FERRAMENTA DE BUSCA PARA REDAÇÃO JORNALÍSTICA

Marina Flavia S. Santos, Leandro Luiz de Almeida, Francisco Assis da Silva, Almir Olivette Artero

Faculdade de Informática – Universidade do Oeste Paulista (UNOESTE) – Presidente Prudente – SP – Brasil. almir@unoeste.br

RESUMO

Muitos jornais apresentam dificuldades com o arquivamento das edições impressas. O problema se torna maior, considerando o fato de que este meio de comunicação (mídia impressa) possui uma importância legal e histórica dos fatos e atos que acontecem na localidade onde está situado. Este projeto propõe a implementação de realizar um sistema de armazenamento das edições digitais, preservando a forma com que as mesmas foram impressas. Para auxiliar nas buscas de edições, é utilizado o processo de Mineração de Textos e suas funcionalidades. O projeto propõe ainda um módulo para gerenciar fotografias, além de armazená-las e renomeá-las, pode-se efetuar buscas por meio de palavras-chaves.
Palavras Chaves: XSL, PDF, XML, Mineração de Textos, Processamento de Linguagem Natural, *Stemmer*, *Stopwords*.

SEARCH TOOL FOR JOURNALISTIC WRITING

ABSTRACT

Many newspapers have difficulties with the filing of the printed editions. The problem becomes bigger, considering the fact that the media (print media) has a legal and historical importance of the facts and acts that occur in the locality where it is located. This project proposes the implement to achieve a storage system of digital editions, preserving the way they were printed. To aid in searches of editions, you will use the process of Text Mining and its features. The project also proposes a module for managing photos, store them and rename them, making searches who can be performed by keywords.
Keywords: XSL, PDF, XML, Text Minning, Natural Language Process, Stemmer, Stopwords

1. INTRODUÇÃO

Com o avanço da informatização e a possibilidade de se ter arquivos em tamanhos maiores tornam-se mais complicado armazenar e recuperar esses mais tarde, mesmo porque existe inúmeros arquivos. Baseado nisso e na realidade vivida em muitas redações jornalísticas, surgiu à idéia de desenvolver um software para facilitar a busca de informações, uma forma otimizada e dentro de arquivos mais leves, no caso XML que está indexado com o arquivo PDF um arquivo mais complexo, seguro e relativamente pesado computacionalmente falando. Em paralelo ao armazenamento das páginas a ferramenta contribui para solucionar outro grande problema enfrentado pelas redações: o gerenciamento do arquivo fotográfico. O sistema traz vários benefícios para uma redação jornalística, pois agiliza o processo de busca em edições já publicadas e tornou mais fácil manipular o arquivo fotográfico.

2. MATERIAIS E MÉTODOS

A implementação do aplicativo, consiste na utilização da linguagem de programação Java, com acesso a banco de dados PostgreSQL, geração do arquivo XML através do software InDesign da Acrobat e Técnicas de Mineração de Textos para extrair as informações relevantes e armazená-las no SGBD (Sistema de Gerenciamento de Banco de Dados), para se efetuar as buscas posteriores.

Para o desenvolvimento da ferramenta, foi necessário explorar alguns padrões, por exemplo XML, PDF, entre outros, além de técnicas de Mineração de Texto e Plataforma de desenvolvimento Java.

2.1 PDF

O formato de arquivo .PDF é utilizado, devido a consistência obtida em todos os computadores, ou seja, ele aparece de maneira idêntica em qualquer plataforma em que o mesmo estiver sendo lido ou impresso. E devido a essa capacidade, se tornou praticamente padrão no mundo inteiro.

2.2 Java

A escolha pela plataforma de desenvolvimento Java, foi pelo fato de ser segura, compatível em qualquer ambiente computacional, como por exemplo, Windows e Linux. Além de possibilitar a utilização de recursos que melhor se adaptam ao processo de funcionamento da ferramenta e disponibilizar a ferramenta na Internet. E é gratuito.

