InqExpert: uma aplicação para tratamento textual das Inquirições de 1258

Luís Mendes Gomes, Hélia Guerra, Luís Pires, Luís Furnas
Centro de Matemática Aplicada e Tecnologias de Informação
Departamento de Matemática, Universidade dos Açores
lmg@uac.pt, helia@uac.pt, lpires@uac.pt

Resumo

Neste artigo, vamos apresentar a construção de uma aplicação de software particularmente vocacionada para a análise quantitativa do texto das Inquirições de 1258, designada InqExpert, que permite obter, de forma rápida e rigorosa, informação acerca de vários domínios de estudo (e.g. metrologia). Este texto representa na historiografia portuguesa um paradoxo: investigadores que muitas vezes se queixam de falta de fontes recuam perante tal extensão de informação. Para ajudar a resolver este paradoxo o InqExpert foi desenvolvido à medida das necessidades dos historiadores, mas também foi preparado para suportar evoluções no sentido de acomodar técnicas do TextMining.

Palavras-chave: Bases de Dados Documentais, Bases de Dados XML, Análise Quantitativa de Texto, Engenharia de Software, Inquirições de 1258.

Abstract

In this paper, we present the construction of a software application, particularly tailored for the quantitative analysis of Royal Inquiries of 1258, named InqExpert, which enables us to look for information, in several domains (e.g. metrology), over all its corpus in a fast and rigorous way. This corpus represents in itself a paradox in the Portuguese historiography: researchers that usually complaint about the lack of sources then move backwards in presence of a huge amount of information. To help them in solving this paradox, the InqExpert was developed to fit the usual historian needs but also to support releases that integrate several advanced techniques in Text Mining.

Keywords: Document-Oriented Databases, XML Databases, Quantitative Text Analysis, Software Engineering, Royal Inquiries of 1258.

1 Introdução

As Inquirições de 1258 constituem um rico manancial de informação para o estudo do Portugal ducentista sob um grande número de aspectos. As suas actas estão disponíveis, sob a forma impressa, desde o século XIX. No entanto, a respectiva vastidão é desencorajadora de estudos aprofundados. Correspondendo a estrutura do corpus das Inquirições de 1258 a três níveis (alçada, unidade administrativa ou julgado, e paróquia), podemos considerar três grandes tipos de necessidades básicas dos utilizadores: determinar a existência de uma

determinada informação, avaliar a importância quantitativa dessa informação e distribuir as ocorrências pela estrutura topográfica referida.

Pela experiência retirada do trabalho conjunto entre historiadores e engenheiros de software, particularmente no âmbito do projecto INQ1258, constatamos que as aplicações de produtividade geral mais utilizadas pelos primeiros (e.g. MS Word) não permitem satisfazer de forma eficiente, pelo menos, aquelas necessidades básicas. Mesmo recorrendo a outras aplicações um pouco mais robustas, nomeadamente as que utilizam expressões regulares (e.g. OpenOffice), não conseguimos mais do que localizar de forma eficiente ocorrências de padrões textuais - a construção e aplicação de expressões regulares não é prática regular entre historiadores e a sua aprendizagem apresenta dificuldades e resistências. Há outras ferramentas mais robustas no tratamento de texto com expressões regulares, que incluem tratamento quantitativo elementar do corpus (e.g. PowerGREP), mas que já são bastante diferentes das supracitadas em termos das funcionalidades que oferecem e ao seu acesso e, por isso, não permitem uma transição fácil e rápida pelos historiadores. Mais importante, é o facto destas aplicações não considerarem uma estrutura específica para um determinado corpus e, assim, no caso das Inquirições de 1258, não permitirem, por exemplo, saber: em que paróquias ocorre o padrão X; se o padrão Y ocorre nas unidades administrativas A, B e C e quantas vezes.

Existe, ainda, outra fonte de aplicações de pesquisa de padrões em texto proporcionadas pelo projecto Lucene (vide [Luc]), bem como em projectos congéneres, nomeadamente o GATE (vide [Gate]) e o LingPipe (vide [LingPipe]) (ou, para um referência integrada, vide [Konchady 2008]) substancialmente mais completas e evoluídas, no sentido de permitirem análises textuais, em vários suportes (e.g. Web), usando implementações de algoritmos proporcionados pelo Text Mining, pela Processamento de Linguagem Natural, entre outras. Neste caso, a sua utilização pelos historiadores não só exigia um enorme esforço de transição (à partida, desencorajador) mas, também, competências técnicas de informática e de análise de texto que não possuem, em parte, neste último caso, por não terem conhecimentos apropriados de estatística descritiva e inferencial.

Tendo em conta tudo isto, optamos por produzir uma aplicação Java desktop específica, à medida das necessidades dos historiadores das Inquirições de 1258, sendo escalável para o estudo das Inquirições de 1220 e para integrar outras implementações de algoritmos do Text Mining, que sejam apropriados à grande diversidade de estudos potenciados por esta obra. Esta aplicação segue uma abordagem diferente daquela usualmente seguida pela Recuperação de Informação (Information Retrieval) que é baseada numa estrutura de dados matricial cujas linhas representam os documentos e as colunas os padrões a pesquisar. Como nas Inquirições de 1258 (e nas Inquirições de 1220), em cada livro existem dezenas de unidades administrativas e, em cada unidade administrativa, existem centenas de paróquias, optamos por não utilizar esta estrutura de dados e considerar a pesquisa sobre todo o corpus (em formato texto) anotado com a sua estrutura topográfica e referencial (livros, páginas e colunas), permitindo que os historiadores possam trabalhar na aplicação sobre uma única estrutura documental, tal como habitualmente fazem com as edições em papel. E, por isso, não utilizamos a API disponibilizada pelo Lucene e por outros projectos congéneres.

Neste artigo, na secção 2, apresentamos as várias etapas que envolvem a preparação do corpus das Inquirições de 1258. Na secção 3, fazemos o levantamento dos requisitos e apresentamos o respectivo diagrama de casos de utilização UML e, em seguida, na secção 4, propomos um diagrama de classes UML que resulta da análise dos requisitos conceptuais da aplicação. Na

secção 5, discutimos algumas soluções técnicas para a implementação do diagrama de classes da secção 4 usando o Java. Na secção 6, apresentamos a interface gráfica e alguns exemplos simples de utilização. E, finalmente, na secção 7, discutimos algumas limitações da solução e apontamos alguns aspectos possíveis para a sua evolução.

2 Preparação do Corpus das Inquirições de 1258

Para conseguir uma versão digital do corpus das Inquirições de 1258, no formato ASCII, para posterior anotação topográfica e referencial, seguimos a metodologia DAR em [Marinai 2008]. Começamos por digitalizar as suas 1145 páginas e, após um tratamento digital a cada uma das imagens digitalizadas, utilizando o Adobe PhotoShop, produzimos uma aplicação Web para implementar um eBook topográfico e referencial, no formato PDF, que está actualmente disponível em http://www.uac.pt/uacweb (com acesso restrito). Em seguida, usamos o Abby FineReader para transformar todo o conteúdo das imagens digitalizadas para texto no formato ASCII. Como este texto está escrito em português medieval (muito próximo do Latim) e numa impressão mecânica do séc XIX, o resultado da leitura do OCR produziu centenas de erros por página, o que inviabilizou de imediato a sua incorporação directa na aplicação InqExpert.

