógica simples, porém muito eficiente, possibilitou aos usuários realizar buscas mais precisas e específicas na *Internet*.

Atualmente existem muitos tipos de *snippets*, ou *rich snippets*, cada qual específico para dado tipo de informação. No trabalho desenvolvido por Nieto (2012), um exemplo prático interessante é adotado para mostrar, entre outros aspectos, a diversidade de informações que se pode conseguir na *Web* em razão de um único termo [Nieto 2012]. No caso, o autor utilizou o termo "jaguar"; reproduzindo esse mesmo exemplo, pode-se notar na Figura 1 que o usuário pode estar procurando por produtos da marca de carros Jaguar, o animal jaguar, fotos de carros ou animais, músicas, uma pessoa de sobrenome Jaguar, um jogo, um clube entre muitas outras possibilidades. Desta forma, é possível entender como uma busca na *Web* pode ser difícil e ambígua. Na figura, o tipo de *snippet* mais comum é denominado *Snippet Comum*. A região intitulada *Snippet Destaque* exibe os resultados buscados com maior frequência sobre o assunto. Para os conjuntos de *snippets Snippet Notícia* e *Snippet Imagem*, são apresentadas notícias e imagens sobre o assunto, respectivamente. É importante salientar que a estrutura das páginas de resultados são variáveis de acordo com o assunto, mas a mais comum é composta apenas por *snippets* similares ao destaque da figura, *Snippet Comum*. Há ainda *rich snippets* específicos para representar localidades, músicas, vídeos, autores e produtos.

Figura 1. Página de resultados para a palavra "jaguar".

Snippets são ótimos para representar a síntese do conteúdo de *sites* inteiros em espaços reduzidos, mas o modo como eles são apresentados para os usuários pode ser, muitas vezes, desvantajosa. Considerando o exemplo da Figura 1, a região inferior da figura (Paginação) informa que o buscador retornou 10 páginas de resultados encontrados para a busca sobre "jaguar". Sabendo disso, imagine um usuário que esteja procurando *sites* com conteúdo sobre o time estadunidense de futebol americano chamado "Jaguars". Na primeira página de busca, menos de 6% dos resultados aborda o assunto, obrigando o usuário a especificar mais sobre o mesmo ou navegar pelas diversas páginas de resultados retornados, agrupando mentalmente aqueles que apresentam conteúdo semântico, isto é, que tratam essencialmente do mesmo assunto. À primeira vista, este exemplo pode parecer insuficiente para concluir que o formato atual dos mecanismos de busca apresenta desvantagens consideráveis, pois no exemplo não foi especificado que se estava procurando pelo time de futebol americano, tendo sido fornecido ao buscador apenas um termo ambíguo. Isto é verdade, pois mecanismos de busca não são obrigados a pressupor o que os usuários estão procurando, mas por outro lado podem facilitar a pesquisa por meio da exibição de possíveis grupos de *sites* que apresentam conteúdo similar em relação ao termo procurado. Ao invés de mostrar os resultados com base apenas no nível de relevância da página, que implica que páginas com conteúdos heterogêneos possam aparecer sequencialmente, os mecanismos também podem agrupar as páginas que possuem conteúdo semântico e retornar esses grupos de assuntos para o usuário, não necessitando de agrupamento mental das páginas relacionadas.

Figura 2. *Golden Triangle* dos resultados de uma página de busca (SERP) – Extraída de [Clay and Esparza 2009].

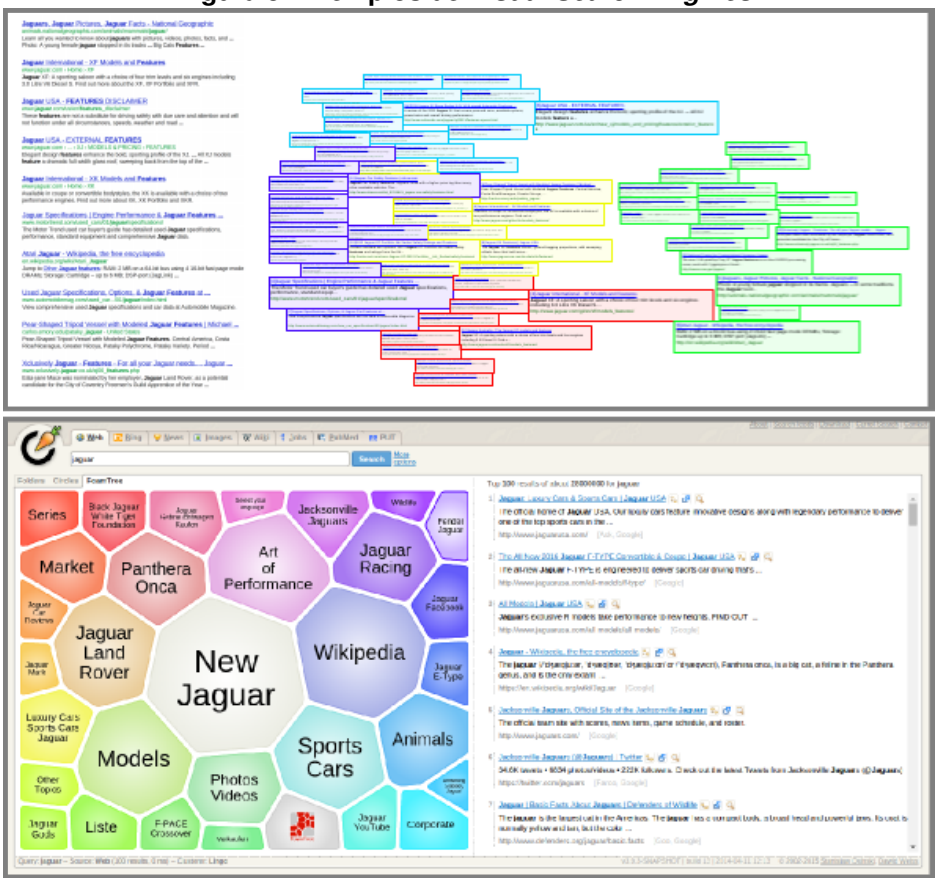


Usuários não costumam navegar por todas as páginas de resultados retornadas pelos mecanismos de busca, muito pelo contrário. Como discutido em [Clay and Esparza 2009], usuários de buscadores não costumam ler nem a primeira página inteira de resultados de uma busca, percorrendo visualmente apenas os primeiros 3 ou 4 resultados da primeira página, de forma triangular, como mostrado na Figura 2; o mapa de calor mostra as áreas mais visualizadas e clicadas (representadas pelo símbolo "X") da página, onde a cor vermelho indica maior intensidade de exploração visual dos usuários e a cor preto indica menor, sendo utilizado um gradiente de cores para represen-

tar intensidades intermediárias. Os especialistas deram à essa região o nome de *Golden Triangle* (do português, Triângulo Dourado), pois é a região com maior destaque de uma página de resultados, página esta também conhecida como *Search Engine Results Page* (SERP). Este fato é utilizado por muitos buscadores como artifício comercial, uma vez que podem comercializar a colocação de um produto ou empresa no *ranking* de resultados.