2.3 XML

O XML é uma metalinguagem de marcação definida pela W3C (World Wide Web Consortium), que nada mais é do que um consórcio que desenvolve tecnologias interoperáveis (especificações, manuais, softwares e ferramentas) para levar a utilização da rede mundial da internet ao seu potencial pleno, é um fórum para a troca de informações, de comércio, comunicação e de conhecimento coletivo. O que permite criar a sua própria linguagem (baseada em *tags*) seguindo as regras que você mesmo definir.

É um conjunto de padrões para troca de informações de forma estruturada, descreve e manipula documentos oferecendo uma estrutura de árvores para todas as aplicações, o que torna a busca mais eficiente.

O conteúdo do elemento não está limitando a apenas texto; os elementos podem conter outros elementos que por sua vez podem ter textos ou elementos e assim por diante.

Um elemento que esteja embutido em outro elemento é chamado de filho. O elemento no qual está embutido é o seu pai. No exemplo mostrado pela Figura 01, o elemento nome possui dois filhos: os elementos pnome e snome, onde esse último é o pai dos dois elementos [1].

```
<nome>

<pnome>Marina</pnome>
  <snome>Flavia</snome>
</nome>
```

Figura 1: Exemplo de um XML

Não é acidental que os documentos XML sejam árvores. As árvores são flexíveis simples e poderosas. Em particular, elas podem ser usadas para colocar qualquer estrutura de dados em série.

2.3.1 Formato INX

Em um documento, tem somente uma raiz, ou seja, todos os elementos do documento precisam ser filhos de um único elemento.

O arquivo com extensão INX nada mais é do que uma biblioteca de interface do XML, sendo o INX um dialeto baseado em XML, é obrigado a ter uma *tag* envolvendo todas as *tags* do documento. Ela não recebe atributos, portanto ela existe apenas para que o XML seja interpretável [6]. **API DOM**

No modelo definido na Figura 2, o documento XML é armazenado na memória em formato de árvores de nodos, todos descendendo de uma raiz.

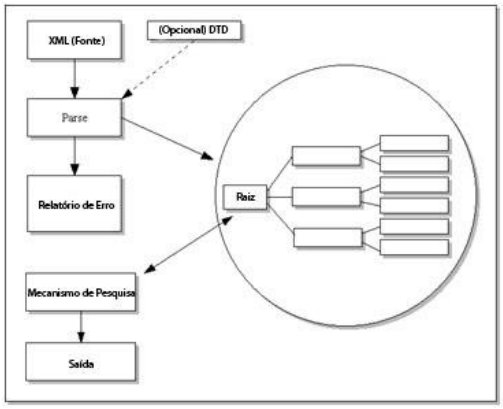

Figura 2: Manipulação de um DOM

Para cada procura ou tipo de manipulação, sempre inicia pelo elemento raiz e vai seguindo a hierarquia. Como todas as informações estão disponíveis na memória, é possível correlacionar e combinar informações como desejar.

A API para trabalhar com DOM é fornecida pela recomendação World *Wide Web Consortiun* (W3C).

A unidade básica considerada no DOM é o Nodo. A interface ***Node*** é implementada por todas as diferentes subcategorias de ***Node***. Esta forma de representação é muito interessante para as linguagens orientadas a objetos como Java. Todos os tipos de Node possuem uma interface correspondente em Java. Além de vários subtipos de ***Node***, DOM define interfaces de coleções de ***Nodes*** como o ***NodeList*** e o ***NamedNodeMap***. DOM também especifica uma interface ***DOMException*** que pode ser utilizada para comunicar erros de exceção [3]. Basicamente, o documento XML é transformado em um DOM constituído de objetos que implementam interfaces. Cada parte do documento é transformada em um objeto, e as conexões entre os objetos refletem a hierarquia do documento.

2.4 Mineração de Textos

Mineração de Textos é um conjunto de técnicas e processos que obtém informação de qualidade a partir da linguagem natural que

descobrem conhecimento inovador nos textos. Com a mineração de texto pode-se extrair informação relevante de uma grande base de textos, sem a necessidade de lê-los previamente. Outra utilização para encontrar o que se deseja. Seguindo a mesma tendência da área de mineração de dados, classificação automática de textos e agrupamento por semelhança são outras funcionalidades comumente utilizadas.