Por isso, tentamos automatizar (parcialmente) a correcção do corpus, no formato ASCII, proveniente do OCR para, assim, minimizar o esforço da intervenção da revisão humana por um especialista. Uma abordagem ensaiada consistiu em gerar automaticamente um dicionário, recorrendo a vários scripts em Perl, a partir das primeiras 10 páginas de cada alçada, que foram previamente corrigidas. E, depois, produzimos scripts em Perl para fazer uma correcção ortográfica automática dos 5 livros (ou alçadas) que constituem as Inquirições de 1258. Não se registaram resultados apreciáveis, devido, em parte, à enorme diversidade vocabular daquele documento.

Como o documento das Inquirições de 1258 possui uma estrutura explícita está particularmente enquadrado na motivação para a utilização da tecnologia de anotação XML, seguindo uma abordagem orientada aos documentos (de texto), no chamado processamento de bases de dados documentais XML [Harold 2002]. Por isso, definimos duas gramáticas DTD (vide [Ramalho 2002]): uma corresponde à anotação topográfica e a outra corresponde à anotação referencial; que nos permitiu validar cada uma das partes do documento à medida que iam chegando da revisão, que passamos a descrever.

O documento das Inquirições de 1258 está dividido em 5 alçadas, numeradas de 1 a 5, que, por sua vez, estão divididas em unidades administrativas que, por sua vez, estão divididas em paróquias. A alçada 1 é constituída por 21 unidades administrativas e 415 paróquias; a alçada 2 é constituída por 20 unidades administrativas e 466 paróquias; a alçada 5 é constituída por 10 unidades administrativas e 274 paróquias; e as unidades administrativas e paróquias para as alçadas 3 e 4 não estão definidas.

DTD para a anotação topográfica

```
<?xml version=”1.0” encoding=”UTF-8”?>
<!ELEMENT inq1258 (alcada)+ >
<!ELEMENT alcada (prologo, unidade_administrativas) >
<!ELEMENT unidade_administrativas (nome_ua, paroquias) >
<!ELEMENT paroquias (nome_paroquia, texto) >
<!ELEMENT prologo (#PCDATA) >
<!ELEMENT nome_ua (#PCDATA) >
<!ELEMENT nome_paroquia (#PCDATA) >
<!ELEMENT texto (#PCDATA) >
```

O documento das Inquirições de 1258 segue uma estrutura standard, i.e. cada livro, numerado de 1 a 5, é constituído por páginas numeradas (sequencialmente, i.e. o número da primeira página de um livro é o número imediatamente a seguir ao último número da página do livro anterior) e cada página está dividida em duas colunas.

DTD para a anotação referencial

```
<?xml version=”1.0” encoding=”UTF-8”?>
<!ELEMENT inq1258 (alcada)+ >
<!ELEMENT alcada (paginas)+ >
<!ELEMENT paginas (numero, colunas+) >
<!ELEMENT colunas (coluna_a, coluna_b) >
<!ELEMENT coluna_a, coluna_b (#PCDATA) >
<!ELEMENT numero (#PCDATA) >
```

Para incorporar o corpus das Inquirições de 1258, no formato ASCII, no InqExpert é necessário dividir o texto integral em 5 partes, tantas quantas as suas alçadas, em que cada uma destas partes é representada por 3 ficheiros: um ficheiro com o texto sem qualquer marcação, um ficheiro com o texto com a anotação topográfica e um ficheiro com texto com a anotação referencial. Antes desta incorporação, cada ficheiro anotado é submetido a um processador XML, que verifica a sua validade segundo as gramáticas DTD definidas acima.

3 Levantamento de requisitos: diagrama de casos de utilização

Nesta secção, vamos estabelecer quais os requisitos a proporcionar, assumindo que o utilizador típico do InqExpert é um investigador e/ou estudante em História Medieval que deseja recolher informação quantitativa sobre vários domínios (e.g. Metrologia). De acordo com especialistas nas Inquirições de 1258, os requisitos funcionais básicos para o InqExpert devem satisfazer as seguintes necessidades:

- associar a cada interacção com o programa uma sessão de trabalho, identificada, pelo menos, pelo utilizador, pela data e pela hora – uma sessão de trabalho é composta por pesquisas de palavras no corpus;
- permitir seleccionar partes do corpus (integral) para constituir o corpus de uma sessão de trabalho;
- pesquisar palavras somente em determinadas unidades topográficas, i.e. alçadas, unidades administrativas e paróquias, incluídas no corpus da sessão;
- pesquisar palavras de forma exacta e por aproximação baseada no prefixo, infixo e sufixo;

- pesquisar palavras usando conectivos lógicos – digamos fazer pesquisas compostas;
- localizar topograficamente cada ocorrência referente a cada pesquisa elementar ou composta no corpus;
- gerar quadros estatísticos, e respectivos gráficos, provenientes das pesquisas;
- guardar o resultado de cada uma das pesquisas de forma a recuperar para consultar posterior (i.e. um arquivo).

Sabendo que esta classe de utilizadores privilegia a facilidade de instalação e operação destas aplicações de software específicas e, também, é muito exigente quanto ao seu rigor no desempenho operacional, é fundamental incorporar todas as funcionalidades supracitadas numa aplicação de software de utilização individual (i.e. desktop) cuja instalação seja possível em qualquer sistema operativo e possua uma interface gráfica funcional e apelativa.

Considerando esta lista de requisitos funcionais, vamos apresentar um diagrama de casos de utilização UML (vide e.g. [Fowler 2003]) para descrever graficamente os intervenientes, as funcionalidades da aplicação que devem ser disponibilizadas aos intervenientes e as suas dependências.