Com o objetivo de solucionar o problema de agrupamento mental por parte dos usuários, surgiram outras formas de representar os conteúdos procurados na *Web*, como os *Visual Search Engines*. *Visual Search Engine* (do português, Mecanismo de Busca Visual), é um tipo de buscador que representa os *sites* por meio de componentes pictóricos, como imagens do aspecto das páginas (*snapshot*), figuras, símbolos, ou até mesmo *snippets*. Geralmente, esses componentes são agrupados de acordo com suas características semânticas, levando em consideração tanto a relevância do resultado quanto sua semântica, sendo possível encontrar e selecionar otimizadamente os resultados de conteúdo similar. Exemplos desses sistemas são o TouchGraph Google Browser, Liveplasma, RedZ, Quintura, Carrot 2 e oSkope. Todos estes sistemas são baseados na *Web*, o que implica que os usuários podem utilizá-los por um navegador, sem a necessidade de instalar programas externos. Também existem ferramentas que foram desenvolvidas para serem executadas a partir de *Desktops*, localmente nas máquinas dos usuários, como a ProjSnippet [Nieto 2012], desenvolvida por Nieto (2012), e a PEx, descrita em [Paulovich et al. 2007].

Figura 3. Exemplos de *Visual Search Engines*.



Em seu trabalho, Nieto (2012) constata que a representação de resultados de busca na *Web* por meio de técnicas de Visualização contribui para um aumento das informações extraídas dos dados dos resultados, dando uma visão geral dos relacionamentos entre eles. Isso implica em buscas mais rápidas e precisas, pois o usuário não precisava agrupar mentalmente resultados similares. A ferramenta desenvolvida nesse estudo (ProjSnippet) está ilustrada na Figura 3 (em cima), que apresenta um exemplo de utilização com base no buscador do Google, mostrando uma visualização de 64 *snippets*, mais de 6 páginas de resultados.

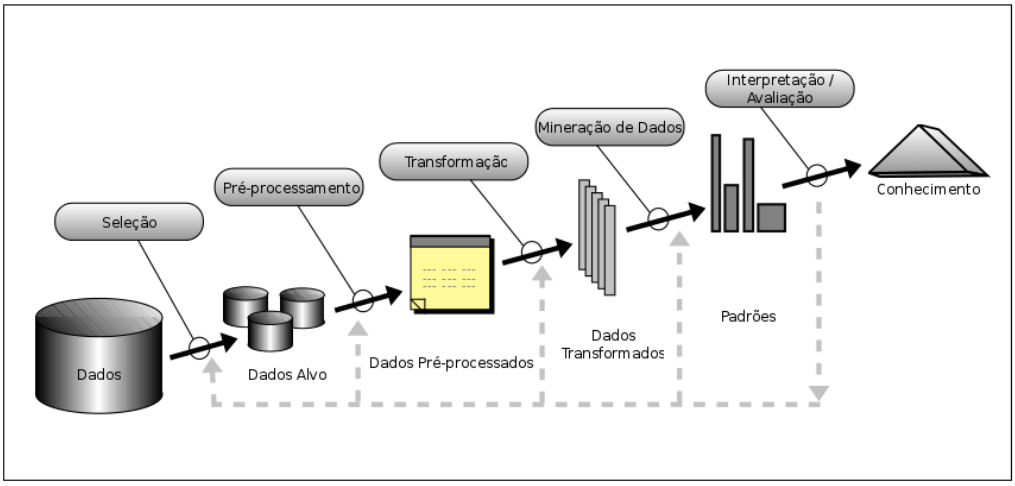
Carrot 2 e TouchGraph Google Browser também são exemplos de como a Visualização pode apoiar na exploração de resultados em mecanismos de busca. A Figura 3 (embaixo) ilustra o funcionamento da primeira ferramenta, na busca pelo termo jaguar. Com isso, é possível entender como diferentes formas de representação de um mesmo conjunto de dados podem influenciar consideravelmente na qualidade de conhecimento adquirido. No exemplo, podem ser reconhecidos mais rapidamente os agrupamentos de *sites* relacionados de acordo com assuntos comuns, como *Sports Cars* e *Animals*. Esta visualização similar a uma colméia arranja esses grupos hierarquicamente, onde as formas maiores representam os grupos com maior relevância e as menores os com menor. A localização desses grupos, bem como suas cores também são hierárquicas; os grupos mais próximos ao centro e as cores mais claras são mais relevantes do que os periféricos ou com cores mais intensas.

3. Fundamentação teórica

Segundo descrito em [Fayyad et al. 1996], a Mineração de Dados é uma das etapas de um processo complexo conhecido como Descoberta de Conhecimento em Bases de Dados, que se designa a provêr significado em grandes volumes de dados, permitindo que um analista extraia mais informações úteis se comparado a uma análise manual de exploração e extração de significados dos dados. É válido frisar que a aplicação de KDD só faz sentido em grandes volumes de dados, em que a atividade manual de extração de informações é considerada trabalhosa ou pouco otimizada, em razão do tempo e do trabalho despendido para minerar os dados visualmente; em KDD, quanto maior o volume de dados processados, mais precisas costumam ser as informações que podem ser extraídas. Na Figura 4, o processo de KDD é ilustrado como uma composição iterativa e cíclica de tarefas ou subprocessos, necessários para extrair conhecimento de bases de dados.

Fayyad (1996) afirma que no processo de KDD o primeiro passo é selecionar, dentre um conjunto grande de dados, os dados alvo da análise, da qual se deseja extrair informações úteis ainda não conhecidas. Isso faz com que o foco do processo seja direcionado para apenas uma parcela reduzida dos dados originais, sendo esta fase denominada **Seleção**. Na fase seguinte, o **Pré-processamento** é realizado com o objetivo de remover possíveis "impurezas" do conjunto de dados alvo, como informações erradas e incompletas ou, no caso de dados do tipo texto, palavras consideradas dispensáveis (como os pronomes "eu", "tu" e "nós" e os artigos "um", "o" e "as", que geralmente não agregam informação útil para o conjunto de dados analisados) para o processo de extração de conhecimento. Após esta etapa, os dados selecionados e já pré-processados são enviados à etapa de **Transformação**, que padroniza o conjunto de dados segundo uma estrutura digital para criar um ambiente de análise homogêneo. A fase de **Mineração de Dados** se responsabiliza por aplicar algoritmos de descoberta ou reconhecimento de padrões

Figura 4. Etapas do processo de KDD – Adaptada de [Fayyad et al. 1996].



nos dados, como valores parecidos, crescimentos ou anomalias. O processo de KDD não foi desenvolvido para ser uma atividade independente e autônoma, e por isso necessita de um analista ambientado ao conjunto de dados para realizar a **Avaliação** dos resultados do processo e, com isso, ser capaz de extrair conhecimento acerca dos dados [Fayyad et al. 1996].