Um processo inteiro de mineração de texto consiste em um mecanismo de coleta, uma etapa de pré-processamento, um mecanismo de indexação, aplicação do algoritmo e finalmente a análise dos resultados.

A seguir as etapas que compõem a Mineração de Textos serão apresentadas, mostrando conceitos que serviram como base as diferentes metodologias que podem ser empregadas e a interpretação dos resultados.

2.4.1 Tipos de Abordagens dos Dados

Existem duas formas de abordagem dos dados textuais. A análise semântica baseada na funcionalidade dos termos e a análise estatística baseada em freqüência. Essas abordagens podem ser utilizadas sozinhas ou em conjunto. Para o desenvolvimento da ferramenta foi utilizada a análise semântica, por atender melhor o que é necessário. Essa técnica avalia a seqüência dos termos no contexto da frase, para a correta identificação da função de cada termo. A análise semântica utiliza fundamentos e técnicas baseados no processamento de linguagem natural.

O emprego desse tipo de análise justifica-se pela melhoria em qualidade da Mineração de textos quando incrementado de um processamento lingüístico.

2.4.2 Preparação dos Dados

A preparação dos textos é a primeira etapa do processo de descoberta de conhecimento em textos. Seleciona os dados que constituirão a base de textos de interesse e o trabalho inicial para tentar selecionar o núcleo que melhor expressa o conteúdo dos textos, ou seja, toda a informação que não for relevante será desprezada. Além de promover uma redução dimensional, esta etapa tenta identificar similaridades em função da morfologia ou do significado dos termos, de modo a aglomerar sua contribuição.

2.4.3 *Stopwords*

É a tentativa de retirar tudo que não constitui conhecimento nos textos. Nessa etapa, uma lista contendo palavras a serem descartadas é formada. Esse conjunto de palavras é chamado de *stopwords* (conhecido também como *stoplist*). *Stopwords* são palavras consideradas não relevantes na análise de textos; são palavras ou termos que não traduzem a essência dos textos. Normalmente, isso acontece por serem palavras auxiliares ou conectivas (preposições, pronomes, artigos, e outras classes) e que não fornecem nenhuma informação discriminativa na expressão do conteúdo dos textos.

2.4.4 *Stemmer*

O processo de *stemming* é realizado pela extração de cada palavra do texto, considerando a palavra isoladamente e tentando reduzi-la a sua provável palavra raiz. Os algoritmos de *stemming* correntes não utilizam informações do contexto para determinar o sentido correto de cada palavra, e realmente essa abordagem não parece ajudar. Não são freqüentes casos em que o contexto ajuda no processo de *stemming*, e a maioria das palavras pode ser considerada como apresentando um significado único. Os erros resultantes de uma análise de sentido imprecisa das palavras não compensam os ganhos que possam ser obtidos pelo aumento da precisão do *stemming*.

2.5 Adobe Indesign

Este software cria documentos em formato próprio (com extensão .indd), editável, que posteriormente pode ser exportado para o formato PDF, XML, entre outros formatos específicos de impressão. Embora o InDesign permita gerar e distribuir os documentos em sua forma digital, o documento final normalmente é utilizado para a geração de matrizes para a posterior impressão. Esta fase em que o InDesign é utilizado para a criação dessa matriz é conhecido como *pré-press*, pré-impressão.

3. FUNCIONAMENTO DA FERRAMENTA

O funcionamento da ferramenta é realizado por 2 módulos distintos, que serão detalhados a seguir:

3.1 Módulo de gerenciamento fotográfico

O módulo de gerenciamento fotográfico funciona da seguinte forma: o funcionário cadastra as imagens na ferramenta, fornecendo algumas informações. O usuário digita o conteúdo que pretende pesquisar. A ferramenta faz as buscas necessárias dentro do SGBD. O resultado encontrado é mostrado para o usuário, que escolhe a imagem que melhor se adéqua, e sugere algum tratamento se for o caso. O funcionário faz o tratamento necessário e disponibiliza para a redação.