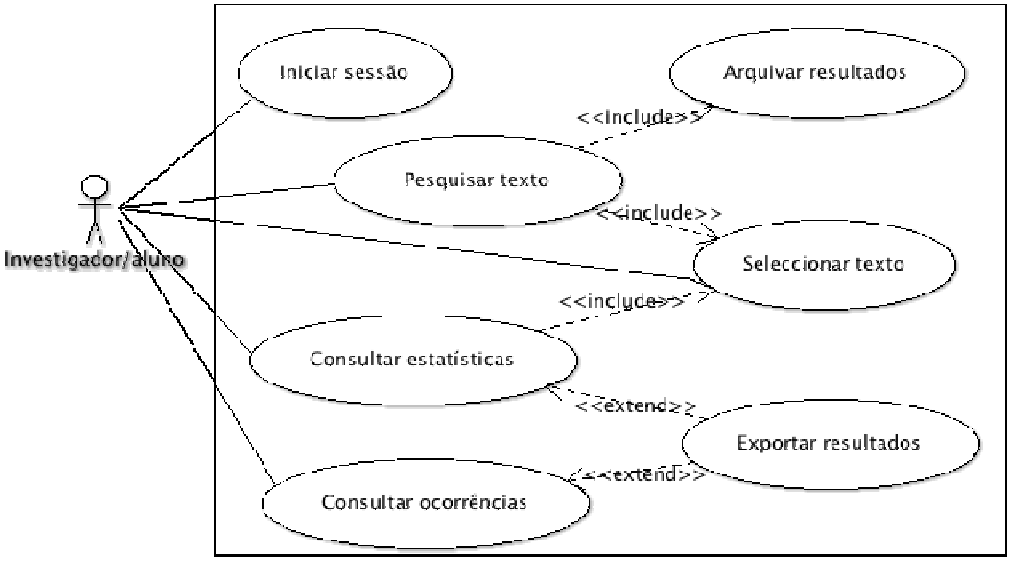

Figura 1: Diagrama UML de casos de utilização

Da observação da Figura 1, podemos dizer que identificamos um único perfil de potenciais utilizadores: o dos investigadores/alunos com interesses em História Medieval. Cada utilizador pode executar as seguintes tarefas: iniciar uma sessão de trabalho, seleccionar o corpus para a pesquisa, pesquisar padrões no corpus (texto) seleccionado e consultar as ocorrências e as estatísticas resultantes. O sistema arquiva os resultados de cada pesquisa em ficheiro e permite a sua exportação, bem como das estatísticas para outros formatos (e.g., xlsx). Esta visão de casos de utilização, servirá de base para o modelo de desenho conceptual apresentado na secção seguinte.

4 Diagrama de classes

Considerando o diagrama de casos de utilização introduzido na secção anterior, vamos, em seguida, como aconselham as boas práticas da engenharia de software, definir um modelo para o desenho do sistema, recorrendo a um diagrama de classes UML (vide e.g. [Fowler 2003]).

Como podemos observar, o diagrama na Figura 2 contém várias classes e associações que vamos passar a descrever. A classe Sessão caracteriza uma sessão de trabalho realizada por um interveniente (e.g. investigador/aluno de História).

Como referido anteriormente, o corpus é constituído por alçadas, que por sua vez são formadas por unidades administrativas que são divididas em paróquias. Esta hierarquia topográfica está representada pelas classes Corpus, Alçada, UnidAdministrativa e Paróquia e respectivas composições. Desta forma, qualquer objecto destas classes é considerado uma parte topográfica do corpus e representamos esta designação generalizada pela classe ParteTopográfica. Assim, a combinação de várias partes topográficas constituí a parte do corpus para análise em cada sessão.

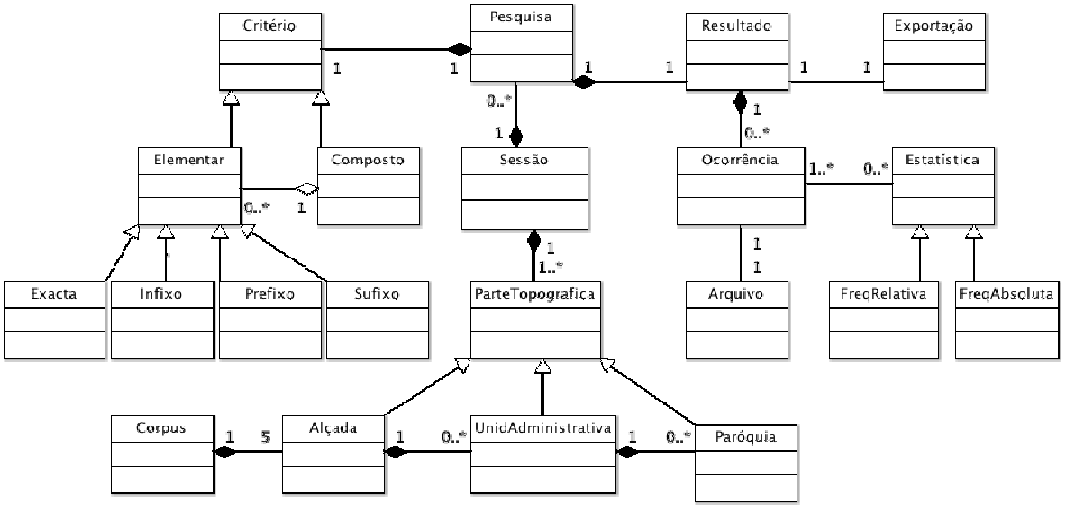

Figura 2: Diagrama de classes para a estrutura do InqExpert

Uma sessão é caracterizada por um conjunto de pesquisas sobre a(s) parte(s) topográfica(s) seleccionada(s). Cada pesquisa é composta por um critério (objecto da classe Critério) que caracteriza a expressão que se pretende pesquisar e por um resultado que indica o número de ocorrências da expressão no texto e a respectiva localização. Esta expressão pode ser elementar, ou seja, uma palavra a procurar de quatro formas distintas e não disjuntas: exacta (classe Exacta), como prefixo (classe Prefixo) ou sufixo (classe Sufixo), ou como infixo (classe Infixa); ou composta que consiste numa combinação linear de expressões elementares usando os conectivos lógicos conjunção, disjunção e negação.

Cada objecto da classe Resultado representa uma lista de ocorrências da expressão de pesquisa no corpus (classe Ocorrência) que pode ser guardada num ficheiro (classe Arquivo) ou exportada para outras aplicações (classe Exportação). É possível, seleccionar várias (ou

mesmo todas as ocorrências encontradas) e efectuar estatísticas descritivas elementares (classe Estatística), nomeadamente frequências absolutas (classe FreqAbsoluta) e relativas (classe FreqRelativa) da(s) palavra(s) a pesquisar.

5 Implementação Java: o problema da eficiência das pesquisas

Nesta secção, descrevemos, resumidamente, alguns aspectos da implementação do InqExpert em Java. Nomeadamente, a descrição de dois algoritmos para tornar mais eficiente a pesquisa no corpus, tendo em conta a exiguidade dos recursos de memória que são, geralmente, dispensados pela CPU para a sua execução.

Um dos maiores problemas no que diz respeito ao corpus é o tamanho dos ficheiros das alçadas. Isto é, manter o texto guardado na memória não é viável, visto que retira recursos de CPU. Por isso, no Algoritmo 1, apresentamos uma solução para tornar a leitura do corpus para a pesquisa menos exigente no consumo destes recursos de memória. A solução passa por copiar o texto com anotação topográfica (em Alçada_Topografica.xml) para um novo ficheiro assegurando que todos os parágrafos ficam marcados com as anotações início e fim.

1.	Criar ficheiro xml para guardar todo o texto;
2.	Para cada alçada:
2.1.	Ler texto sequencialmente dos ficheiros Alçada.txt e Alçada_Topografica.xml;
2.2.	Marcar o início e o fim de cada parágrafo, ignorando as linhas em branco;
2.3.	Marcar a posição da alçada como a soma dos tamanhos das alçadas anteriores; 0 se for a primeira alçada;
2.4.	Marcar as posições e designações de unidades administrativas e paróquias;
2.5.	Marcar as posições e designações das páginas e colunas.
3.	Escrever o novo texto anotado no ficheiro.