A Visualização de Dados pode ser caracterizada como uma área da computação responsável por facilitar o entendimento de dados, através de sua representação como formas pictóricas, como gráficos ou imagens. Isto permite que a Visualização possa ser utiliza durante o processo de KDD para auxiliar na interação entre as etapas do seu processo. McCormick (1987) a define como um método de computação utilizado para transformar dados simbólicos em geométricos ou gráficos, possibilitando a extração de informações desconhecidas a respeito de um conjunto de dados [McCormick et al. 1987]. De acordo com Schroeder (2003), a Visualização de Dados pode ser dividida entre *Scientific Visualization* (do português, Visualização Científica) e *Information Visualization* (do português, Visualização de Informação) ou *InfoVis* [Schroeder et al. 2003], tendo sido a Visualização Científica a primeira a ser desenvolvida, como relatado em [McCormick et al. 1987]. A Visualização Científica se preocupa em representar dados de um jeito mais realista e condizente com sua natureza, abrangendo áreas como a Estatística, a Cartografia e a Medicina [McCormick et al. 1987]. Já a Visualização de Informação trata essencialmente de representações de caráter abstrato e sem significado ou forma definida/realista, como dados pluviométricos, mensagens de texto de fóruns de discussão e dados da *Web*. Em [North and Shneiderman 2000], uma interessante observação é feita, salientando como a visualização é utilizada em muitas situações do cotidiano, como nos sistemas operacionais, que utilizam estrutura de diretórios em árvore e a representação dos arquivos por meio de ícones, muito mais intuitivo do que se representados apenas pelo nome. O sistema de semáforo utilizado no trânsito também pode ser caracterizado como um modo de visualização, que simboliza com as cores verde, amarelo e vermelho os dados para seguir em frente, atenção e parada, respectivamente.

Há muitos fatores que influenciam na qualidade da informação que pode ser extraída de um conjunto de dados. O atributo temporal é um importante fator a ser

considerado, pois permite saber informações como ordem de ocorrência, períodos e pontos temporais, que por sua vez permitem saber quando determinada ocorrência aconteceu, se uma ocorrência aconteceu antes de outra e em quais períodos as ocorrências estão concentradas. Trabalhos neste sentido podem ser encontrados em [Chittaro et al. 2003] e [Shimabukuro 2004]. Quanto à representatividade dos dados pelas visualizações, a questão da dimensão aplicada é um dos fatores mais influenciantes [Inselberg and Dimsdale 1990]. Para determinados tipos de informações, representar dados bidimensionalmente pode ser mais recomendado, enquanto que para outros a representação tridimensional é melhor. Sabendo dessa relatividade, um novo processo surgiu com o propósito de juntar as melhores características de cada tipo de visualização, que utilizando coordenações auxiliam o usuário a extrair uma quantidade maior de informação se comparada à aplicação de apenas uma técnica de representação.

A Coordenação se destina a estudar sistemas de representação de dados que combinam diferentes técnicas de Visualização, que podem ter objetivos e dimensões variáveis. Esta coordenação visa oferecer ao usuário múltiplas visões de um mesmo conjunto de dados, utilizadas para melhor compreender seus aspectos, conforme descrito em [North and Shneiderman 2000], que apresenta um estudo comparativo das vantagens e desvantagens da coordenação em relação à técnicas de Visualização semi-coordenadas ou não coordenadas. Um modelo genérico de coordenação é apresentado em [Boukhelifa et al. 2003] e [Boukhelifa and Rodgers 2003]. Neste último trabalho, os autores caracterizam a coordenação como um poderoso artifício para interação com múltiplas visões, utilizado para auxiliar na descoberta de novos relacionamentos entre conjuntos de dados; permite diferentes pontos de vista sincronizados que ajudam a refinar parâmetros de busca e reformular metas. Quanto à interatividade com o usuário, em [Boukhelifa and Rodgers 2003] é apresentado um protótipo de ferramenta (CViews) utilizado para realizar múltiplas visões coordenadas e sincronizadas, fazendo com que diferentes pontos de vista sejam sincronizados e que mudanças de um componente disparem mudanças em outros componentes, por meio de transformações geométricas, como translação ou rotação, por exemplo.

4. Abordagem proposta

Baseado na definição de Fayyad (1996) para Descoberta de Conhecimento em Bases de Dados, a ferramenta Pinda foi desenvolvida seguindo todas as etapas deste processo. A palavra *pinda* vem do tupi-guaraní, e em português significa "anzol". Este nome foi escolhido para denominar a ferramenta pois, em determinado nível de abstração, informações são "pescadas" na *Internet* quando se utilizam mecanismos de busca. Neste sentido, uma das ferramentas para pescar essas informações pode ser o anzol, que por meio de iscas (palavras-chave) é utilizado para "pescar" informações (resultados de busca) na *Internet*.

Pinda foi desenvolvida para o ambiente *Web*, implementada em módulos que caracterizam as etapas do processo da KDD, que vai desde a fase de *Seleção* até a fase de *Interpretação/Avaliação*.

4.1. Seleção

Na etapa de Seleção, foi dada preferência ao uso de APIs (do inglês *Application Programming Interface*s, são interfaces utilizadas para acessar serviços externos de uma aplicação,

sem contudo fazer necessariamente parte do projeto dessa aplicação, com foco direcionado apenas na utilização de seus serviços, similar ao uso de bibliotecas) de busca gratuitas e ilimitadas para o fornecimento dos dados selecionados segundo as buscas dos usuários, o que implicou na escolha de duas APIs baseadas no idioma inglês, a Faroo - *Free* API e a DuckDuckGo - *Instant Answer* API. Embora ambas sejam gratuitas, com limite de 100 resultados retornados por busca, escritas em Javascript e sem limite de quantidade de requisições, elas possuem diferenças em questão de tempo e diversidade de resultados encontrados. Comparada à API do DuckDuckGo, a Faroo retorna um volume maior de resultados e com maior diversidade, mas em questão de tempo é muito inferior à DuckDuckGo, pois limita a recuperação de resultados à uma página por segundo, o que não acontece na DuckDuckGo. Esta particularidade motivou a inserção das duas APIs na ferramenta, para que o usuário possa alternar entre elas.

4.2. Pré-processamento

O papel destas APIs de busca resume-se apenas em retornar o conteúdo dos *snippets* relacionados às buscas dos usuários, cada um composto por um título, um endereço de uma página e um resumo desta página. Para automatizar o processo de agrupamento dos resultados de acordo com seus assuntos, optou-se por utilizar como dados de mineração tanto os textos dos títulos quanto dos resumos dos *snippets*, pois um maior volume de dados válidos tende a implicar em maior qualidade de conhecimento extraído. Para potencializar a quantidade de informações similares a serem comparadas, foi necessário realizar um Pré-processamento da combinação destes dados, títulos e textos. Este processo consiste em reduzir o conjunto de dados a ser processado, por meio da eliminação de impurezas textuais como acentos, palavras irrelevantes (como artigos ou preposições gramaticais) ao contexto, heterogeneidade de caixa (letras maísculas e minúsculas) e stemização ou radicalização (redução de palavras para seu radical). Em testes realizados com o Algoritmo de Porter para radicalização de palavras, foi constatado que a redução de termos é, em média, de apenas 10% para os dados dos *snippets*. Considerando ainda o fato do truncamento natural dos termos processados, ou seja, a consequente modificação de termos como “apple” para “apl”, caso não seja feito um tratamento adequado, concluisse que o custo deste processamento é inviável no contexto, implicando na não utilização apenas deste método de pré-processamento na ferramenta.

