Aglomeração não Hierárquica em Sistemas Distribuídos de Recuperação de Informação

José Luis Padrão Exposto

Mestrado em Informática

Dissertação submetida à Universidade do Minho para obtenção do grau de Mestre em Informática.

Área de Especialização em Sistemas Distribuídos, Comunicações por Computador e Arquitecturas de Computadores

Dezembro 1997

Abstract

The search for relevant documents in huge collections requires very high computer load and storage overhead.

Although, many research has been made towards the minimization of the document overall space overhead through stoplist techniques and stemming, the storage needed to support so big colections is still very high.

Putting together the decomposition of big colections using clustering algorithms, and their distribution in a high speed network, it would be possible to divide the total document space by each of the network machines, and yet to get concurrent computational processing resources from those same machines.

It is the goal of this thesis to verify the real potencialities of clustering distribution making a comparative study of the performance of an Information Retrieval system changing the number of clusters and confronting a local and distributed mode of that system.

Resumo

A procura de documentos relevantes em colecções de grandes dimensões é um processo que envolve uma carga computacional muito elevada e uma enorme necessidade em termos de capacidade de armazenamento de dados.

Apesar de toda a investigação feita, no sentido de minimizar o espaço físico ocupado pelos documentos, através de técnicas de filtragem, eliminação de palavras comuns e radicalização, são ainda exigidas grandes necessidades de armazenamento devido ao grande número de documentos das colecções.

Se aliarmos as técnicas de aglomeração à distribuição de cada um dos aglomerados, por máquinas ligadas por uma rede de grande velocidade, podemos repartir o espaço ocupado pela totalidade da colecção e tirar ainda partido da utilização concorrente do poder computacional de várias máquinas, quer no processo de classificação, quer no processo de selecção de documentos relevantes a pedidos de utilizadores.

A investigação apresentada nesta tese tem por objectivo verificar as potencialidades reais da distribuição dos aglomerados de documentos e fazer uma estudo comparativo do desempenho de um sistema de Recuperação de Informação variando o número de aglomerados nos modos local e distribuído.

Agradecimentos

Ao meu orientador, Professor Vasco Freitas, pelo tema de investigação e pela oportunidade que me facultou em trabalhar numa área que tanto me motivou.

Ao meu pai, à minha mãe e ao meu irmão que permitiram a minha chegada até aqui e sempre me motivaram para continuar em frente.

Aos colegas de mestrado, especialmente ao Rufino pelas dicas de revisão, ao Albano e à Maria João, agora colegas de profissão, pela ajuda e o bom ambiente de trabalho proporcionado.

À Ana e à Elsa que souberam dar o apoio certo na altura certa.

A todos os meus amigos que conseguiram viver sem mim nos tempos críticos.

Ao Fernando Mina pela revisão do texto.

À Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Bragança, pelas facilidades concedidas durante o período de elaboração desta tese.

O trabalho desenvolvido nesta dissertação foi financiado pela JNICT, no âmbito do programa PRAXIS XXI, Ref. BM/315/94.

Índice

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

1.1 Informação

Informação pode ser definida como o conhecimento que reside no cérebro humano, em qualquer registo escrito ou electrónico ou noutro meio físico. Actualmente, a informação é um alimento indispensável a qualquer elemento de uma sociedade moderna e em constante mudança. A transmissão e armazenamento de informação, para além do cérebro humano, remonta à invenção da própria escrita, entre 3000 e 2000 a.C., altura em que surgiram as primeiras bibliotecas. Todavia, foi com o Renascimento, no Século XV, que a procura e a oferta de informação cresceu consideravelmente, devido não só às tendências intelectuais da época, mas também devido à invenção da tipografia que, revolucionou a difusão do material escrito. No Século XIX, surgiram novos meios de fluxo de informação, tais como publicações periódicas e documentos científicos. Mas é no Século XX que a difusão da informação começa a envolver as grandes massas da população. Iniciam-se as emissões de rádio e televisão e a informação fica disponível de uma forma cada vez mais rápida e envolvente.

O aparecimento dos computadores veio revolucionar completamente a forma de encarar a informação. Neste contexto, emerge a ciência Informática, sendo tomada, comummente, por todos os que a ela estão ligados, como aquela que se encarrega do processamento automático da informação.

Entretanto, as quantidades de informação processada e armazenada foram largamente ultrapassadas com a ajuda da evolução tecnológica na área da microelectrónica. A variedade de ramos que emergiram da informática leva à necessidade da criação de uma ciência que chame a si as tarefas de colectar, organizar, armazenar, recuperar e disseminar o conhecimento. Surgiu então a ciência da informação.

A crescente necessidade bibliográfica, acompanhada pelo avanço tecnológico do Século XX, levou à informatização da quase totalidade das bases de dados, catálogos

e colecções. É neste sentido que a ciência da informação começava a dar os seus frutos.

Mas se a utilização do computador veio, por um lado, trazer enormes vantagens na concretização dos objectivos da ciência da informação, veio por outro permitir o aumento desmesurado dos repositórios de informação e alguma perda de controlo sobre a sua localização e conteúdo.

1.2 A difusão da informação

A indústria dos computadores, apesar da sua juventude, rapidamente se inseriu noutras áreas, tais como meios de transporte, comunicações, serviços, indústrias de fabrico, etc. O crescimento da aquisição, processamento e distribuição da informação aumenta com uma sociedade cada vez mais empresarial e competitiva, necessitando-se de um tratamento mais sofisticado da mesma.

O aparecimento das comunicações por computador trouxe consigo uma nova visão do mundo. O aumento rápido do poder computacional dos processadores já prometia a sua utilização por várias pessoas. A interligação dos computadores foi o passo essencial para que tal se concretizasse.

Adicionalmente, o aumento da largura de banda proporcionou o acesso à informação remota em tempos comparáveis ao acesso a um disco local. A rede de computadores tranformou-se numa extensão dos periféricos locais de um computador. O computador potente e isolado que satisfaz as necessidades computacionais dos seus utilizadores, dá agora passo a um número mais elevado de computadores distantes, no entanto, interconectados. Torna-se, assim, possível a partilha de recursos e de dados, e a distribuição da carga computacional por vários computadores.

A conectividade dos computadores veio efectivamente beneficiar todas as áreas que apostaram na sua indústria. A partilha de informação e o seu intercâmbio entre secções diferentes de uma empresa tornou-se viável; por exemplo, as distâncias entre filiais deixam de ser um factor significativo. Por outro lado, veio assegurar a continuidade da informação através da sua replicação, permitindo o funcionamento contínuo de tarefas de alto risco.

Mas se o mundo empresarial saiu beneficiado, mais beneficiados, ainda, sairam os investigadores e cientistas, que vêm assim uma forma rápida e prática de difundir as suas ideias e partilhar informação.

1.3 A Biblioteca Universal

O contínuo aumento da largura de banda das redes de computadores e a progressiva evolução das tecnologias multimédia vieram dar azo a Tim Berners-Lee para a

criação da *World Wide Web* (WWW) [BLCL+94]. A partilha da informação feita até então através dos protocolos *telnet*, *ftp*, *e-mail*, *gopher* e *wais*, passa a ser feita por um sistema de visualização de páginas que dão acesso a outras páginas remotas ou locais através de hiperligações, em que é possível a conjugação de todos os meios multimédia disponíveis (imagem, som, vídeo). Esta integração foi possível graças à implementação de um novo protocolo ao nível da aplicação designado por *HyperText Transfer Protocol* (HTTP) [FGM+97] e da criação de uma linguagem de etiquetas orientada para a multimédia e para as hiperligações, designada por *HyperText Markup Language* (HTML) [BLC95].