Algoritmo 1: Leitura do corpus para a pesquisa

Cada alçada é definida por três ficheiros: Alçada.txt que contém o texto da alçada; Alçada_topografica.xml que contém o texto anotado com a estrutura topológica da alçada; Alçada.xml que contém o texto anotado com a estrutura referencial da alçada.

O texto anotado proveniente da execução do algoritmo anterior permite aumentar a eficiência da pesquisa, uma vez que adopta o princípio das tabelas de dispersão, que como se sabe, são estrutura de dados que permitem efectuar pesquisas de forma bastante eficiente (i.e. com complexidade na classe O(1)).

As pesquisas são feitas recorrendo ao pacote java.util.regex (c.f. [Friedl 2002]), mais concretamente às classes Pattern e Matcher: a primeira representa uma expressão regular e a segunda representa o texto onde se vai pesquisar usando um objecto da classe Pattern. A pesquisa utiliza a técnica divide-and-conquer, pesquisando, separadamente, cada alçada numa thread e, assim, tornando o processo mais eficiente. Notamos que as expressões regulares assumem aqui um duplo papel: localizar padrões para a análise quantitativa e seleccionar a porção do corpus a partir do ficheiro com a anotação topográfica.

Como o texto das Inquirições de 1258 é bastante extenso, criar um objecto Matcher com muito texto é inviável tendo em conta a memória CPU disponível. Então, dividimos o texto por parágrafos e a pesquisa de uma alçada é executada sobre os parágrafos, em vez da alçada completa. Apesar de serem efectuadas mais pesquisas, as mesmas são feitas em textos

pequenos, que acreditamos ser mais eficiente do que efectuar uma pesquisa sobre um texto extenso.

Em Java, o que consome mais memória é a geração de novos objectos e, assim, um ciclo com a geração de objectos deve ser evitável. Na pesquisa de uma alçada é efectuado um ciclo para pesquisar todos os parágrafos. Uma possível abordagem seria criar um objecto Matcher por cada parágrafo, visto que isto é ineficiente, devido ao número de parágrafos que uma alçada pode conter, foi utilizada outra abordagem que consiste na criação de um objecto Matcher e na substituição do texto contido nesse objecto por cada parágrafo da alçada.

Apesar deste melhoramento, a criação de objectos no ciclo é inevitável, pois é necessário guardar informação topográfica (unidade administrativa e paróquia), referencial (página, coluna e número da alçada), a posição onde se inicia e termina a palavra, a posição do parágrafo e a palavra encontrada para cada pesquisa efectuada.

O Algoritmo 2, descrito a seguir, permite que numa pesquisa o que é percorrido não seja o texto do parágrafo mas as posições do parágrafo no texto. A leitura do texto é feita por acesso através da indicação da posição do texto onde começa o último parágrafo e a dimensão (número de caracteres) do parágrafo.

```
1. Para cada parágrafo da alçada:
1.1 para cada palavra encontrada;
1.1.1 guardar a posição da palavra no texto;
1.1.2 guardar a página onde se encontra a palavra;
1.1.3 guardar o número da alçada;
1.1.4 guardar a unidade administrativa onde se encontra a palavra;
1.1.5 guardar a paróquia onde se encontra a palavra;
1.1.6 guardar a posição do parágrafo onde se encontra a palavra;
1.1.7 guardar a palavra encontrada;
1.2 actualizar a visualização dos resultados.
```

Algoritmo 2: Pesquisa por parágrafo

Todos os resultados das pesquisas são guardados, num ficheiro binário, para visualização posterior usando uma thread dedicada de forma a não diminuir o desempenho do programa.

6 Interface gráfica

Nesta secção, vamos apresentar os aspectos mais relevantes da estrutura da interface gráfica do InqExpert, produzida usando o Java Swing [Fischer 2005], intercalados com exemplos ilustrativos de várias pesquisas numa secção de trabalho.

A interface gráfica impõe uma determinada disciplina de trabalho ao investigador na realização de cada uma das pesquisas. Na Figura 3 apresentamos um diagrama de actividades que descreve os passos da actividade de pesquisa de um termo no InqExpert da responsabilidade do utilizador.

Quando a aplicação é executada é iniciada uma nova secção com a exibição de uma fotografia de uma das páginas do documento original das Inquirições de 1258 e, depois, passamos a uma janela onde podemos escolher a parte topográfica do corpus em que desejamos efectuar pesquisas. Podemos por exemplo (vide Figura 4), seleccionar os níveis alçada (1ª), unidade administrativa (Judicato de Prado) e paróquia (todas as paróquias do Judicato de Prado).

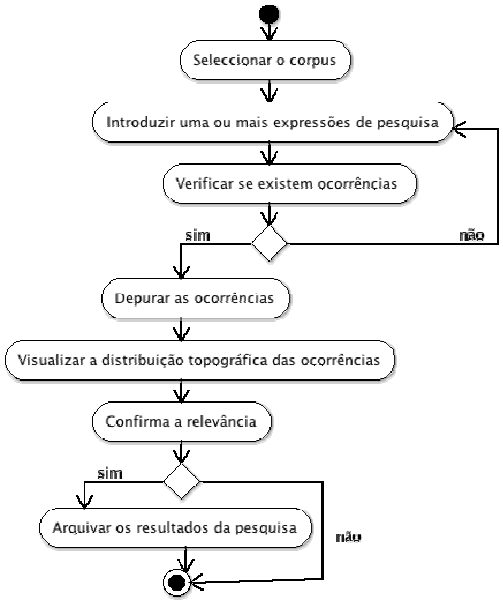

Figura 3.Diagrama da actividade de pesquisa da responsabilidade do utilizador.

Como podemos observar na Figura 5, o separador Pesquisa está dividido em Pesquisa Simples e Pesquisa Composta. Cada uma tem as suas particularidades mas possuem um formato comum composto por três secções, sendo estas a secção de Pesquisa (dados a serem pesquisados), secção de Ocorrências (resultados da pesquisa efectuada), e secção de Texto (mostra o texto seleccionado e os resultados assinalados da pesquisa).

As janelas dos tabuladores Pesquisa Simples e Pesquisa Composta proporcionam as seguintes operações:

- introduzir uma ou mais palavras para pesquisar nas formas prefixa, infixa e sufixa (vide secção Pesquisa no canto superior esquerdo);
- visualizar, na forma de lista vertical de pares (palavra, número de ocorrências), o resultado da pesquisa (vide secção Ocorrências no canto inferior esquerdo) – aqui há botões que permitem movimentos para cima e baixo na lista;
- visualizar, no corpus da secção, cada uma das ocorrências listadas e a sua localização (página, coluna) no corpus (integral).