4.3. Transformação dos dados

Com todos os dados padronizados, o resultado é a criação de um vetor de textos, onde cada posição representa um *snippet*, que contém um único texto reduzido em uma listagem de palavras com letras minúsculas, sem acento, números ou termos considerados irrelevantes, ordenadas de acordo com sua ocorrência no texto original. Isto por si só já define a fase de Transformação dos Dados, mas ainda é preciso elaborar um modo de identificar padrões nos dados, ou seja, não basta simplesmente transformar os dados resultantes do Pré-processamento, é preciso transformar esses dados pensando nas dependências da Mineração de Dados. Isto implicou na utilização de uma ferramenta de Mineração de Textos para identificar os agrupamentos dos resultados, a Projection Explorer (PEx), originada do projeto descrito em [Paulovich et al. 2007]. Trata-se de uma ferramenta de Mineração e Visualização de Dados do tipo texto, de código aberto e gratuita para fins acadêmicos.

Embora tenha sido escrito em Java para *Desktops*, a PEx possui muitos recursos de Mineração de Dados, e o fato de ser de código aberto influenciou decisivamente para que pudesse ser utilizada neste projeto. O foco foi adaptar a ferramenta para que ela pudesse receber requisições da Pinda e dispôr os textos transformados em grupos ou *clusters* hierárquicos, identificando e organizando esses dados em assuntos e sub-assuntos. Deste modo, foi necessário criar uma matriz de documentos (*snippets*) por termos no formato suportado pela PEx, matriz essa composta por um cabeçalho contendo todos os termos existentes em todos os textos, sem repetição, e um eixo vertical ou coluna com o nome de cada *snippet*, seguido da frequência absoluta de cada termo em relação aquele *snippet*, alinhado à coluna referente ao termo. Com essas informações, a PEx é capaz de criar uma representação dos *snippets* utilizando algoritmos de Redução de Dimensionalidade (Projeção Multidimensional) dos termos, que consiste em transformar a quantidade de características (termos) dos textos em um número suficientemente pequeno, sem perda de informações, para que possa ser mais facilmente comparado com outros elementos. No caso, foi utilizado o algoritmo *Interactive Document Map* (IDMAP), por se tratar de uma otimização de outras técnicas de Projeção Multidimensional [Paulovich et al. 2007]. Este algoritmo faz com que a PEx retorne uma lista de pontos bidimensionais representando os *snippets*. Essas coordenadas dispõem os *snippets* em um espaço abstrato de duas dimensões, permitindo que sejam aplicados algoritmos de reconhecimento de grupos de dados melhor relacionados, considerando entre outros fatores suas distâncias euclidianas.

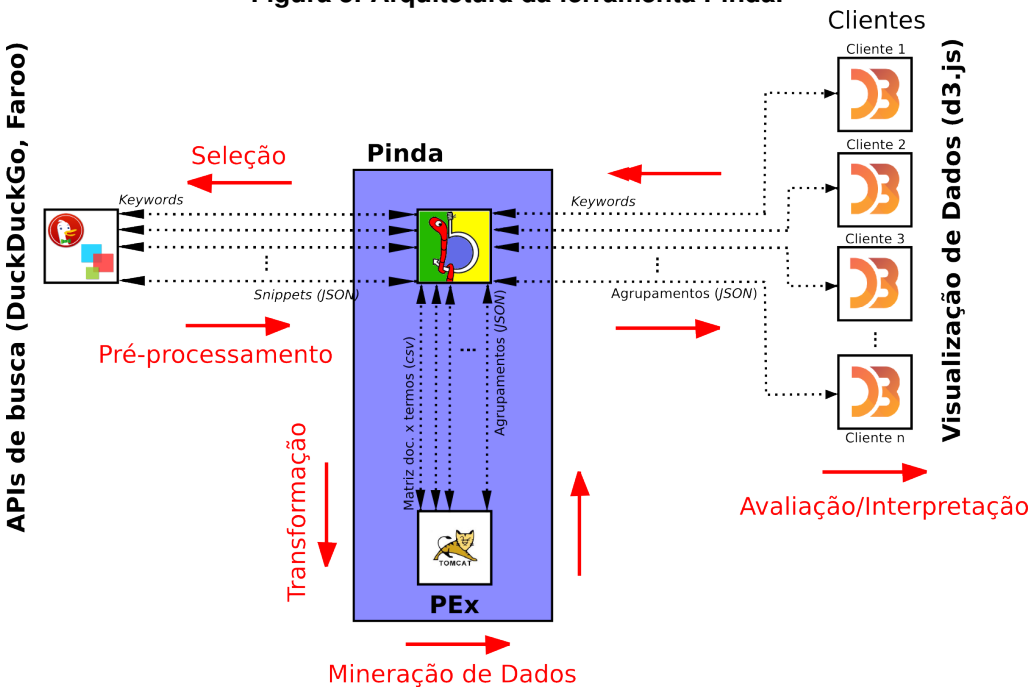
4.4. Mineração de Dados

O algoritmo de clusterização escolhido para agrupar os assuntos foi o K-means, por ser eficaz, simples, e um dos mais utilizados pela comunidade científica. Ele objetiva particionar n elementos dentro de k grupos, onde cada elemento pertence ao grupo mais próximo da coordenada da média aritmética da distância entre os elementos de um grupo atual, sendo k um número fixo da quantidade de grupos que se deseja obter, informado *a priori*, menor ou igual ao número total de elementos n. No caso do agrupamento hierárquico, a PEx foi adaptada para realizar sucessivos agrupamentos em todos os grupos formados, até que a quantidade de elementos de todos os grupos de assuntos seja menor do que k. Na ferramenta Pinda, o valor de k foi definido como 5, baseado no estudo descrito em [Clay and Esparza 2009] e abordado na Seção 2, que descreve o conceito do Triângulo Dourado.

Todas as adaptações realizadas na PEx foram implementadas no código-fonte original da versão 1.6.3, resultando na criação de um servidor Tomcat (versão 7.0.62) para intermediar as interações entre os resultados buscados com a Pinda, utilizando as linguagens PHP e Javascript, e a clusterização hierárquica desses resultados por parte da PEx modificada, utilizando Java. Na Figura 5, a arquitetura da ferramenta Pinda é ilustrada em módulos que sintetizam o processo completo de busca de conteúdo na *Internet* e agrupamento de resultados por assunto. Como se pode observar, a maior parte do processamento é realizada no servidor onde está hospedada a ferramenta, que intermedia requisições para mecanismos de busca via APIs afim de agrupar os resultados por assuntos, retornados para os clientes ou usuários através de arquivos do tipo JSON para finalmente serem representados por técnicas de Visualização utilizando a biblioteca *d3.js*, a partir dos computadores dos usuários. A região da figura destacada em azul representa o relacionamento entre a Pinda e as APIs utilizadas, assim como a PEx. Neste último caso, a comunicação entre as

ferramentas se dá através de uma matriz de documentos por termos, descrita na subseção anterior. O papel do servidor Tomcat é intermediar essas requisições com os métodos *GET* e *POST*, para que seja possível retornar o arquivo *JSON* de grupos hierárquicos gerados pela PEx, escolhido por ser simples de entender, fácil de processar e "leve" o bastante para ser transmitido pela *Internet*.