A Figura 3 mostra o esquema gráfico do módulo de gerenciamento fotográfico.

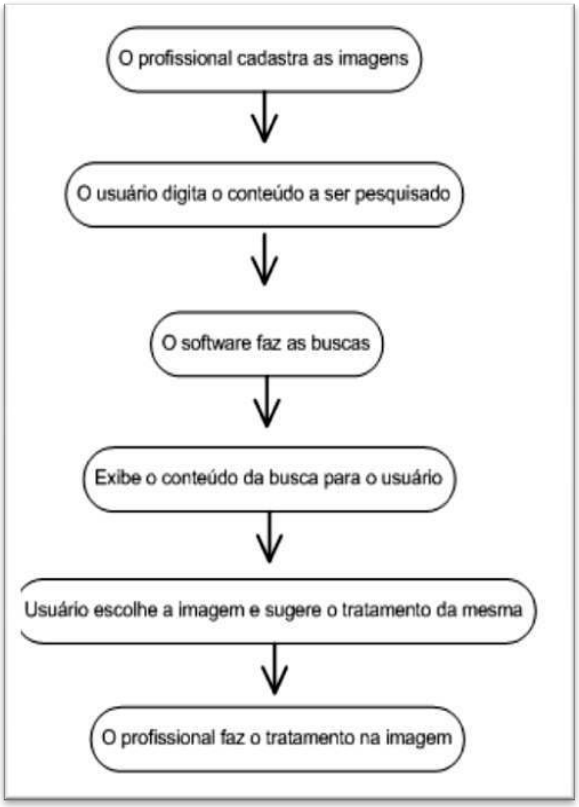

Figura 3: Módulo Gerenciador Fotográfico

3.2 Módulo de gerenciamento de edições

O módulo de gerenciamento de edições funciona da seguinte maneira: o jornal é editado no aplicativo Adobe InDesign como de costume, gerando o arquivo com extensão .indd, este é exportado para PDF e XML. Após esse processo de exportação os arquivos são abertos e lidos pela ferramenta, e informações como data de publicação e caderno são fornecidas, além do caminho do arquivo PDF e XML, para serem armazenadas no gerenciador de banco de dados. Internamente a ferramenta percorre a *tag pcnt*, que é onde encontra o texto que interessa para o usuário; nesse momento é aplicado o algoritmo de *stopwords*, ou seja, o algoritmo que retira as palavras que não apresentam significados relevantes, ou seja, pronomes, artigos e preposições. O próximo passo é retirar o radical das palavras, ou seja, aplicar o algoritmo que extrai do radical da palavra analisada, esse processo é denominado de *stemming*, que é a retirada dos sufixos, prefixos e vogais temáticas.

Depois de realizar essas tarefas, finalmente as informações são acrescentadas no SGBD. Depois de armazenadas as informações o usuário digita o conteúdo a pesquisar e as buscas são realizadas retornando os possíveis resultados ao usuário.

A Figura 4 mostra o esquema gráfico do funcionamento da ferramenta.