Notamos que esta última operação permite localizar e visualizar o contexto em que participa cada uma das ocorrências da palavra.

A Pesquisa Simples funciona como qualquer pesquisa normal de texto com inserção da palavra ou palavras a pesquisar. Nas opções de pesquisa temos o prefixo (procura palavras começadas pelos dados inseridos), infixo (palavras que possuem pelo menos um carácter antes e depois dos dados inseridos) e sufixo (palavras que terminam com os dados inseridos).

A Pesquisa Composta procura as palavras seleccionadas baseando-se numa condição lógica. De uma forma simplificada, são pesquisadas palavras por Paróquias em que a condição seja verdadeira. Assim ao inserir palavras temos de indicar se queremos a conjunção (“E”) ou a

disjunção (“OU”). Se for escolhida conjunção, então só serão procuradas Paróquias em que todas as palavras estejam presentes. Se for escolhida a disjunção, então os resultados indicaram as Paróquias em que surgem uma ou mais palavras. A negação apenas indica que a palavra seleccionada não poderá aparecer.

Como podemos observar na Figura 5, foi introduzida a expressão de pesquisa “leira”, como prefixo e visualizada a existência de 16 ocorrências do termo no texto seleccionado, sendo 2 na forma “leiram” e 14 na forma “leiras”.

O botão “+” (“-“) adiciona (retira) termos para a pesquisa e o botão “Pesquisar” despoleta a pesquisa. Os botões abaixo permitem movimentar, para a primeira, última, anterior e seguinte, entre as ocorrências que estão assinaladas (a azul) na janela à direita. O botão “Limpar” apaga a Janela de resultados. A janela texto, contém o número da ocorrência (Ocorrência 10 em 230), o número de página (Página 304) e a coluna (Coluna Esquerda).

Nesta pesquisa as Ocorrências mostram o texto seleccionado permitindo percorrer as várias ocorrências. Aquando de uma ocorrência surge informação extra na parte inferior do Texto.

Como podemos observar na Figura 6, na janela da tabulação Ocorrências é exibida uma lista integral das pesquisas listadas na secção Ocorrências do tabulador Pesquisas. A partir desta tabela é possível efectuar uma primeira selecção das pesquisas através da eliminação (linha-a-linha ou por resultado da pesquisa). As Ocorrências geram dois tabuladores: Ocorrências da Pesquisa Simples e Ocorrências da Pesquisa Composta.

As tabelas podem ser estruturalmente diferentes consoante a origem dos resultados das pesquisas, i.e. provenientes da pesquisa simples ou composta. No caso das ocorrências da Pesquisa Simples (vide Figura 5), surgem na tabela as palavras que foram encontradas na pesquisa. Ao abrirmos, acedemos às informações já descritas na implementação. O duplo clique na linha correspondente a uma palavra específica, abre uma nova janela em que surge o parágrafo no qual a palavra ocorre, estando a palavra marcada. Nas ocorrências da Pesquisa Composta surgem na tabela, não as palavras, mas sim as Unidades Administrativas em que a expressão ocorre. Ao abrirmos, temos acesso às Paróquias nas quais surge a expressão. Aqui o duplo clique originará uma nova janela que mostrará o texto correspondente à Paróquia completa, sendo as palavras assinaladas todas as que não tiveram negação.

Na janela “Termos de pesquisa”, podemos ver os termos que foram escolhidos a partir dos termos pesquisados (neste caso só temos o termo “leira” em duas formas). Na tabela Topográfica/Referencial (abaixo), podemos ver a localização física (página e coluna) e localização topográfica (alçada, unidade administrativa e paróquia) para cada ocorrência do termo pesquisado. Em baixo à esquerda, o botão “Voltar” permite retornar à Pesquisa, “Remover” permite retirar, da tabela, uma ou mais ocorrências seleccionadas e “Seguinte” permite avançar para a tabulação “Estatísticas”. E à direita, o botão “Seleccionar Tudo” para seleccionar todas as ocorrências, “Expandir” permite ver todas as ocorrências (a partir do resumo) e o “Desfazer” (neste caso, inactivo) permite anular a última remoção. Com um duplo “click” numa ocorrência abrimos uma janela com o texto do parágrafo onde ocorre o termo.

As Estatísticas surgem com a necessidade de comparação de dados. Este separador está dividido em ”Selecção de dados de Pesquisa Simples” e “Gráficos”.

Na selecção de dados (Figura 7) temos a oportunidade de seleccionar quais os dados irão criar estatísticas. Partindo de duas tabelas são seleccionados os dados das pesquisas efectuadas e adicionados à tabela de dados para estatísticas. Aqui também poderá ser anulada a última operação tanto de adicionar como de remover. No exemplo, foi eliminada a forma das ocorrências "leiram".

A partir desta selecção podemos aceder ao primeiro nível das estatísticas no botão "seguinte". Nesta, surgem as palavras seleccionadas anteriormente, mostrando agora a sua distribuição por palavra, ou seja, mostra quantas vezes a palavra surge e qual a sua percentagem no total (Figura 8). Aqui poderão também ser seleccionadas apenas algumas palavras para o segundo nível, nível este que permite a visualização por palavra da sua distribuição ao longo do Corpus. Ainda neste nível poderemos seleccionar que campo queremos comparar entre palavras diferentes (também poderá ser a mesma palavra) para uma melhor visualização. No exemplo, podemos visualizar a distribuição topográfica das ocorrências da forma "leiras" no seio das paróquias pertencentes ao Judicato de Prado.

Na janela "Termos de pesquisa", podemos ver os termos que foram escolhidos a partir dos termos pesquisados. Abaixo, podemos ver 2 tabelas; a de cima é a tabela Topográfica/Referencial da tabulação "Ocorrências" e a de baixo é uma tabela resumo (filtro) da tabela de cima, cuja função é guardar a informação seleccionada para as "Estatísticas". Os botões, no lado direito superior, permitem a gestão da tabela, i.e. seleccionar todas as ocorrências, adicionar as ocorrências seleccionadas à tabela "Resumo" e "Desfazer" para anular a execução da última acção. Os botões, no lado direito inferior, permitem a gestão da tabela resumo tal como na tabela de cima (tabela Topográfica/Referencial). Para passar à tabulação "Estatísticas", temos de pressionar o botão "Seguinte", em baixo à direita", e o botão "Voltar" permite retroceder à tabulação "Ocorrências".

A tabulação Estatísticas é dividida em dois tabuladores: Selecção de dados de Pesquisa Simples e Gráficos. No primeiro tabulador, é exibida uma tabela com o resultado integral da pesquisa de palavras e a possibilidade de seleccionar, para a tabela abaixo, algumas destas para a representação gráfica (Barras ou Circular).

Na tabulação "Estatísticas", está uma tabela com 5 colunas: "+/-" (ampliar/reduzir) seleccionar "Palavra", "Valor" (frequências absolutas) e "%" (frequências relativas). Os botões em baixo: "Voltar" à tabulação anterior; "Seguinte" aceder à tabulação posterior; "Expandir" faz a ampliação de todas as ocorrências; "Remover" retira as ocorrências seleccionadas; "Imprimir" imprime a tabela.