Figura 5. Arquitetura da ferramenta Pinda.



O processo de clusterização hierárquica realizado pela PEx modificada sintetiza a fase de Mineração de Dados do processo de KDD; é nessa fase que padrões e regras são descobertos. Considerando o caso dos *snippets*, é nessa fase que se sabe quais resultados tratam de assuntos próximos e, portanto, são similares, sem contudo ter o trabalho de analisar manualmente os textos.

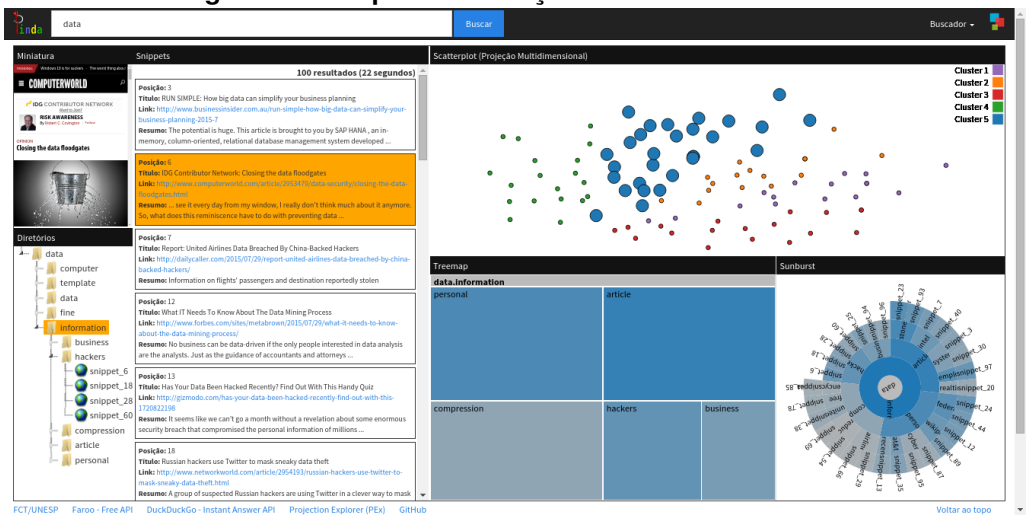
4.5. Interpretação/Avaliação

No processo de Descoberta de Conhecimento em Bases de Dados, a Interpretação/Avaliação dos dados é a última fase deste processo iterativo. Isso significa que, após a Mineração de Dados, já é possível obter conhecimento a respeito dos dados processados, sendo possível julgar hipóteses ou tomar decisões com base no conhecimento obtido. Contudo, entre as fases de Mineração de Dados e Interpretação/Avaliação, assim como em qualquer outra fase do processo, a Visualização de Dados pode ser aplicada com o objetivo de otimizar a qualidade de percepção do analista ou avaliador a respeito do fenômeno analisado.

Pinda foi desenvolvida com foco em Visualização de Dados na fase de Interpretação/Avaliação, almejando ser intuitiva para o usuário comum, contando com visualizações coordenadas dos dados, e eficaz para o usuário avançado, contando com múltiplas técnicas de Visualização de Dados, como pode ser conferido na Figura 6 (também inserida no Apêndice A para melhor visualização). Nesta figura, é ilustrado

o ambiente de busca da ferramenta Pinda. A barra superior é utilizada para realização de buscas e seleção da API a ser utilizada. Abaixo dessa barra são exibidas formas distintas de representação dos dados minerados. Com exceção da visualização de Miniatura, todas as outras 5 possuem eventos de coordenação para as demais visualizações, inclusive para a Miniatura.

Figura 6. Exemplo de utilização da ferramenta Pinda.



A *Treemap* exibe os resultados reunidos em grupos hierárquicos, sendo possível navegar apenas por um grupo por vez clicando nos retângulos representativos, que mostram a quantidade de *snippets* agrupados ao passar o mouse sobre o retângulo. Para voltar para um grupo acima, basta clicar sobre a barra entre o título desta visualização e os retângulos agrupados, que também informa a ordem hierárquica dos agrupamentos selecionados, separados por pontos. Assim como a Diretórios, a *Sunburst* é similar à *Treemap*, mas possui a vantagem de permitir total visualização dos sub-grupos de um *cluster* selecionado, sendo necessário clicar no círculo interno central para voltar para um nível superior de agrupamento. A *Scatterplot* exibe o plano abstrato de dados originados do processo de Redução de Dimensionalidade realizado pela PEx. Nela é possível observar os grupos mais generalizados dos agrupamentos, que representam os *clusters* pais de todos os grupos. *Snippets* é a representação clássica de resultados de busca, embora esteja coordenada com as demais visualizações. Ao selecionar um *snippet* tanto por meio da visualização *Snippets* quanto pela *Scatterplot* com um clique único, todas as demais visualizações são coordenadas para representar os mesmos elementos de maneiras diferentes e relativas à cada visualização, sendo que o evento de clique duplo abre uma nova guia no navegador direcionada para a página do *snippet* selecionado.

5. Aplicação

O objetivo principal do desenvolvimento da Pinda, a solução do problema de agrupamento mental de *snippets* por parte dos usuários, não implica que a ferramenta só possa ser utilizada nesse sentido. O fato de possuir múltiplas visualizações coordenadas faz com que a mesma possa ser utilizada em diferentes situações, com diferentes objetivos.

A visualização Miniatura é útil para que o usuário verifique instantaneamente o conteúdo real de uma página, auxiliando-o a formular hipóteses de relacionamentos entre

páginas ou conteúdos, podendo navegar pelos *sites* acessados sem sair da ferramenta de busca.

Utilizando a visualização de Diretórios, é possível compreender mais diretamente a organização hierárquica dos grupos de assuntos identificados, em razão de ser uma técnica amplamente utilizada em sistemas operacionais para representar hierarquias de arquivos e diretórios ou pastas. Trata-se de uma técnica simples, fácil de ser compreendida e consagrada, permitindo obter uma visão geral de todos os relacionamentos existentes.

Embora não apresente hierarquicamente os agrupamentos dos conjuntos de resultados, a visualização *Scatterplot* é fundamental para informar mais precisamente quais as "verdadeiras distâncias" semânticas entre os resultados, possibilitando conhecer de imediato os principais grupos de assuntos para volumes de dados muito grandes.

Treemap e *Sunburst* são visualizações análogas e com características diferentes. Ambas são destinadas a informar todas as hierarquias organizadas entre os grupos de dados, mas além da geometria utilizada, a principal diferença entre estas técnicas utilizando a ferramenta é o fato da *Treemap* limitar a visualização da hierarquia dos grupos apenas para o nível atual que está sendo visualizado, isto é, por questões de interatividade, impossibilita a informação dos demais grupos de um nível de agrupamento, exibindo apenas o atual. Em compensação, embora a *Sunburst* não apresente esta limitação, é mais sobrecarregada de informações, exibindo de uma vez toda a hierarquia de assuntos identificada, o que não chega a ser prejudicial quando representados um número reduzidos de dados.