A WWW veio permitir a disponibilização de uma série de recursos aos quais todos os utilizadores da Internet têm acesso. O sistema WWW veio afectar a Internet de tal maneira, que o crescimento de computadores chega aos 9% mensais. A receita foi simples: à visualização da informação de forma gráfica, possibilitando observar simutaneamente texto, imagens, som e vídeo, foi aliada a interacção através de hiperligações incorporadas no texto. Desta forma, o utilizador pode "movimentar-se" facilmente de recurso em recurso. Tendo em conta que todos os recursos podem ter uma ligação remota, obtemos uma teia [1] complexa de informação, na qual, um utilizador uma vez emaranhado no meio dela, sente sérias dificuldades em se orientar.

O crescimento de informação na WWW é deveras prodigioso. Repare-se na evolução do número de máquinas na internet até Julho de 1997 no gráfico da figura 1.1 [Wiz97]. Se estimarmos o número de utilizadores que a elas estão ligados, imagine-se a quantidade de informação que poderão publicar!

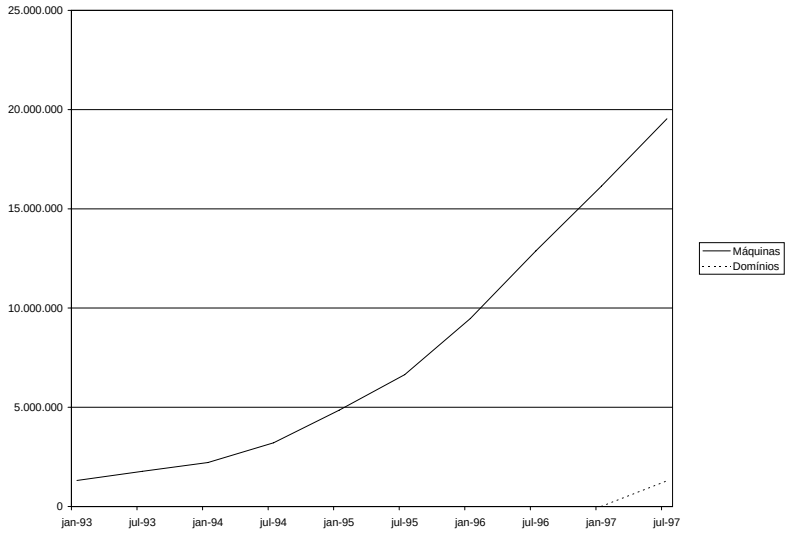

Figura 1.1: Evolução do número de máquinas e domínios na internet.

A facilidade de publicação e a liberdade de expressão existentes na Internet

[1] Do inglês *web*.

atraiu também um grande número de empresas que aproveitam o potencial enorme número de clientes para divulgar e vender os seus produtos. Todo este cenário permitiu um aumento tão desmesurado da quantidade de informação, que a sua procura se manifesta uma tarefa difícil.

Resumindo, a WWW é, por excelência, a biblioteca universal. O seu crescimento surpreendeu tudo e todos. As suas dimensões e o seu conteúdo ultrapassam o conhecimento de qualquer ser humano e, por isso, assume um papel muito valioso em termos de disseminação do conhecimento. Mas todo este poder de fogo, de nada serve se a informação pretendida por um utilizador não for encontrada facilmente e com êxito. Daí a constante preocupação com a pesquisa na WWW.

1.4 Descoberta de recursos na WWW

Nesta secção são mencionados alguns dos serviços de descoberta de recursos de informação na WWW. São categoricamente divididos em sistemas integrados e não integrados, para marcar a distinção entre aqueles que pretendem abordar o problema directamente com as infra-estruturas relacionadas com o funcionamento protocolar da WWW e os outros que assumem os recursos como documentos, passando por cima de pormenores intrínsecos à WWW e que constituem a solução colocada em prática actualmente. Esta última visão é colocada a um nível superior, dependente da camada protocolar, e qualquer modificação que seja efectuada no funcionamento interno da WWW, pode criar sérios problemas, que poderão levar à reimplementação de raiz deste tipo de sistemas. Esta divisão apesar de um pouco rude, tem como objectivo facilitar a análise do estado da arte nesta matéria, de forma a distinguir aquilo que está disponível actualmente e o que seria ideal colocar em funcionamento.

Seria de todo fundamental que fosse colocada "ordem na WWW" ao nível do protocolo, isto é, recorrendo a mecanismos directamente integrados dentro do próprio sistema de devolução de recursos [WD94]. O sistema de resolução de nomes universais vem precisamente propôr esta solução [Sol97]. Por ser uma proposta radical, a sua aceitação deve ser feita de forma atómica e cuidadosa, isto é, duma vez só.

Enquanto não são tomadas decisões relativas ao funcionamento do fulcro da WWW, novas soluções de pesquisas têm vindo a ser desenvolvidas como forma de dar um novo fôlego à descoberta de recursos. Os chamados "motores de pesquisa" têm feito grande furor dentro da comunidade da WWW, devido à ânsia em encontrar uma solução que vise a procura de recursos em quantidade e qualidade suficientes.

1.4.1 Serviços integrados

A pesquisa e organização da informação na WWW foi uma questão desde sempre tomada como um problema pertinente e, por isso, pensada desde que ela foi criada.

Para a identificação universal de cada rescurso que se encontra na WWW é utilizada um Localizador Universal de Recursos[2](URL)[BLMM94]. No entanto, estes identificadores não são tolerantes a faltas, torna-se difícil a sua manutenção, para além de não disporem de qualquer informação de qualidade de serviço. Como alternativa foi pensado o Nome Universal de Recursos[3](URN)[SM94] que permite uma identificação persistente, única, independente da localização e com possibilidade de referência a múltiplos recursos permitindo ainda vários critérios na decisão do melhor recurso disponível em determinada altura. A conversão de um URN para URL pressupõe um serviço de resolução semelhante ao Serviço de Directoria DNS. Para que sejam feitas decisões para a escolha do URN existe ainda um outro Identificador Universal de Características[4](URC)[RDM95], que contem a informação particular a cada URL. Este identificador para além de conter a meta-informação (informação sobre a informação) das localizações do recurso, pode ainda conter informações sobre o tipo de recurso, versão, datas de criação e modificação, distância na rede e inclusivamente dados bibliográficos e autenticação.

Existem actualmente vários sistemas que implementam o serviço de resolução de URNs, não tendo ainda nenhum sido adoptado na prática. Destaca-se o WHOIS++ [DFM95] pela sua qualidade como serviço de directoria genérico, o Simple Discovery Protocol (SDP) [HK95] que tira partido do Multicast IP para resolução de URNs e o Resolver Discovery Service (RDS) [Slo97] que tem sido abordado recentemente.

O princípio de funcionamento destes sistemas consiste na delegação da informação entre servidores inseridos dentro de uma rede de servidores. A técnica de propagação e procura de informação é baseada num serviço de directoria com maior generalização que o DNS.

Facilmente se depreende que a criação de sistemas deste tipo, prometem seriamente a resolução das lacunas referidas para a existência de apenas URLs, podendo também contribuir para a pesquisa de informação mais pormenorizada, tais como endereços de correio electrónico, documentos publicados electronicamente e até mesmo páginas da WWW. Torna-se ainda possível a filtragem de informação, o controlo de acesso de acordo com regras estipuladas, a manutenção de privaciade e a autenticação dos recursos, tema tão em voga actualmente.

O serviço de URCs é um tema que tem levado algum tempo a ser concretizado uma vez que, devido ao seu grau de integração, a sua estabilização envolve a definição de uma série de normas e generalizações e de modificações que afectam o próprio funcionamento da WWW.

[2]Do inglês *Universal Resource Locator*.
[3]Do inglês *Universal Resource Name*.
[4]Do inglês *Universal Resource Characteristic*.

1.4.2 Serviços não integrados

Dentro dos serviços não integrados incluem-se os catálogos e os motores de pesquisa. Nos primeiros são feitas divisões estanques sobre assuntos de interesse geral. Estes serviços envolvem uma forte componente humana, constituindo um forma bastante eficiente de procurar aquilo que se pretende, carecendo no entanto de totalidade, ou seja, é bem possível encontrar algo que se pretende, mas longe de encontrar tudo o pretendido.