Na tabulação "Nível 2", na janela "Termos Seleccionados", a lista em cima à esquerda contém os termos seleccionados na tabulação "Estatísticas Nível 1". A tabela abaixo contém a distribuição topográfica da palavra seleccionada, que está dividida em 3 partes: na parte mais à esquerda, está representada a distribuição (em frequências absolutas e relativas) em cada alçada; na parte ao centro, está representada a distribuição (em frequências absoluta e relativa) em cada unidade administrativa; na parte mais à direita, está representada a distribuição (em frequências absoluta e relativa) em cada paróquia. O botão "Comparar" permite seguir para a tabulação "Comparação estatística".

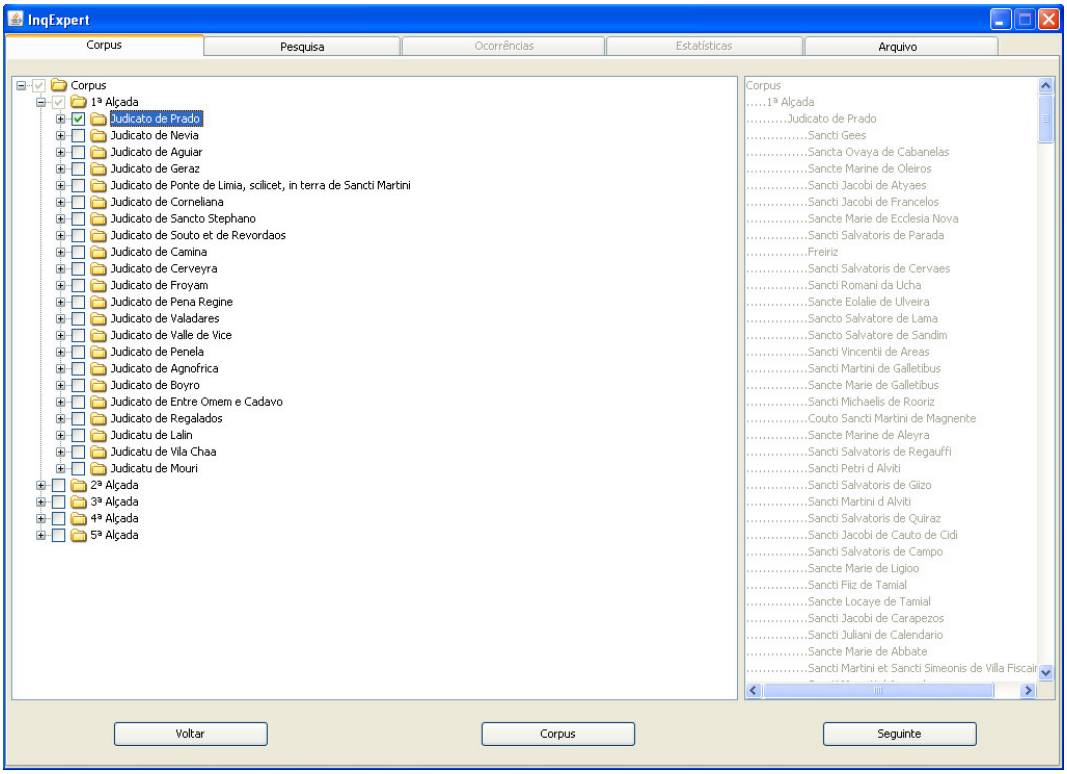

Figura 4: Escolha da parte topográfica

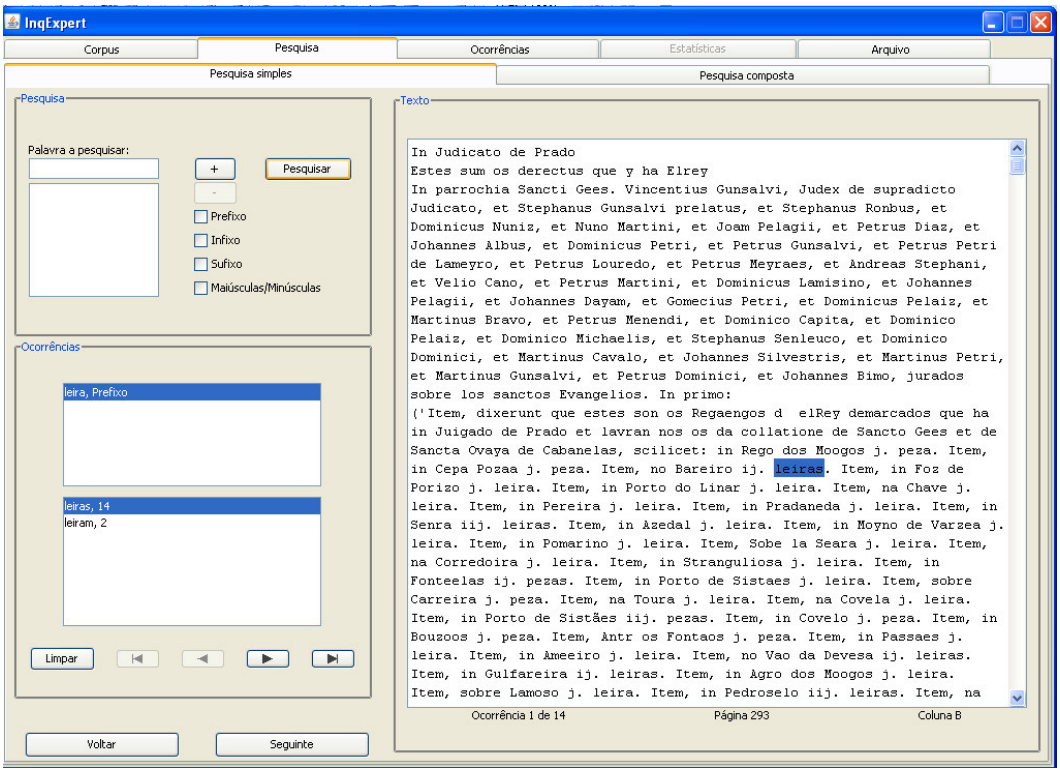

Figura 5: Tabulador Pesquisa Simples

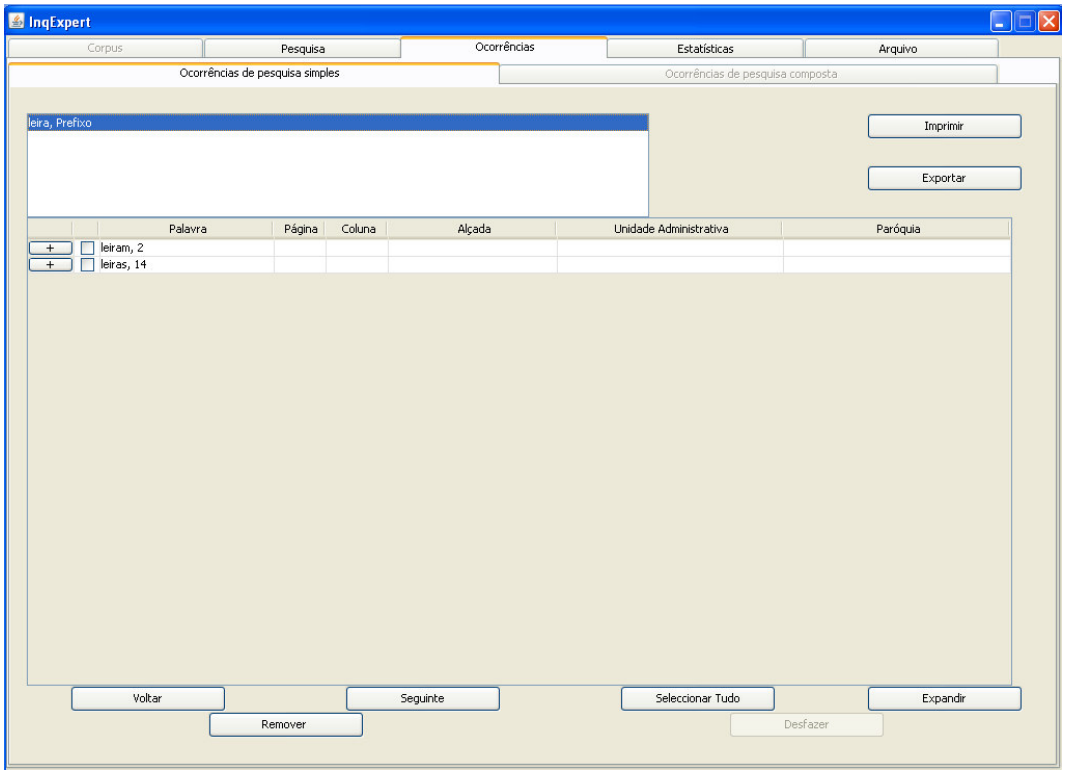

Figura 6: Tabulador Ocorrências

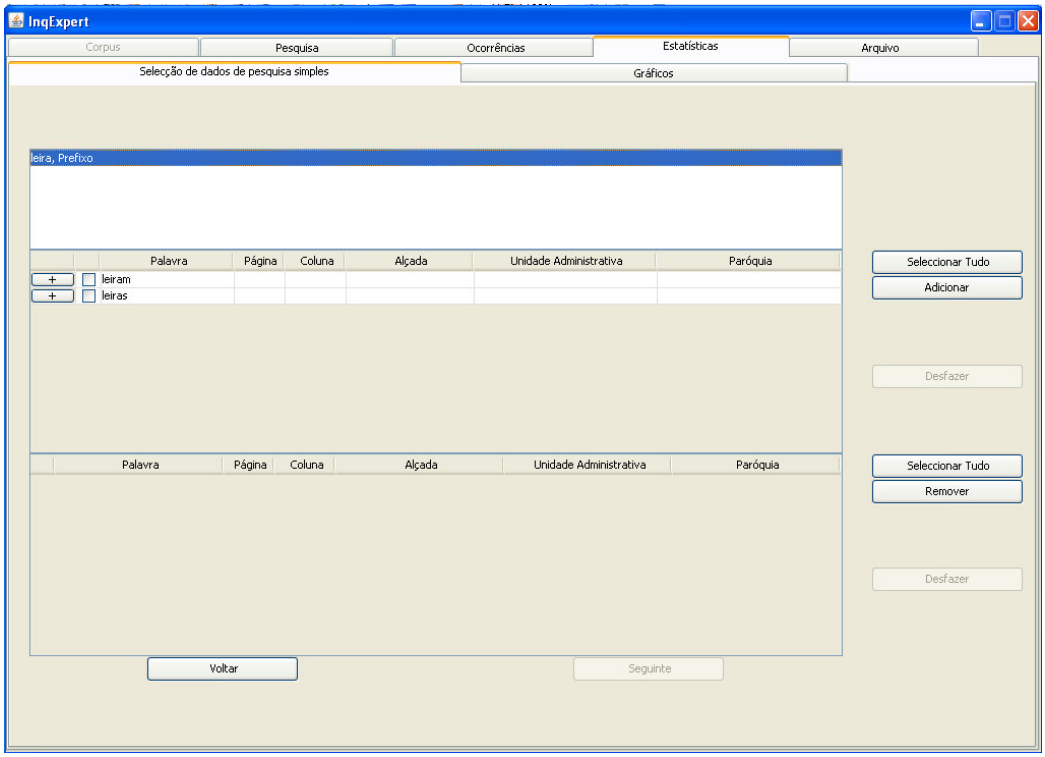

Figura 7: Tabela Topográfica

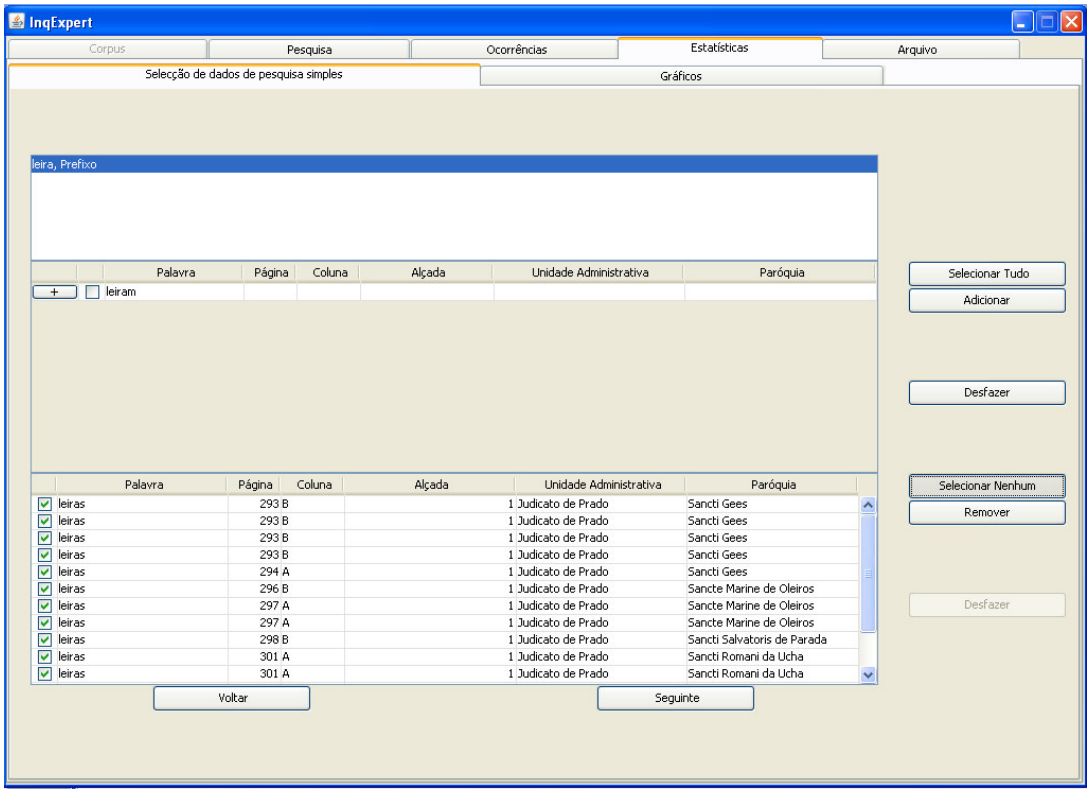

Figura 8: Estatísticas

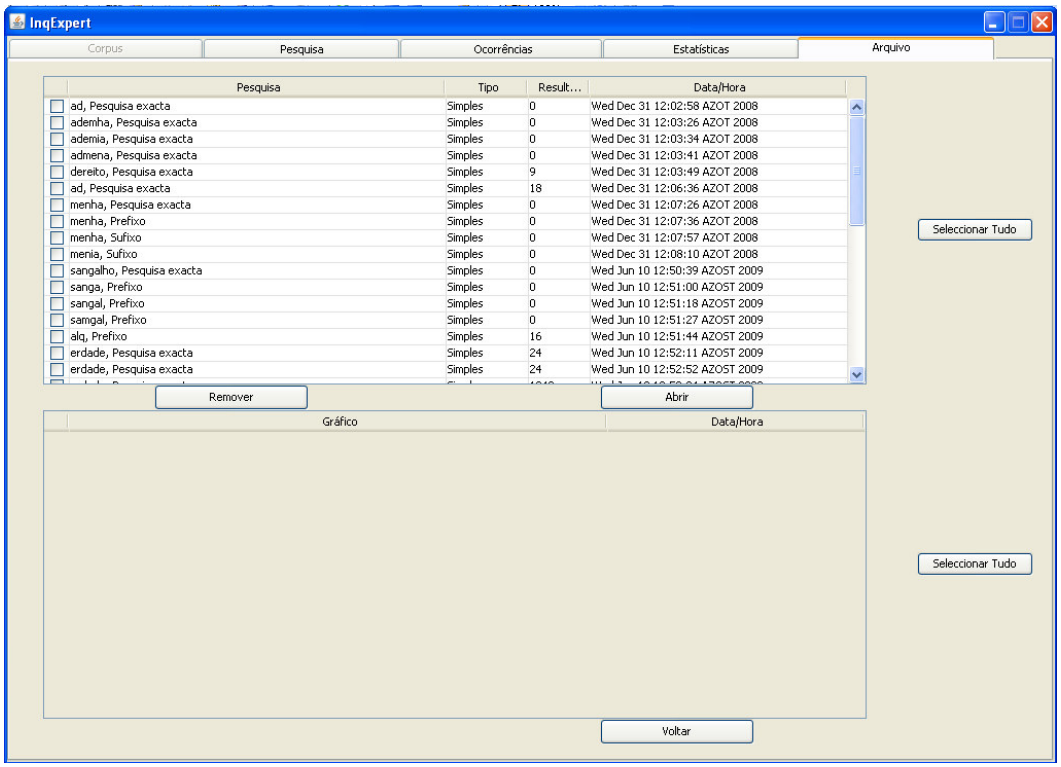

Figura 9: Arquivo

No tabulador Arquivo (vide Figura 9) podemos ver uma tabela de pesquisas com as colunas palavra pesquisada, tipo, número de ocorrências e data/hora. Cada uma destas pesquisas pode ser recuperada para edição e geração gráfica da sua distribuição topográfica.

Na tabulação "Comparação estatística", podemos comparar tantas os termos seleccionados que contêm a distribuição (em frequências absoluta e relativa) por unidade topográfica. O botão "Sair" permite retornar à tabulação "Estatísticas – Nível 2".