Considerando um exemplo de busca utilizando o termo "brazil problem", com o objetivo de conhecer os temas relacionados com os problemas do Brasil, através da utilização da API do Faroo é possível constatar que os principais grupos de assuntos identificados são estádio, vida, mundo e América. Também são relacionados os temas corrupção, construção e Olimpíadas ao tema estádio, assim como os temas estádios, Rússia, Qatar e FIFA, que estão associados ao tema mundo. Esses relacionamentos podem ser interpretados como reflexos das suspeitas de corrupção envolvendo a construção de estádios para a Copa do Mundo de 2014 (sediada no Brasil), e o fato de que a próxima Copa será sediada pelo Qatar em 2022, assuntos relativamente recentes no panorama nacional. A ferramenta também apontou como um dos principais temas relacionados à saúde do Brasil o uso de *crack* (entorpecente derivado da cocaína) por todo o território brasileiro. Com isso, este exemplo mostra como a ferramenta também pode ser utilizada para conhecer e explorar relacionamentos desconhecidos entre assuntos abrangentes, sem contudo necessitar especificação prévia de palavras-chave sobre determinado tema, o que torna a busca mais ampla e favorece a identificação de um conjunto maior de temas relacionados.

6. Avaliação

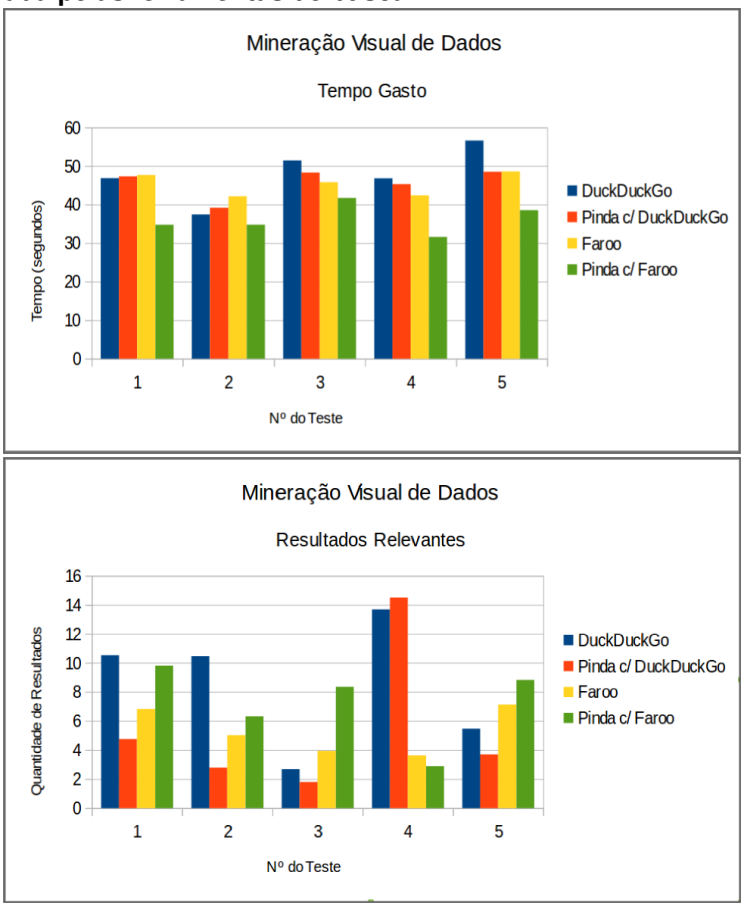
Para avaliar a ferramenta Pinda, foram selecionados 30 alunos do curso de Bacharelado em Ciência da Computação para testarem a ferramenta por meio de seu uso. Eles tiveram que responder a um questionário cuja intenção era comparar os sistemas dos originadores das APIs DuckDuckGo e Faroo, com a utilização de suas APIs por meio da ferramenta Pinda. Como o objetivo era avaliar tanto a interface quanto o processamento da ferramenta desenvolvida, assim como o impacto de sua utilização, o teste foi dividido em duas partes, a inicial contendo perguntas que deveriam ser respondidas com a utilização das ferramentas, e a segunda com perguntas de avaliação pessoal dessas 3 ferramentas (DuckDuckGo,

Faroo e Pinda), com quatro elementos de avaliação (DuckDuckGo, Faroo, Pinda utilizando DuckDuckGo e Pinda utilizando Faroo). Para que o teste pudesse ser imparcial e não influenciar no real impacto de uso da ferramenta desenvolvida, nenhuma apresentação ou descrição prévia de suas características foi mencionada, nem mesmo sua utilidade.

6.1. Parte 1

Na primeira parte do teste, os participantes tiveram que utilizar as ferramentas para mensurar o tempo de mineração visual necessário para descobrir os resultados relacionados ao tema proposto e quantificar o número de resultados considerados relevantes, para todos os quatro elementos de avaliação mencionadas, com o tempo máximo de 1min30seg (um minuto e trinta segundos) para minerar visualmente a página de resultados.

Figura 7. Dados gerados pelo teste para comparar a Mineração Visual de Dados proporcionada pelas ferramentas de busca.



Foram utilizados os temas listados abaixo, respectivamente, em razão de sua heterogeneidade, por possuírem diferentes significados ou interpretações de acordo com o contexto inserido, por apresentarem um volume relativamente grande de resultados (por volta de 100) e por serem de fácil entendimento. Para não "direcionar" os resultados da avaliação, estes temas foram previamente testados nas ferramentas para garantir que nenhuma delas levasse vantagem sobre as outras. No caso da ferramenta Pinda, isso foi garantido verificando-se a ocorrência de grupos imediatos de assuntos sobre o tema proposto, sendo que para as outras ferramentas essa garantia se refletiu na quantidade e heterogeneidade dos resultados isso apresentados.

1. Utilizar o termo *"cuba"* para encontrar resultados sobre a diplomacia cubana;
2. Utilizar o termo *"imagine"* para encontrar resultados sobre a música *Imagine*;
3. Utilizar o termo *"frog"* para encontrar resultados sobre sapos venenosos;
4. Utilizar o termo *"doom"* para encontrar resultados sobre o jogo *Doom*;
5. Utilizar o termo *"turing"* para encontrar resultados sobre a Máquina Universal de Turing.

Inicialmente, os participantes não entenderam muito bem o que realmente era para ser feito, nem como a ferramenta funcionava ou para que servia, mas a necessidade de comparação com as outras ferramentas de busca, infringida pelo teste, fez com que a maioria começasse a explorar os recursos da Pinda e, salvo algumas exceções, todos conseguiram executar o teste sem solicitar auxílio de uso da ferramenta.

Com base no gráfico exibido na Figura 7, percebe-se que, em relação ao tempo gasto pelos participantes para encontrar os resultados mais relevantes aos temas propostos, a Pinda utilizando a API Faroo exige menos tempo de busca do usuário – quase a metade do tempo gasto para procurar resultados análogos no DuckDuckGo. Visualizando o gráfico de cima desta figura, também se pode concluir que, na maioria dos casos o uso dos buscadores DuckDuckGo e Faroo exige menor tempo de busca quando integrado à ferramenta Pinda utilizando suas APIs. Em relação à quantidade de resultados relevantes encontrados, a ferramenta Faroo e a Pinda se mantiveram relativamente estáveis, variando de 4 a 10 resultados identificados pelos usuários. A DuckDuckGo apresentou os maiores resultados, seguido pela Pinda utilizando sua API. Ambas variam de 3 a 15 resultados encontrados.