Os motores de pesquisa baseiam-se no princípio da busca exaustiva da informação através de *robots* que a colectam e que será indexada em bases de dados locais. A grande dificuldade destes serviços é a gestão das gigantescas bases de dados geradas, apesar da eliminação de grande parte da informação através das técnicas que veremos mais à frente. Actualmente os motores de pesquisa constituem o ponto de partida principal para a pesquisa na `WWW`.

1.5 Objectivos da tese

A descoberta de recursos de informação é um tema que tem que ser tomado com bastante preocupação dado o crescimento verificado na `WWW`. Talvez por isso, os investigadores tenham sido apanhados de surpresa ao não preverem a situação caricaticamente caótica existente actualmente.

Os serviços integrados constituem, sem dúvida, a solução por excelência, já que, conseguem abordar o problema da procura de recursos, e incluir outro tipo de meta-informação. Por outro lado, é sabido que a `WWW` sofrerá alterações, se bem que não se sabe até que ponto. Fazendo uma abordagem de fundo, é possível resolver as questões actuais e evitar que outros problemas surjam no futuro.

Os motores de pesquisa baseiam-se num conceito muito simples: a concentração, ou seja, percorrer a `WWW` e concentrar a informação. Actualmente as empresas de desenvolvimento de motores de pesquisa na `WWW` estão a concentrar os seus esforços no sentido de colocar em prática a melhor utilização das técnicas de Recuperação de Informação [5] (RI) [Cro95].

Em ambos os tipos de serviços aqui apresentados, a necessidade em adquirir informação é uma característica comum. Informação essa que precisa de ser indexada de alguma forma, de tal modo que permita a pesquisa em tempos aceitáveis. Este é um problema que já tem sido abordado muito antes da existência da `WWW` pela ciência de Recuperação de Informação

O presente trabalho tem como orientação a descoberta de recursos de informação na `WWW`, partindo dos conceitos associados à RI e tendo em vista um possível aproveitamento das conclusões que sejam retiradas deste trabalho, por qualquer um dos

[5] Do inglês *Information Retrieval*

serviços de descoberta de recursos referidos, ou seja o aproveitamento de técnicas de indexação e a disseminação de informação numa rede de servidores que constituem o sistema. Para isso, foram tomadas como ideias base os aspectos fundamentais que cada um dos serviços aproveita melhor.

Pretende-se, então, construir e verificar o desempenho de um sistema que alie o poder de aquisição e concentração de informação de um motor de pesquisa, a uma divisão de documentos, comparável à *classificação* realizada pelos catálogos, tirando todo o partido de uma base de dados distribuída.

Note-se que os resultados deste trabalho poderão servir de base tanto para investigações ao nível dos serviços integrados, uma vez que é focado o aspecto da distribuição de índices, e para os serviços não integrados pois a classificação automática e a distribuição das bases de índices constituem uma inovação a este nível, sendo esta uma das maiores lacunas destes sistemas.

O sistema será construído como um banco de teste, a fim de poderem ser tiradas as conclusões necessárias para a possibilidade de viabilidade em ambiente laboratorial. Naturalmente, alguns pormenores serão desprezados, dada a extensão de matérias abrangidas pelo estudo. É dada particular importância aos resultados experimentais, pois só através destes será possível extrair algum tipo de conclusões.

Será o objetivo final confrontar a diferença de tempos de desempenho, quer em cálculos quer em transporte pela rede, e o desempenho das técnicas utilizadas entre um sistema centralizado e distribuído a fim de verificar a possibilidade de sucesso da distribuição das bases de dados.

1.6 Resumo dos capítulos seguintes

O texto que se segue, toma uma sequência lógica desde a introdução aos sistemas de RI até à apresentação dos resultados e respectivas conclusões.

Assim, no capítulo 2 é apresentado o pano de fundo que rodeia a RI, definindo conceitos actualizados e apresentando o esquema geral de recuperação de informação, descrevendo os passos necessários para uma aquisição de informação e sua devolução efectiva.

No capítulo 3 são descritos os modelos conceptuais disponíveis na RI e as suas características, dando particular relevância ao modelo do espaço vectorial e ao modelo de aglomeração. É ainda, referido o processo de avaliação de sistemas de RI e a forma como devem ser apresentados os resultados para uma melhor comparação entre as técnicas.

O capítulo 4 é dedicado à descrição da plataforma implementada para a concretização dos objectivos que este trabalho se propõe. É feito um resumo de alguns dos sistemas disponíveis actualmente. São desenvolvidos os aspectos que levaram à

selecção das técnicas de RI nas suas diversas fases e, ainda, os pormenores de implementação que tiveram que ser tomados em conta para a optimização do sistema.

O capítulo seguinte é dedicado à descrição e análise de todos os testes efectuados com a colecção *Cranfield*. São apresentados resultados da eficiência e eficácia do sistema.

Por último são retiradas as conclusões do presente trabalho com base nos resultados obtidos e apresentadas sugestões para trabalho futuro.

Capítulo 2

Recuperação de Informação

A Recuperação de Informação (RI) é a área que tem como finalidade a aquisição, armazenamento e devolução de informação perante uma especificação dada por um utilizador.

Apesar de não ser explícito no nome, as técnicas inerentes à RI têm carácter automático, colocando de parte, desde logo, qualquer interacção humana no processo de devolução de informação. Não confundir, no entanto, uma das técnicas utilizadas na RI, chamada realimentação de relevância[1] que tira partido de informações dadas pelo utilizador depois de já ter sido disponibilizada a informacão pretendida. O tipo de recursos envolvidos podem ir desde texto, imagens, som ou video [Man97]. No entanto, no que respeita ao tema da tese, apenas será abordado o primeiro tipo.

O aspecto mais importante da RI é que esta não informa o utilizador acerca do assunto a que o seu pedido se refere, mas sim acerca da existência ou não desse assunto e a sua localização em algum recurso que o sistema refira [vR79].

Frequentemente a RI é confundida com Recuperação de Dados (RD). Vejamos as principais diferenças esquematizadas na Tabela 2.1 [FBY92, vR79].

Ao contrário do que sucede num sistema de Recuperação de Dados (RD), os sistemas de RI executam operações de comparação entre os objectos de dados de forma parcial, ou seja, a comparação é feita com base na optimização de uma função de dois argumentos, que são substituídos pelos objectos de dados. A métrica de comparação é baseada em dados estatísticos, como seja a frequência de palavras em cada objecto de dados. Como resultado obtemos então a devolução de um conjunto de itens considerados relevantes, enquanto que no sistema de RD, são devolvidos os itens resultantes de uma comparação precisa, sendo por isso sensível a falhas.

No que respeita às interrogações, nos sistemas de RI, para além de não necessitarem de uma especificação completa, podem também ser descritas através de linguagem natural.

[1] Do inglês *relevance feedback*.

	Recuperação de Dados	Recuperação de Informação
Objecto de Dados	Tabela	Documento
Comparação	Exacta	Parcial, Melhor
Inferência	Dedução	Indução
Modelo	Determinístico	Probabilístico
Linguagem da interrogação	Artificial	Natural
Especificidade da interrogação	Completa	Incompleta
Itens Desejados	Por comparação	Por relevância
Resposta a Erros	Sensível	Insensível

Tabela 2.1: Comparação entre a Recuperação de Informação e a Recuperação de Dados

2.1 Definições

Para se compreender melhor o funcionamento e a constituição estrutural de um sistema de RI, vamos assentar algumas definições importantes que serão usadas ao longo deste texto.