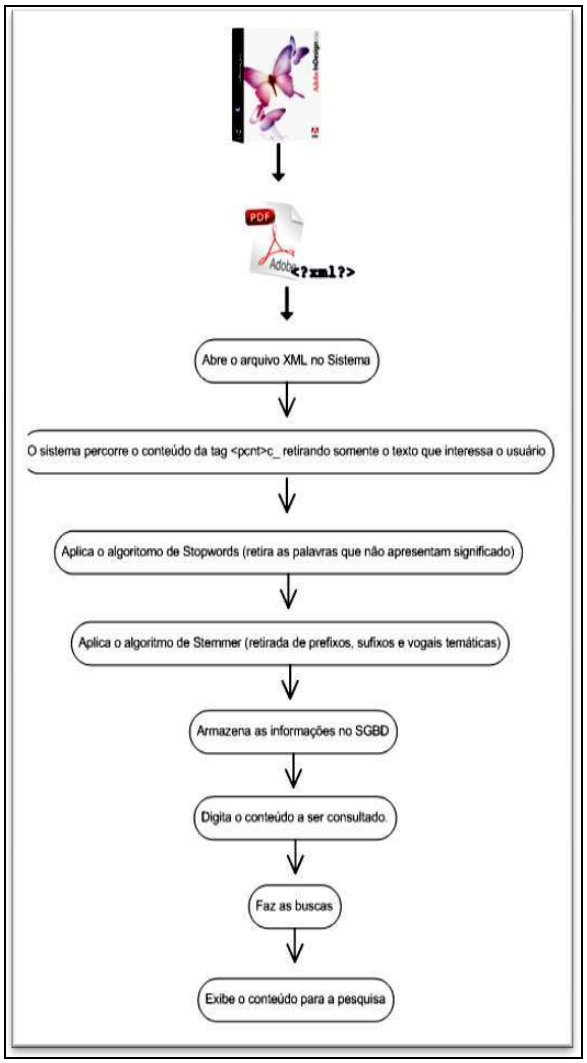

Figura 4: Módulo Gerenciador de Edições

4. EXPERIMENTOS E RESULTADOS

Alguns testes foram realizados tanto com o Módulo Gerenciador Fotográfico, quanto com o Módulo Gerenciador de Edições. Nesses testes pode-se observar a melhoria que a ferramenta pode trazer. Como por exemplo, automatização das pesquisas tanto nas imagens, quanto nas edições de jornais já publicadas.

4.1 Módulo Gerenciador Fotográfico

Com as imagens e suas respectivas palavras chaves já cadastradas em um SGBD, as buscas são efetuadas.



Figura 5: Exemplo Banco de Imagens

Teste

Palavra Digitada: Natureza

Resultado:

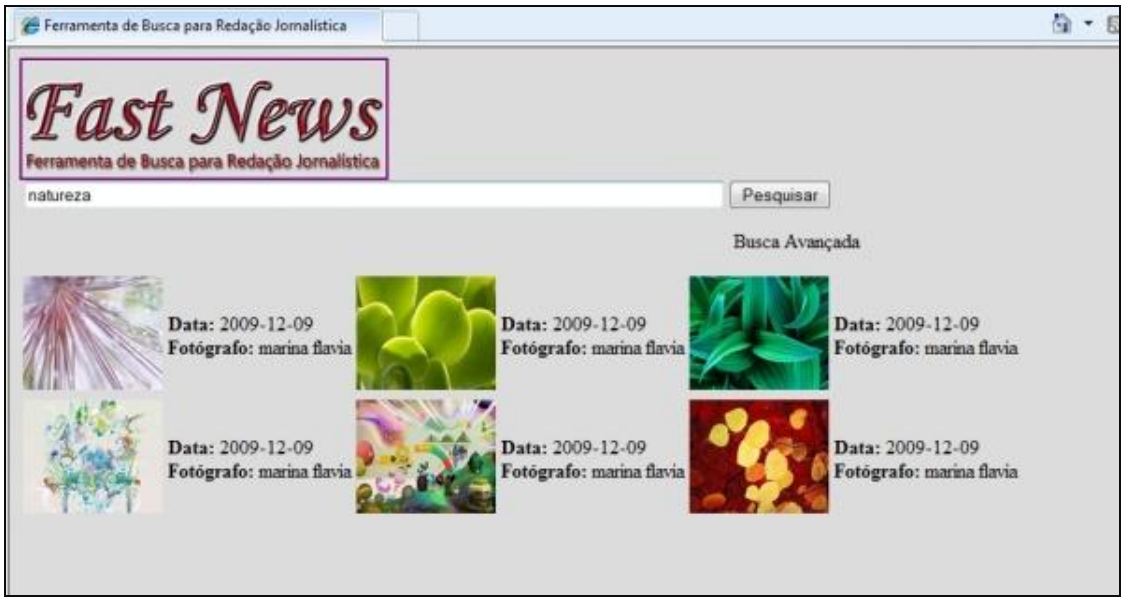

Figura 6: Imagens retornadas na pesquisa

Sem a ferramenta essas imagens são procuradas uma a uma pelo nome do arquivo, pois se encontram em diretórios do sistema operacional, com a ferramenta a busca ocorre por palavras chave, onde todo conteúdo relacionado é retornado ao usuário.