7 Conclusões e trabalho futuro

O InqExpert é uma aplicação de software para a análise quantitativa de texto das Inquirições de 1258 que se propõe servir de forma eficiente e fiável a comunidade de historiadores nacionais e estrangeiros, em particular aqueles interessados no estudo da história medieval portuguesa. Salientamos que, sem a utilização de um recurso computacional munido de uma aplicação como o InqExpert, não seria viável o estudo completo deste documento, como se demonstra pelo facto do reconhecimento unânime da riqueza das inquirições contrastar com o pequeno número de estudos que lhe são integralmente dedicados. Devido à grande importância deste documento para se conhecer o Portugal ducentista, perspectivamos, também, a utilização do InqExpert por parte de investigadores e estudantes de outras áreas, tais como linguistas e antropólogos.

O InqExpert permite obter a informação quantitativa base (e.g., tabelas de frequências) para a utilização de ferramentas mais avançadas (usadas no âmbito do Text Mining). Assim, em termos conceptuais e estruturais, o InqExpert está preparado para acomodar outras ferramentas que possam proporcionar novas utilizações (e.g. incorporar as actas das Inquirições de 1220). Outra vertente da evolução do InqExpert, que já está em desenvolvimento com a designação InqWeb, é uma versão Web que, para além de proporcionar todas as funcionalidades disponibilizadas actualmente pelo InqExpert a uma comunidade mais vasta de utilizadores, poderá integrar outras ferramentas de análise quantitativa de texto e, ainda, ferramentas do Text Mining.