Um *documento* é o elemento fundamental de um sistema de RI, é este que o utilizador pretende adquirir. A descrição da informação que o utilizador deseja é chamada a *interrogação*. Os documentos que contêm informação relacionada com uma interrogação são designados *documentos relevantes*. Um conjunto elevado de documentos constitui uma *colecção*. Como apenas serão abordados documentos textuais, estes podem ser ficheiros de texto, recursos da `WWW` ou apenas um conjunto relativamente pequeno de parágrafos, como é o caso de algumas colecções de teste, que são utilizadas na avaliação do desempenho de um sistema de RI. Num sistema de RI, os documentos assumem duas formas: o documento original que constitui a colecção, e que é o objectivo do utilizador; e a representação interna que é feita do documento, que naturalmente é uma simplificação do primeiro. Enquanto não surgir o risco de confusão chamar-lhe-emos documento a ambas as formas. Cada documento é constituído por um conjunto de *termos*, ou seja, por palavras, as quais, para se obter a representação do documento, são sujeitas a um tratamento especial, como veremos na secção 2.5.

A construção de um sistema de RI é sempre feito tendo em vista a optimização dos algoritmos que manipulam as estruturas de dados que conterão a informação necessária para o seu funcionamento. A optimização é feita sob dois pontos de vista: *eficiência*[2] e *eficácia*[3]. Enquanto que a primeira mede o modo como são aproveitados os recursos computacionais, como o tempo de processamento, memória necessária

[2]Do inglês *efficiency*.
[3]Do inglês *effectiveness*.

e espaço em disco para armazenar as estruturas, a segunda mede a forma como o sistema responde positivamente àquilo que se pretende. Neste caso, a devolução do maior número de documentos relevantes, evitando a devolução de documentos não-relevantes. A eficácia de um sistema é medida em termos de *precisão*[4] e *totalidade*[5]. A precisão é dada pelo número de documentos devolvidos considerados relevantes, sobre o número total de documentos devolvidos. A totalidade é dada pelo número de documentos devolvidos considerados relevantes sobre o número de documentos realmente relevantes (devolvidos ou não). Estes aspectos serão vistos mais em pormenor na secção 3.6

2.2 Estatística na RI

Para nós humanos, a leitura de um texto permite-nos identificar facilmente os aspectos mais importantes do seu conteúdo. Para um computador o processo já não se torna tão simples. Se bem que se tem feito investigação na área da Inteligência Artificial no sentido de analisar e compreender textos através do processamento de linguagem natural (PLN) [RL94], as técnicas utilizadas possuem um consumo computacional muito elevado, pelo que a sua utilização é bastante restrita e não vulgarizada em casos práticos reais. Resta então pegar na solução mais viável, transparente e inerte - a análise da palavra. Como não é feita qualquer conotação semântica, uma vez que é utilizada a palavra só por si, podemos ter um grau de generalidade muito maior, possibilitando a sua aplicação a variações de conteúdo e mesmo criar alguma independência em relação às variações linguísticas.

A palavra pode ser considerada o elemento básico de um sistema de RI, uma vez que é através dela que o sistema realiza as operações básicas de comparação, tanto entre documentos entre si, como entre documentos e interrogações. Quanto mais palavras houver em comum maior relacionamento existe entre os objectos formados pelas palavras. Se adicionalmente for contabilizada a frequência com que essas palavras ocorrem no documento, facilmente somos conduzidos à formulação de um modelo estatístico.

Esta característica permite desenvolver, nos sistemas de RI, o seu maior potencial - a quantificação dos objectos -, permitindo a utilização de métricas quantitativas e respostas distribuídas segundo um grau de relevância. Isto significa que, perante uma interrogação, os documentos são ordenados por similaridade, podendo assim criar-se dois grandes grupos de documentos: relevantes e não relevantes. Os primeiros serão aqueles que servirão como resposta ideal, enquanto que os segundos deverão ficar excluídos, restringindo assim, o número de soluções para o utilizador.

[4]Do inglês *precision*.
[5]Do inglês *recall*.

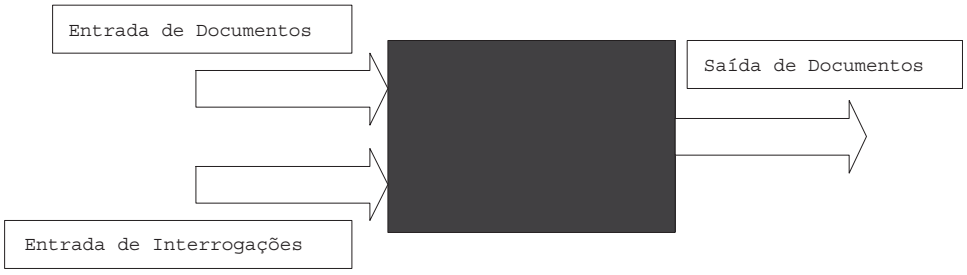

Figura 2.1: Modelo de caixa preta para um sistema de RI

2.3 Estrutura de um sistema de RI

Um sistema de RI pode ser visto, grosso modo, através de um modelo de caixa preta, com duas entradas e uma saída. Uma das entradas é o canal por onde é armazenada toda a informação relativa à colecção de documentos. Estes podem permanecer de forma estática, sem actualização, ou então de forma dinâmica, sendo assim possível a sua renovação ou incrementação gradual. Esta entrada não está directamente relacionada com o utilizador do sistema, cabendo apenas ao gestor a decisão do tipo e número de documentos que serão lá introduzidos.

Os restantes canais são aqueles que têm actividade directa com o utilizador. Pela outra entrada é encaminhada a interrogação, à qual o sistema responderá pelo canal de saída com os documentos relevantes ao seu pedido.

Observando agora o contéudo da caixa (figura 2.1), a cada canal é correspondido com um módulo de processamento. Para sustentar este conjunto de processos, o sistema dispõe ainda de uma base de dados central que contém as representações dos documentos.

Para conseguir retirar a informação mais significativa do texto dos documentos originais e colocar o resultado em estruturas de dados suficientemente eficientes, é necessário proceder a um processo de transformação dos documentos nos seus representantes. Este processo pode ser decomposto em duas fases: o processamento de texto e a indexação automática. A primeira fase encarrega-se de decompôr o texto em palavras, evitando as repetições e caracteres sem significado, como por exemplo pontuação ou caracteres de controlo. A segunda tem por objectivo integrar os termos dos documentos da forma mais optimizada possível na estrutura de dados do sistema de RI. As palavras são confrontadas com uma lista de paragem, ou seja, lista de palavras comuns, e eliminadas do documento. Posteriormente, são sujeitas a um processo de radicalização, resultando na unificação de palavras que tenham ficado com o mesmo radical. Este processo pode ser visto mais detalhadamente nas secções 2.4 e 2.5.

O sistema, para ser activado, necessita de receber um pedido, que é analisado e interpretado consoante o tipo de linguagem de interrogação que seja utilizado. A

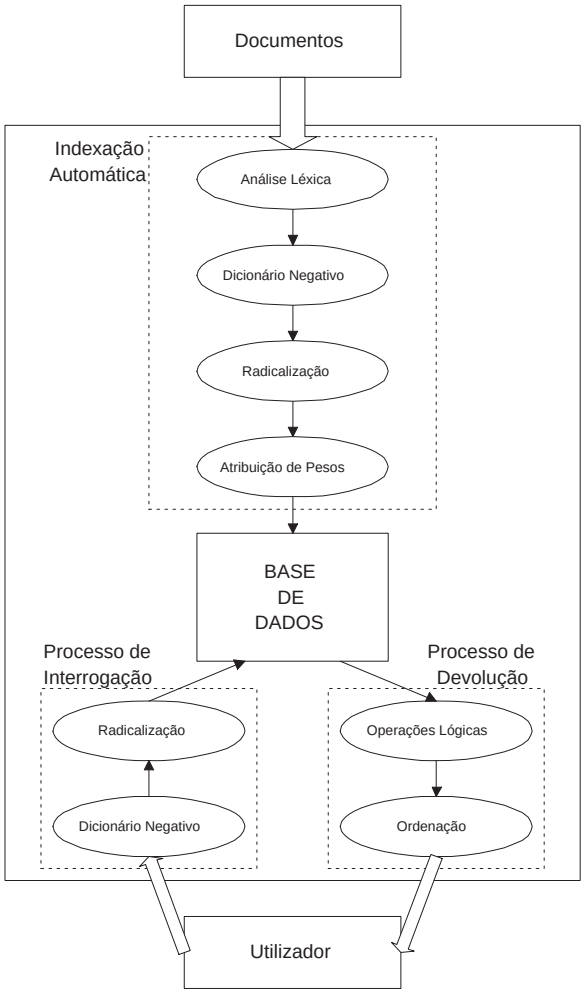

Figura 2.2: Esquema funcional de um Sistema de RI

interrogação é também sujeita ao mesmo processo de eliminação de palavras comuns e de radicalização, para garantir uniformização dos termos em ambos (documentos e interrogações).