Com base nestas informações, é razoável perceber que as visualizações da ferramenta Pinda são decisivas ao permitir que os usuários encontrem mais resultados e de forma mais rápida do que os próprios serviços de busca que fornecem estes dados.

6.2. Parte 2

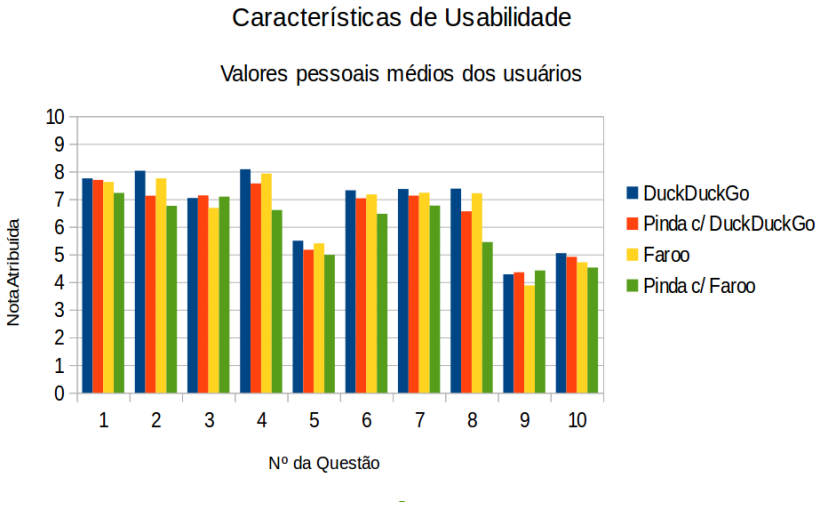
A segunda parte do teste foi baseada em opiniões pessoais dos participantes, que tiveram que pontuar de 0 a 10 a lista de tópicos abaixo relacionados, considerando a utilização independente das quatro plataformas de busca.

1. É fácil entender o que é o sistema e qual seu propósito?
2. A interface do sistema é simples e intuitiva?
3. A apresentação de resultados do sistema é abrangente e condiz com o requisitado?
4. Os resultados são exibidos de forma rápida?
5. O sistema apresenta falhas ou inconsistências?
6. Os resultados apresentados são claros e concisos?
7. O uso do sistema é satisfatório quanto ao seu propósito?
8. O sistema pode ser utilizado por qualquer perfil de usuário?
9. Você substituiria seu sistema de busca atual por esse sistema? Por quê?
10. Com que frequência você costuma utilizar ou utilizaria sistemas de busca, e em quais ocasiões?

O propósito desta etapa era avaliar o real impacto de uso da ferramenta Pinda em relação ao modelo convencional de mecanismos de busca, realizando a avaliação de atributos de usabilidade, entendimento, satisfação e ocorrência de erros, que foram utilizados

para entender as necessidades dos usuários e adaptar a ferramenta de acordo com suas necessidades em comum e gerais. No gráfico da Figura 8, a média aritmética dos valores informados pelos participantes do teste indica que o nível de entendimento do propósito de todas as plataformas é parecido, com notas entre 7 e 8 de entendimento.

Figura 8. Dados gerados pelo teste para comparar características de usabilidade proporcionadas pelas ferramentas de busca.



Em relação à interface do sistema, os participantes citaram, entre outros pontos, que as visualizações utilizadas na Pinda são um pouco difíceis de entender no começo, mas que com o tempo é possível se acostumar com esse modelo de representação. Sobre a abrangência e validade dos dados retornados, a ferramenta Faroo obteve menor pontuação, embora não esteja tão distante dos valores das outras plataformas. Quanto ao tempo de busca despendido pelas plataformas, pode-se atribuir a baixa nota da Pinda utilizando o Faroo em razão do fato da API Faroo limitar o retorno de páginas em no máximo uma página por segundo. Em suma, esta parte do teste serviu para descobrir que a Pinda consegue cumprir com o esperado, agrupando resultados de acordo com assuntos relacionados para otimizar as buscas, embora apresente dificuldades de uso por utilizar representações (visualizações) não usuais para exibir os resultados de busca aos usuários, fator não decisivo para conseguir identificar sua utilidade no teste realizado.

7. Considerações finais

Neste trabalho foi estudado e aplicado o processo de Descoberta de Conhecimento em Bases de Dados em resultados extraídos de mecanismos de buscas por meio de APIs, para auxiliar os usuários na atividade de procurar conjuntos de páginas na *Web* intimamente relacionadas por assuntos em comum, de forma automatizada e sem a necessidade de agrupamento mental de *snippets*.

Durante a fase de Revisão Bibliográfica, constatou-se a importância de se utilizar múltiplas técnicas de Visualização para representar o mesmo conjunto de dados, e como a Coordenação é fundamental para garantir ampla interatividade à ferramenta e, com isso, maior entendimento à respeito do conhecimento extraído.

A partir do teste de avaliação da ferramenta foi possível compreender como realmente se dá a interação dos usuários com a ferramenta Pinda, e com base na medição

de características de processamento e opiniões pessoais dos participantes do teste, concluisse que a ferramenta auxilia consideravelmente a identição otimizada de resultados específicos de busca, que implica em tempo reduzido de mineração visual e melhor identificação de *snippets* relevantes ao tema procurado. As diferenças de desempenho e avaliações das APIs já eram esperadas, onde o menor tempo de mineração visual utilizando a Farro se deve em razão de sua ocorrência de resultados mais amplos e heterogêneos, assim como a maior quantidade de resultados identificados como relevantes à busca realizada.

Há fatores que ainda impossibilitam a implantação de soluções como esta em mecanismos de busca, como interesses financeiros nas colocações dos *snippets* mais relevantes e espaço para propagandas nas páginas de busca, mas a perspectiva é que a otimização de ferramentas como a Pinda, tanto em quantidade de requisições suportadas, de plataformas de utilização e volume de dados retornados, velocidade de acesso, precisão de resultados e *design* de interfaces acessíveis, influenciará automaticamente na mudança de comportamento dos usuários desses serviços de busca. Por isso, considera-se a realização de trabalhos futuros nesse campo de pesquisa, também conhecido como *Search Engine Optimization* (SEO), além da possibilidade de inclusão de módulos destinados à descoberta de conhecimento de perfis de usuários para verificação de sua influência nas buscas e consequente adaptação da ferramenta, ou ainda a Mineração de Dados diretamente da *Internet*, sem a necessidade de utilização de APIs ou serviços externos de busca.