Agradecimentos
i) Este projecto é financiado pela Direcção Regional para a Ciência e Tecnologia (DRCT), segundo o disposto no contrato M2.1.2/1/008/2006 (Dezembro de 2006 a Dezembro de 2009).
ii) Agradecemos a Isabel Amaral e ao Henrique Wallenstein pela sua ajuda no processo de digitalização/OCR e na produção do eBook topográfico. Agradecemos, também, a Mário Viana, historiador e coordenador do projecto INQ1258, pelo seu conhecimento das Inquirições de 1258 e, sobretudo, pela sua contribuição na definição dos requisitos do InqExpert.

Bibliografia

Arnold, K. et al. (2005). The Java Programming Language (Fourth Edition). Addison Wesley.

Fischer, P. (2005). An Introduction to Graphical User Interfaces with Java Swing. Addison Wesley.

Fowler, M. (2003). UML Distilled: A Brief Guide to the Standard Object Modeling Language. Addison Wesley, 3rd Edition.

Gate website. http://gate.ac.uk/

Harold, E. R. (2002). Processing XML with Java: A Guide to SAX, DOM, JDOM, JAXP, and TrAX. Addison Wesley.

Konchady, M. (2006). Text Mining Application Programming. Thomson.

Konchady, M. (2008). Building Search Applications: Lucene, LingPipe, and Gate. Mustru Publishing.

LingPipe website. http://alias-i.com/lingpipe/

Lucene website. http://lucene.apache.org/

Marinai, S., Fujisawa, H. (2008). Machine Learning in Document Analysis and Recognition. Studies in Computational Intelligence: Springer.

Ramalho, J. C., Henriques, P. (2002). XML & XSL. FCA.