Após ter sido calculado o conjunto solução, o terceiro processo encarrega-se de devolvê-la ao utilizador, ordenado por ordem de relevância.

O cálculo do conjunto solução constitui o cerne de um sistema de RI, e é dependente do tipo de abordagem que é feita em cada sistema em particular. Cada tipo de abordagem oferece uma variação tanto na solução como no desempenho geral. O estudo de alguns modelos funcionais é feito no capítulo 3.

2.4 Estruturas de dados

As estruturas de dados de qualquer programa de computador são o suporte fundamental para a obtenção de algoritmos poderosos e eficientes. No caso dos sistemas de RI, a parte das estruturas de dados tem obtido particular interesse devido à dimensão extremamente enorme que as colecções de documentos podem atingir.

Os avanços tecnológicos do *hardware*, nomeadamente na área de armazenamento de dados, têm contruibuido para o aperfeiçoamento e aumento da capacidade da memória central.

Devido ao tamanho das colecções e à sua característica persistente, é necessário recorrer à utilização de suportes secundários de armazenamento.

A utilização de discos duros é sem dúvida a mais diversificada uma vez que permite uma velocidade de acesso bastante elevada. O discos ópticos são uma tecnologia que tem vindo a ser utilizada gradualmente. As vantagens vão desde a sua imensa capacidade de armazenamento e durabilidade dos dados até a uma maior portabilidade.

2.4.1 Ficheiro Invertido

Os ficheiros invertidos são dos tipos de estruturas de ficheiros mais utilizadas nos sistemas de RI , não só pela sua simplicidade na implementação, mas principalmente pelo seu desempenho.

O armazenamento da informação nos ficheiros invertidos, contraria a organização lógica do documento em si. A figura 2.3 apresenta uma representação de um ficheiro invertido. Em vez de se guardarem os documentos sequencialmente juntamente com o conjunto de termos respectivos, solução que tornar-se-ia demasiado ineficiente, guardam-se os termos individualmente e uma referência para os documentos que contêm esse termo, conjuntamente com outra informação adicional, constituída pela frequência desse termo no documento.

Para manter esta estrutura, por razões óbvias de dimensão e persistência, ela é armazenada em disco através de um array ordenado, árvores-B ou árvores PAT [FBY92]. As árvores-B, apesar de necessitarem de mais espaço, cerca de 10% a 100% do tamanho do texto original, possuem uma maior facilidade de actualização [CP92]. A ordem de complexidade para o tempo de pesquisa é $\mathcal{O}(\log n)$. As árvores PAT têm uma performance idêntica mas abrangem um raio mais amplo de aplicações, como é o caso de pesquisas em frases, pesquisas com expressões regulares e pesquisa aproximada com cadeias de caracteres.

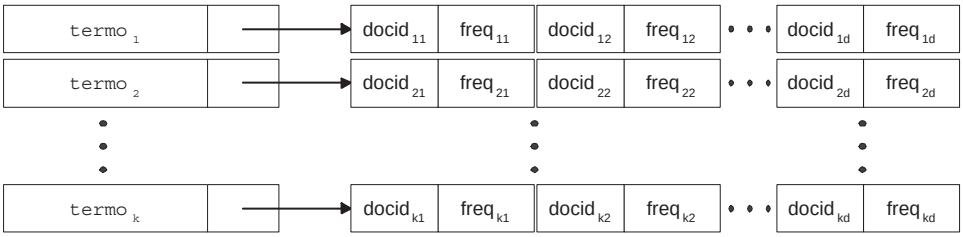

Figura 2.3: Estrutura de um ficheiro invertido

2.5 Indexação Automática

A indexação automática é um processo em que se realiza a transformação do texto original em termos. O principal objectivo da indexação é extrair do texto a sua essência, ou seja, as palavras que contribuem mais para a sua descrição. Por outro lado, pretende-se também reduzir ao máximo o seu conteúdo por forma a diminuir o espaço necessário para o seu armazenamento, tentando eliminar palavras que em nada contribuem para a sua interpretação.

O texto é primeiramente decomposto em palavras, às quais são retiradas aquelas consideradas sem significado semântico, após o que são reduzidas ao seu radical mínimo através de regras previamente definidas. Por último, cada termo é afectado de um valor que reflicta a importância desse termo no documento. Em muitos sistemas são ainda utilizados dicionários de sinónimos com o objectivo de reduzir ainda mais a quantidade de termos resultantes. No entanto a sua construção automática torna-se bastante complexa, não sendo ainda muito utilizado nos sistemas de RI [JC93]. O resultado é um conjunto mínimo de pares termos / frequência, que é calculado durante o processo, sendo armazenados na estrutura de dados adequada.

A definição de dicionários negativos e a criação das regras de radicalização são passos dependentes da língua que o texto original se encontra.

2.5.1 Análise Léxica

A preparação do texto original, para que seja possível a sua transformação em termos indexados, passa por uma primeira fase de separação atómica da unidade textual - a palavra. Um texto é uma sequência de caracteres que, para além de constituirem palavras, podem servir como separadores ou controlo. Dentro dos caracteres separadores e controlo podemos incluir a pontuação que é utilizada na escrita convencional, os espaços, quebras de linha e marcas que permitem a formatação do texto. Todos estes elementos têm uma função específica e necessária no respectivo contexto. Na análise léxica ajudam na separação das palavras, acabando por ser filtrados, uma vez que não contribuem para a compreensão semântica do texto.

A análise léxica é um passo muito sensível na indexação automática, pois surgem

por vezes situações ambíguas, que podem levar à interpretação errada do texto. É por exemplo, a distinção entre uma maiúscula forçada por um início de frase e um nome próprio, ou então a distinção entre um ponto final e um ponto de um número real ou de uma abreviatura. Muitos destes problemas podem ser minimizados recorrendo a aplicações como o *lex* ou *flex*, que permitem a filtragem sucessiva através da utilização de expressões regulares [GT94].

2.5.2 Dicionário negativo

O dicionário negativo[6] é um conjunto de palavras, que são, normalmente, enumeradas manualmente, ou criadas automáticamente através da análise de frequência de palavras em colecções. As palavras com maior frequência são supostamente as que farão parte do dicionário negativo. Não sendo, no entanto, garantida uma margem de erro de 0%, uma vez que podem ser incluídas palavras com utilidade contextual.

A ideia básica dos dicionários negativos é eliminar palavras sem contexto semântico, tais como determinantes, pronomes, preposições, conjunções e advérbios, de modo a optimizar tanto o tempo de pesquisa, como o espaço ocupado pelos documentos, que pode ser reduzido entre 30% e 50% [vR79], sem comprometer demasiado a eficácia do sistema.

É muito comum verificar, em alguns sistemas de RI, a eliminação de palavras mais frequentes apenas na fase de atribuição de pesos (ver secção 2.5.4), pois é nesta fase que é obtida uma quantificação mais precisa da palavra contida em determinado documento.