4.2 Módulo Gerenciador de Edições

Após as edições de jornais estarem gravadas no SGBD, a busca por matérias podem ser executadas.

As palavras são armazenadas no SGBD em forma de radicais, ou seja, são retirados os artigos, as preposições, os pronomes, os sufixos e os prefixos, para que as palavras volte a sua origem, com isso o SGBD fica mais leve como mostra na Figura7.

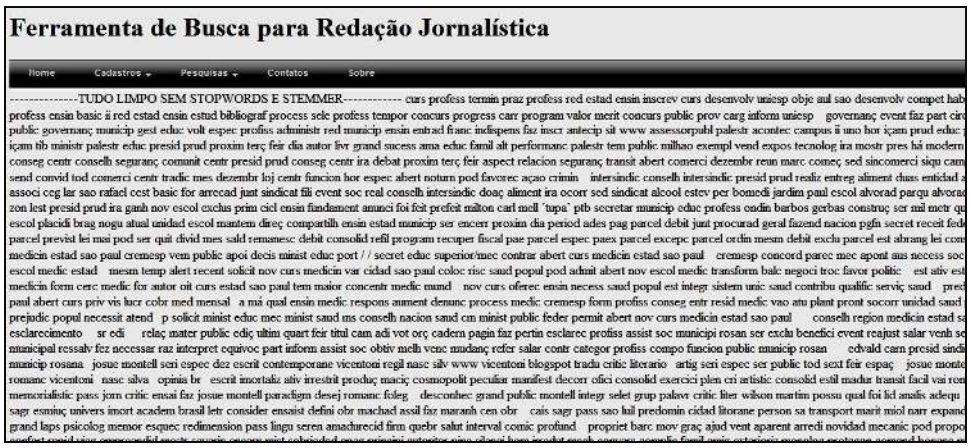

Figura 7: Radicais armazenados no SGBD

Teste 1

Palavra Digitada: Natureza

Resultado:

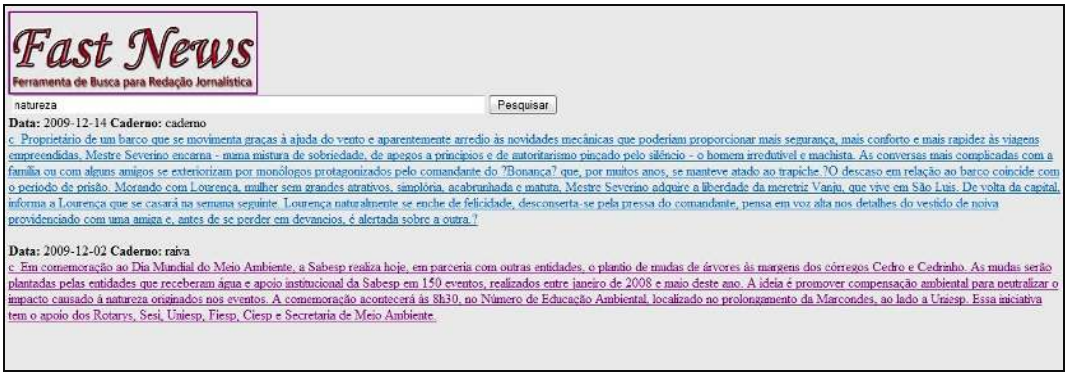

Figura 8: Resultado do teste 1

Teste 2

Palavra Digitada: Sabesp

Resultado:

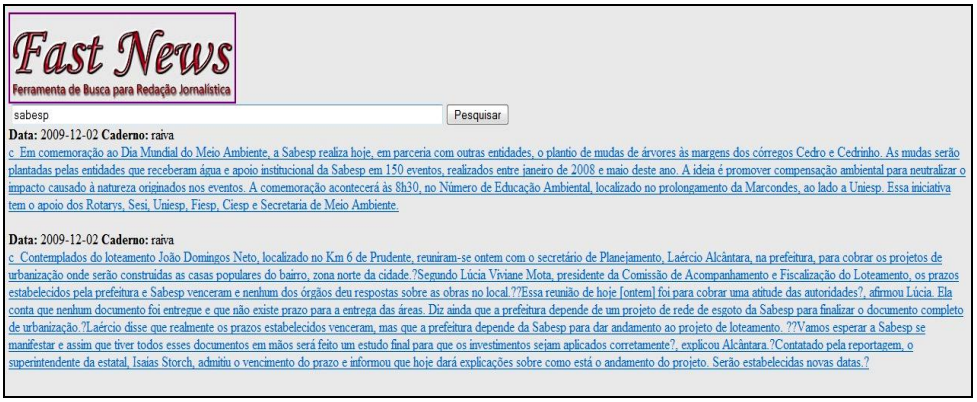

Figura 9: Resultado do teste 2

Após exibir os resultados para o usuário, ele escolhe a matéria que mais interessa, e com apenas um clique, ele tem acesso a notícia do mesmo jeito que a mesma foi publicada, a página de jornal na íntegra.

5. CONCLUSÕES E TRABALHOS FUTUROS

A ferramenta desenvolvida neste projeto visa a busca automatizada por notícias dentro de uma redação jornalística, facilitando muito a realização de uma grande parte das atividades rotineiras dos profissionais da área, pois não precisam executar a exaustiva busca manual, pesquisando as numerosas edições jornalísticas gravadas em DVD's.

Os experimentos realizados demonstraram um grande ganho em relação ao desempenho do sistema, pois as fotografias e temas contidos em edições estão organizados em um SGBD.

Como trabalhos futuros sugere-se:

- a elaboração de buscas de imagens por conteúdo;
- a implementação de busca para qualquer tipo de arquivo, não limitando apenas ao XML;
- utilização de modelos inteligentes de busca, fazendo com que haja o aprendizado pela ferramenta;
- aplicação de processos de inferência *Fuzzy* para a classificação dos resultados obtidos.

6. REFERÊNCIAS

APIs Java para XML, Disponível em: <http://www.inf.ufrgs.br/procpar/disc/inf01008/trabalhos/sem01-1/t2/apis_xml_java/#DOMExemplo> Acessado em 10 de abril de 2009.

FERNEDA, E. **Recuperação de Informação: Análise sobre a contribuição da Ciência da Computação para a Ciência da Informação.** 2003. 147f. Tese (Doutorado em Ciências da Comunicação) – Escola de Comunicação e Artes, Universidade de São Paulo.

Grupo Java, Disponível em <http://www.guj.com.br> Acessado em: 10 de abril de 2009.

InDesign, Disponível em <http://indesignbrasil.blogspot.com/2006/10/formato-inx-export-e-na-janela-de.html>. Acessado em 10 de abril de 2009

MARCHAL, Benoit. **XML Conceitos e Aplicações**, São Paulo: Editora Berkeley, Ed. 1, 2000.

REZENDE, Solange Oliveira. **Sistemas Inteligentes – Fundamentos e Aplicações**. Ed. Mande, 2003.

Stemmer, Disponível em: <http://snowball.tartarus.org/algorithms/portuguese/stemmer.html>Acessado em 20 de julho de 2009.

Stemmer, Disponível em: <http://www.frb.br/ciente/2006_2/BCC/CC.GOMES.etal.F1%20_Rev.%20RC%2023.01.07.pdf> Acessado em 20 de julho de 2009.

Stopwords, Disponível em: <http://snowball.tartarus.org/algorithms/portuguese/stop.txt > Acessado em 20 de julho de 2009.

VELOSO, Renê Rodrigues. **Java e XML: Processamento de documentos XML com Java**. 2ª edição, 2007.