Agradecimentos

De forma geral, agradeço a todos os meus familiares, amigos, colegas e professores por terem, de algum modo, colaborado para o desenvolvimento deste trabalho, direta ou indiretamente. Em especial, agradeço ao meu orientador Danilo M. Eler pela orientação, aos amigos Alisson F. C. do Carmo, João E. M. da Rocha e Priscila A. Macanhã pelo auxílio técnico, à Cintia. Y. Yamada pelo desenvolvimento do logotipo da ferramenta, ao professor Rogério E. Garcia pela ajuda na realização de testes de usabilidade, aos participantes do teste realizado, e à Belquis A. dos Santos por ter sido uma das inspirações. Acredito que a contribuição de todas estas pessoas foi fundamental para as motivações, inspirações e incentivos necessários para o desenvolvimento deste trabalho.

Dedico este trabalho ao meu tio, Wanderley Antunes, por sempre ter sido uma figura carismática, íntegra e generosa, pela amizade sincera e por tudo o mais que não precisa ser dito.

Referências

Boukhelifa, N., Roberts, J., and Rodgers, P. (2003). A coordination model for exploratory multiview visualization. In *Coordinated and Multiple Views in Exploratory Visualization, 2003. Proceedings. International Conference on*, pages 76–85.

Boukhelifa, N. and Rodgers, P. (2003). A model and software system for coordinated and multiple views in exploratory visualization. 2:258–269.

Chakrabarti, S. (2003). *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann, Amsterdam.

Chittaro, L., Combi, C., and Trapasso, G. (2003). Data mining on temporal data: a visual approach and its clinical application to hemodialysis. *Journal of Visual Languages and Computing*, 14(6):591–620.

Clay, B. and Esparza, S. (2009). *Search Engine Optimization All-in-One For Dummies*. –For dummies. Wiley.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). From data mining to knowledge discovery in databases. *American Association for Artificial Intelligence*.

Inselberg, A. and Dimsdale, B. (1990). Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *1st Conference on Visualization (VIS'90)*, pages 361–378. IEEE Computer Society Press.

Levene, M. (2010). An introduction to search engines and web navigation.

McCormick, B. H., DeFanti, T. A., and Brown, M. D. (1987). *Visualization in Scientific Computing*. ACM SIGGRAPH, New York.

Nieto, E. M. G. (2012). Projeção multidimensional aplicada a visualização de resultados de busca textual. Dissertação de mestrado, Universidade de São Paulo, Instituto de Ciências Matemáticas e de Computação, São Carlos, SP - Brasil.

North, C. and Shneiderman, B. (2000). Snap-together visualization: can users construct and operate coordinated visualizations? *International Journal of Human-Computer Studies*, 53(5):715 – 739.

Paulovich, F., Oliveira., M., and Minghim, R. (2007). The projection explorer: A flexible tool for projection-based multidimensional visualization. In *XX Brazilian Symposium on Computer Graphics and Image Processing*, pages 27–36, Washington, DC - USA. SIBGRAPI 2007, IEEE Computer Society.

Schroeder, W. J., Martin, K., and Lorensen, W. (2003). *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. Visualization Toolkit. Kitware, Inc. (formerly Prentice-Hall), 3 edition.

Shimabukuro, M. H. (2004). *Visualizações temporais em uma plataforma de software extensível e adaptável*. Tese de doutorado, Universidade de São Paulo, Instituto de Ciências Matemáticas e de Computação, São Carlos, SP - Brasil.

A. Interface da ferramenta Pinda

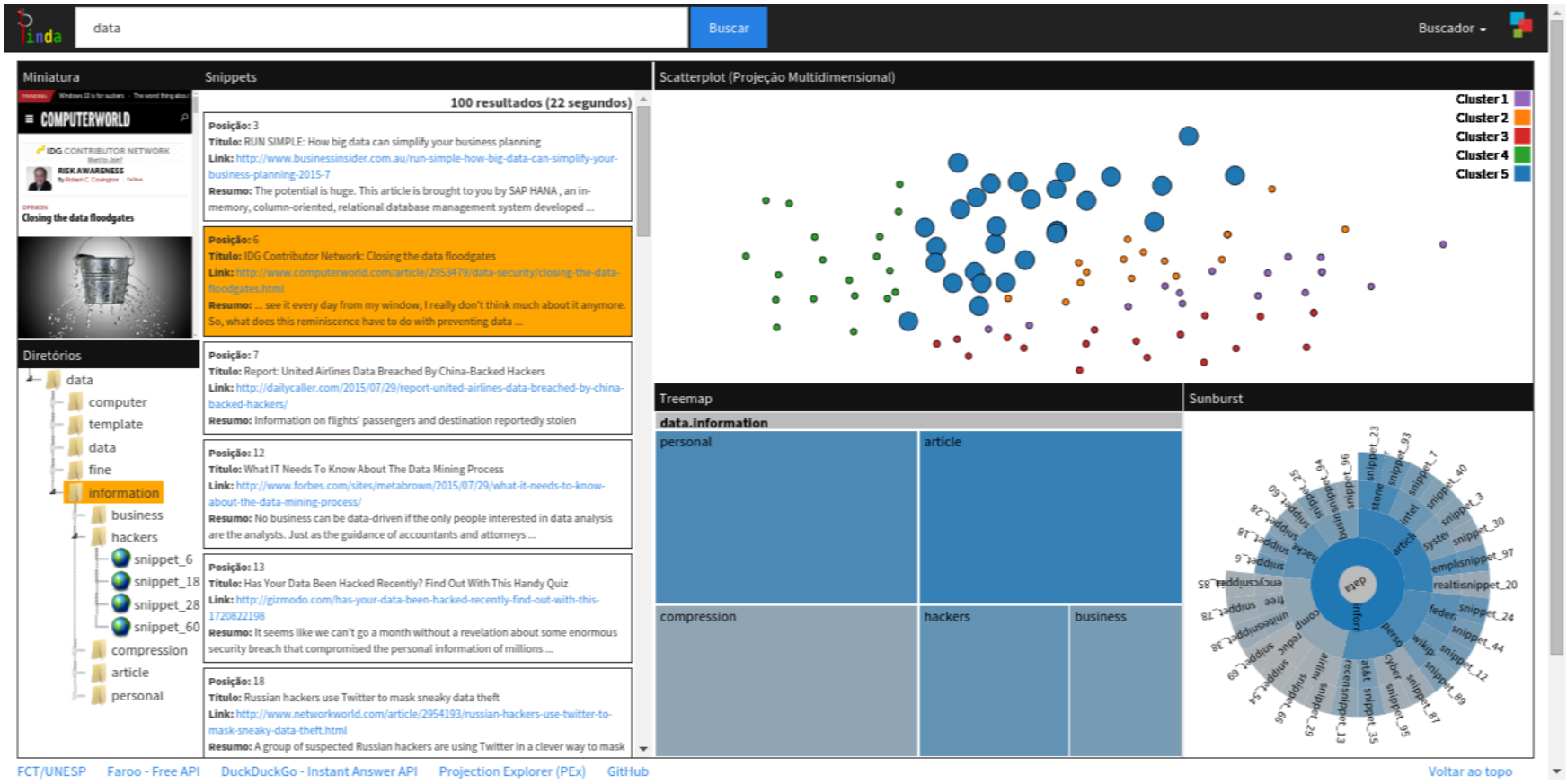


Figura 9. Exemplo de utilização da Pinda.