2.5.3 Radicalização

A radicalização[7] é um processo linguístico de redução de palavras ao seu radical mínimo. Entende-se por radical mínimo a forma morfológica que existe em comum entre as palavras, que na maioria dos casos apresentam o mesmo significado semântico. Por exemplo, as palavras *industrial*, *industrialisation* e *industries* são reduzidas a um único radical, pois todas elas, apesar de serem morfologicamente distintas, referem-se ao mesmo conceito. Assim, torna-se possível a redução do tamanho do ficheiro de dados em cerca de 20% a 50% para colecções pequenas. Em termos de eficácia, os vários estudos que foram feitos revelam que não existe uma variação significativa . Também não se manifesta importante a escolha do tipo de radicalização utilizada. As variações são, isso sim, mais dependentes da natureza das colecções e do seu tamanho [HG96].

[6] Do inglês *stop list*.
[7] Do inglês *stemming*.

2.5.4 Atribuição de Pesos

O processo de atribuição de pesos vem complementar as técnicas anteriores de simplificação de documentos e acrescentar informação útil acerca dos termos que compõem esses documentos. A enumeração dos termos que existem num dado documento, apesar de fazer uma cobertura bastante eficaz, não se torna suficiente para descrever a importância desses termos no seu conteúdo; pelo que, a simples informação da existência, ou não, de um termo no documento, sendo um processo simples de implementar, carece de informação útil que permita ao sistema uma resposta mais eficaz.

Como é possivel, então, quantificar automaticamente o peso que cada termo exerce em cada documento? A frequência com que um termo surge no documento parece ser uma solução promissora. Basta reparar que, num documento quando, uma determinada palavra aparece mais frequentemente, à partida, essa palavra será mais importante que outra que surja menos vezes. Manipula-se aqui o jogo da estatística, ao se pretender obter informação das palavras com base na sua repetição ao longo de um texto. O importante é conciliar o maior número de dados disponíveis ao menor custo. Quando estamos perante uma situação em que a análise sintáctico-semântica é posta de lado, esta informação é de extrema importância. Este tipo de atribuição de peso designa-se por *frequência do termo* (FT).

Foi Luhn [Luh58] quem inicialmente propôs já esta ideia, estabelecendo, no entanto, limites no que respeita a palavras com frequências muito altas e muito baixas. No entanto, o aumento de frequência das palavras não é linear em relação à sua importância para o documento. Para ultrapassar este problema, a solução é utilizar uma função de atenuação para valores mais elevados de frequência.

A atribuição de pesos pode chegar ainda mais longe. Em documentos que possuam caracteres de controle de formatação de texto, como é o caso das linguagens com marcas[8] em que é possível identificar facilmente palavras com conteúdo semântico, pode atribuir-se-lhe um aumento de importância relativa.

A FT não é a única condição que influencia a discriminação de um documento. O peso de um termo pode também ser afectado pela sua distribuição por toda a colecção de documentos, o que é designado vulgarmente pela Inversa da Frequência de Documentos (IFD).

Outro factor a ter em conta é o tamanho do documento, que podem afectar significativamente o desempenho de um sistema de RI, e que é concretizado através de funções de normalização.

A atribuição de pesos depende do modelo de RI utilizado. Como se verá no capítulo seguinte o modelo probabilístico efectua uma dedução matemática destes pesos, ao passo que o modelo do espaço vectorial, efectua-a de uma forma empírica.

[8]Do inglês *Markup Languages*.

Capítulo 3

Modelos conceptuais dos sistemas de Recuperação de Informação

No capítulo anterior, vimos como era possível transformar o texto contido nos documentos em termos armazenáveis em ficheiros de fácil acesso através da indexação automática. Vamos abordar agora, o núcleo de um sistema de RI, onde são realizadas as operações de decisão de quais documentos serão devolvidos ao utilizador.

Quando estamos perante um sistema de RI, pretende-se que este devolva o maior número de documentos relevantes em relação a uma interrogação e consiga ignorar aqueles que não são relevantes.

Uma vez que não é feita qualquer análise semântica ao texto original, contamos apenas com o número de palavras que os documentos têm em comum, conjugado com o factor de relevância (peso) que cada uma tem associado. O processo de indexação automática vem possibilitar a simplificação dos algoritmos de comparação, tentando manter equilibrada a relação desempenho/espaço.

O processo de como esta selecção é feita, define o tipo de modelo conceptual de cada sistema. A figura 3.1 ilustra a decomposição das técnicas utilizadas nos sistemas de RI mais comuns [BC87].

Segundo este autor, a maior subdivisão reside no tipo de comparação feita entre os documentos, a que correspondem dois grupos: comparação exacta e comparação inexacta com melhor aproximação. Na comparação exacta, podem identificar-se as comparações feitas com padrões de texto e as pesquisas booleanas. Nestes dois modelos, os documentos são devolvidos por simples presença de termos contidos na interrogação, resultando apenas uma decisão binária. Um documento ou é ou não é relevante.

Quando nas técnicas de RI é usada uma comparação inexacta, surge a necessidade de obter um modo de poder confrontar os documentos e maximizar uma função de similaridade entre eles. Deixa, assim, de ter-se um modo preciso de procurar

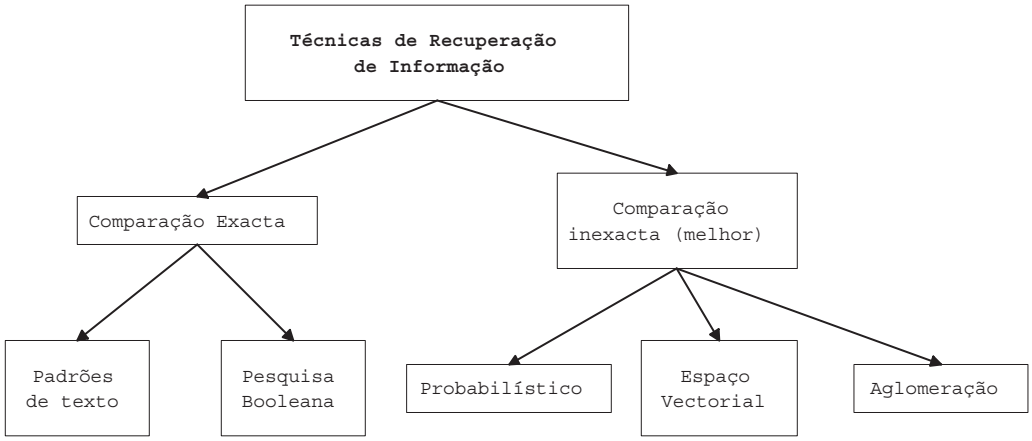

Figura 3.1: Categorização das técnicas de RI

informação nas colecções, e passa a dispôr-se de uma forma de obter, para cada interrogação, um conjunto de documentos potencialmente válidos, ordenados por um valor de semelhança, consoante o resultado da métrica de comparação.

Não existe uma divisão estanque entre as várias técnicas, podendo ser utilizadas em conjunção umas com as outras.

3.1 Modelo de padrões de texto

Este modelo tem como base a comparação directa entre as interrogações e os documentos. Nele podem incluir-se utilitários de pesquisa que façam uso de expressões regulares, como por exemplo o `grep`, o `awk` e ainda o mais poderoso `agrep` [WM92], que para além das potencialidades de utilização com padrões de texto, permite uma comparação aproximada de texto e ainda notação booleana na pesquisa.

Os modelos de padrões de texto são normalmente utilizados em aplicações pessoais de pequena envergadura, ou então como base para aplicações mais complexas, as quais adicionais funcionalidades e tiram partido do tipo de ferramentas deste modelo. Repara-se que as razões destas limitações estão relacionadas com o facto de este modelo operar directamente com os objectos de pesquisa, e não com algum tipo de representante ou simplicação.

3.2 Modelo Booleano

O modelo booleano é dos mais utilizados pelos sistemas comerciais, já que permite uma especificação da interrogação de forma simples e objectiva, permitindo a conjugação dos operadores lógicos entre as palavras que o utilizador pretende pesquisar.

Tal como o modelo anterior, o resultado de sistemas baseados nestes modelos, apenas indicam a relevância ou não de um documento, numa escala binária. Uma característica que permite fazer alguma distinção entre modelos e o anterior, é a sua utilização em colecções de grandes dimensões, dado que já é implementado o conceito de base de dados de termos, onde são armazenadas as representações dos objectos a pesquisar.

Tornou-se bastante comum a utilização deste modelo suportado pelo ficheiro invertido, pois torna-se muito prática a concretização das operações lógicas entre os termos e a organização do ficheiro invertido.

A título de exemplo de uma aplicação do modelo booleano, refira-se o GLIMPSE [MW93] que usa o `agrep` como motor de pesquisa.

Os modelos booleanos devido à simplicidade na definição das interrogações exigem que estas sejam muito limitativas e, por isso, mais adequadas em casos onde o vocabulário dos objectos a pesquisar é restrito ou conhecido à partida.

Para ultrapassar este tipo de problemas, foram desenvolvidos modelos em que a comparação entre os objectos e a interrogação é feita de forma inexacta, atribuindo aos objectos uma ordem de relevância, como se poderá observar como os modelos das secções seguintes.

3.3 Modelo Probabilístico

O modelo probabilístico assume uma abordagem puramente matemática [Sal78] à RI, baseado no estudo da distribuição dos termos pela colecção de documentos, de forma a poder calcular a probabilidade de relevância de um determinado documento face a um conjunto de termos (interrogação).

O desenvolvimento matemático deste modelo situa-se fora do âmbito dos objectivos do presente trabalho [vR79], sendo a sua referência importante para colocar em evidência o aparecimento de algumas relações que são vulgarmente usadas na RI, uma vez que são deduzidas a partir deste modelo [FB93]. São, nomeadamente, as variações que surjem na atribuição de pesos.

3.4 Modelo do Espaço Vectorial

O modelo do espaço vectorial (MEV) trata-se de uma simplificação bem sucedida na RI. O princípio básico deste modelo é considerar cada documento como um vector. A associação entre os documentos e vectores é feita de forma intuitiva, recorrendo a algumas simplificações, incutindo-lhe algumas lacunas. Em primeiro lugar é assumido que o espaço composto pelos termos é independente, o que na realidade, nem sempre acontece, uma vez que existem termos que estão sintactica

e semanticamente associados. É, no entanto, este pressuposto que permite utilizar os modelos matemáticos associados a espaços euclidianos, de que esta técnica tira partido. Assim sendo, torna-se difícil, ao MEV captar as relações inerentes a palavras sinónimas e poliónimas, uma vez que, não permite descrever relacionamente entre os termos.

A figura 3.2 apresenta a representação feita de uma interrogação e de alguns documentos, através da analogia vectorial. Assumindo o espaço euclidiano, facilmente se encontra o documento que se encontra à menor distância da interrogação (o documento B, neste caso).

Apesar dos obstáculos, o MEV continua a ser um modelo bastante utilizado e bem aceite na comunidade científica. Os resultados extraídos deste modelo, continuam a agradar a quem os utiliza, sendo, no entanto, importante a sensibilização de uma transição para métodos mais perfeitos a curto prazo, tal como o desenvolvimento do modelo probabilístico e a introdução de métodos inteligentes .

O melhoramento do MEV é feito dando uma importância diferente aos termos com base em deduções feitas pelo modelo probabilístico ou, inclusivamente, de forma intuitiva, daí a razão de existirem variações nas métricas de similaridade [WS79].

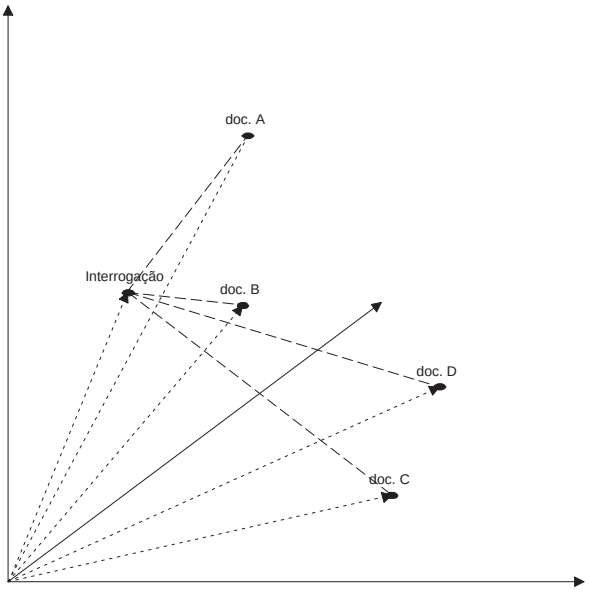

Figura 3.2: Modelo do espaço vectorial

Na secção 2.5, vimos que no final do processo de indexação obtemos uma representação dos documentos através de pares termo/peso. Uma vez que cada termo é único, podemos identificar um documento como um vector, em que cada elemento é o peso de cada termo. Da mesma forma, as interrogações, uma vez que também são um conjunto de termos, podem igualmente ser representadas por um vector.

Assim, representando uma interrogação por Q_i e um documento por D_j, podemos

definir os vectores:

$$Q_i = (q_{i1}, q_{i2}, \ldots, q_{im}), D_j = (d_{j1}, d_{j2}, \ldots d_{jn}) \tag{3.1}$$

em que q_{im} é o elemento do vector que representa o termo m da interrogação i.

Dado que a interrogação é representada por um vector, não existente limitação em relação ao seu tamanho.

Tomando por hipótese que o espaço vectorial é linearmente independente, podemos efectuar o cálculo da proximidade entre documentos, ou entre documentos e interrogações, através do coseno do ângulo formado entre os dois vectores.

$$\cos\theta = \frac{Q_i \cdot D_j}{\|Q_i\|\|D_j\|} \tag{3.2}$$

Sendo a norma dada por:

$$\begin{aligned} \|D_j\| &= \sqrt{d_{j1}^2 + \cdots + d_{jn}^2} \\ &= \sqrt{\sum_{k=1}^{n} d_{jk}^2} \end{aligned}$$

Daqui concluímos que a função de similaridade traduz-se por:

$$sim(Q_i, D_j) = \frac{\sum_{k=1}^{n} q_{ik} d_{jk}}{\sqrt{\sum_{k=1}^{n} q_{ik}^2}\sqrt{\sum_{k=1}^{n} d_{jk}^2}} \tag{3.3}$$

Como já referimos, a suposição de que os termos são independentes trata-se de uma simplificação, já que num texto as palavras estão inter-relacionadas pelas frases que o constituem. No entanto, sem esta simplificação não seria possível proceder ao cálculo da proximidade entre documentos pelo coseno do ângulo formado pelos vectores representantes.

A base de informação para a comparação de documentos, no modelo do espaço vectorial, é o peso associado a cada termo. Já vimos, na secção 2.5.4, que a frequência dos termos (FT) é o peso mais utilizado devido à facilidade de cálculo. O refinamento da FT pode conduzir a resultados bastante mais satisfatórios. Um exemplo disso é a aplicação de uma função de amortização da FT, por causa da não linearidade entre a importância dos termos e a sua frequência. Uma outra forma de minimizar esta inconsistência é a analise da frequência inter-documentos. Nos documentos que tenham uma FT muito alta e que não caracterizem perfeitamente o documento, os termos que lhes pertencem surgirão num maior número de documentos; pelo que se torna aconselhável a entrada desse factor, de modo a diminuir o peso desses termos. Este factor é chamado de Inversa da Frequência de Documentos (IFD):

$$IFD = \log \frac{N}{n_i} \tag{3.4}$$

em que N é o número de documentos da colecção e n_i o número de documentos que contêm o termo i.

3.5 Modelo de Aglomeração

O cálculo da função de similaridade, realizado para o modelo do espaço vectorial, veio trazer inovações no que respeita à comparação dos documentos com as interrogações, mas também na comparação entre os próprios documentos. Ficando à disposição uma forma de compararar documentos, então é possível estabelecer um grau de proximidade entre eles.

3.5.1 Análise de aglomerados

A medida de associação é a base da técnica de análise de aglomerados[1] que é uma técnica de análise estatística multivariante, justamente para gerar categorias de elementos similares num espaço multidimensional [JRSW95]. A estas categorias designamos por aglomerados[2]. Os elementos pertencentes a um aglomerado têm um grau de associação maior entre eles do que em relação a outro elemento de um aglomerado diferente. A análise de aglomerados pode ser encarada como uma forma de classificação automática e tem tido bastante aceitação na comunidade científica para além da RI, nomeadamente na construção de bibliotecas de software [MBK91]. O termo classificação deve, no entanto, ser tomado com cuidado, uma vez que o processo de classificação pressupõe a existência predefinida dos grupos, o que não acontece na classificação automática, já que os grupos são criados à medida que os elementos são associados.

3.5.2 Aglomeração de documentos

A aplicação da análise de aglomerados na RI, é utilizada na repartição de documentos em grupos. Cada um destes conterá documentos que serão de certa forma semelhantes, pois é utilizada uma métrica de similaridade através da qual os documentos podem ser comparados.

A divisão dos documentos de uma colecção em conjuntos mais pequenos é designado por aglomeração de documentos[3], em que cada um possui uma relação de

[1] Do inglês *cluster analysis.*
[2] Do inglês *clusters.*
[3] Do inglês *document clustering.*

semelhança entre si. A aglomeração de documentos é uma técnica que é aproveitada da análise de aglomerados, sendo baseada na chamada *hipótese do aglomerado*. Esta hipótese afirma que documentos associados por aproximação tendem a ser relevantes para os mesmos pedidos [vR79]. Colocando as coisas nestes termos, um sistema de RI, baseado no modelo de aglomeração, pode ser acrescido de um forte aumento em termos de eficiência. Basta pensar que, sendo feita a decomposição de uma colecção em n aglomerados, poderiamos chegar a um limite de optimização do sistema em termos de eficácia, devolvendo ao utilizador o aglomerado de documentos que mais se aproximasse da interrogação, sendo apenas necessário executar $n + \frac{t}{n}$ comparações. No entanto, é necessário também ter em conta a eficiência do sistema.

A selecção do aglomerado mais promissor implica o cálculo de um representante que conterá, em certa medida, toda a informação necessária à substituição do aglomerado. Desta forma, é possível executar comparações entre documentos (ou interrogações) e aglomerados, utilizando o seu representante, que assume a forma de um documento especial. Este representante é também designado por *centroid*, nome que deriva da forma como é calculado.

Seja $D_1, \ldots, D_j$ os documentos de um aglomerado e $d_{j1}, \ldots, d_{jn}$ o vector que representa cada documento, o representante do aglomerado define-se por:

$$C = \frac{1}{j} \sum_{i=1}^{j} \frac{D_i}{\|D_i\|} \tag{3.5}$$

em que $\|D_i\| = \sqrt{d_{i1}^2 + \cdots + d_{in}^2}$.

3.5.3 Métodos de aglomeração

Consoante a estrutura de aglomerados que é criada, é definido o método de aglomeração. A cada método de aglomeração corresponde um algoritmo próprio que é caracterizado pela sua eficiência, requisitos computacionais e complexidade. A distruibuição dos documentos, pelos aglomerados, é dependente, não só pelo método utilizado, mas também pela natureza da colecção. Existem dois métodos de aglomeração principais: não hierárquicos e hierárquicos. Os métodos não hierárquicos organizam os documentos em aglomerados numa estrutura plana. Nos métodos hierárquicos, os aglomerados são criados dinamicamente, obtendo-se uma estrutura hierárquica resultante da subdivisão sucessiva dos documentos em aglomerados.

Métodos não hierárquicos

Este tipo de métodos são, por natureza, os mais simples, quer pela estrutura final que resulta, quer pelo tipo de algoritmos utilizados para gerar essa estrutura. Estes métodos têm uma característica inerentemente heurística, uma vez que existe

uma série de parâmetros que é necessário definir à partida, tal como o número de aglomerados, um limite mínimo de inclusão de documentos nos aglomerados e a possibilidade de sobreposição.

Os algoritmos dos métodos não hierárquicos são baseados num algoritmo denominado de passagem única (APU)[4], que pode ser descrito por:

1. Atribuir o primeiro objecto a um aglomerado, ficando este o seu representante;
2. Cada novo objecto é comparado com todos os representantes dos aglomerados que existam na altura do processamento;
3. O objecto é atribuído a um aglomerado (ou mais, caso exista sobreposição) baseado numa métrica de comparação;
4. Calcular de novo o representante do aglomerado em que o objecto foi atribuído.
5. Se o objecto não obtiver um valor mínimo definido à partida para os representantes existentes, é atribuído a um novo aglomerado;

A simplicidade deste algoritmo permite, no entanto, criar algumas variações, nomeadamente na alteração de parâmetros definidos à partida, nas funções de comparação ou no cálculo do representante.

O seu carácter iterativo leva este algoritmo a deixar que a estrutura dos aglomerados dependa da ordem pela qual os documentos são processados. Por outro lado, a sua simplicidade permite obter $\mathcal{O}(NM)$ em termos de desempenho; onde N é o número de documentos e M o número de aglomerados. Para suplantar o problema da dependência da ordem pode ser utilizado um algoritmo de re-colocação que é iniciado depois da estrutura de aglomerados estar concluída. O objectivo é minimizar os desajustes provocados pela sequência em que os documentos são inseridos e ter uma visão mais global da estrutura, movendo os documentos para aglomerados mais similares.

Métodos hierárquicos

Nos métodos hierárquicos, a estrutura que resulta da decomposição dos objectos tem a forma de uma árvore. Além da estrutura resultante, uma diferença fundamental entre estes métodos e os não hierárquicos, é a necessidade de criar uma estrutura de dados, que albergue o grau de semelhança entre todos os pares de documentos, denominada matriz de similaridade, o que leva a que o problema da dependência de ordem seja eliminada. No entanto, o tempo necessário para a geração da estrutura hierárquica, chega a atingir $\mathcal{O}(N^2)$ podendo chegar a $\mathcal{O}(N^3)$ se for utilizado

[4]Do inglês *Single-Pass algorithm.*

um acesso simples à matriz de similaridade. Em termos de armazenamento pode-se chegar a um aumento na ordem de $\mathcal{O}(N^2)$ se a matriz de similaridade for armazenada em disco. Não sendo armazenada em disco, a ordem de complexidade para os requisitos de armazenamento, fica em $\mathcal{O}(N)$, havendo, no entanto, sempre a necessidade de re-calcular a matriz que tem requisitos em termos de tempo de $\mathcal{O}(N^2)$ [Voo86, FBY92].

Existem vários métodos de aglomeração hierárquica [Bur95] que se distinguem, principalmente, pelo método de associação feita na altura de aglomeração dos documentos. Todos eles se baseiam no seguinte algortmo:

1. Identificar os dois pontos mais próximos e agrupá-los num aglomerado;
2. Identificar e agrupar os dois pontos mais próximos seguintes, tomando os novos aglomerados como pontos.

A matriz de similaridade é de vital importância, pois contém valores numéricos resultantes de uma função de comparação entre todos os pares de documentos. Em cada iteração do algoritmo, o objectivo é calcular o maior dos valores da matriz, sendo o par de documentos associado a esse valor combinado num aglomerado, baseado num valor limite pre-definido. A estrutura é formada de forma natural pela associação sucessiva entre documentos ou documentos e aglomerados já existentes.

3.6 Avaliação de sistemas de RI

Após a observação dos modelos da RI e das técnicas a eles associados, é necessário um método que permita a comparação dessas técnicas ou de outras propostas, de forma absoluta, em termos de eficiência e eficácia. Esta secção descreve a forma como deve ser feita esta avaliação, a fim de se obterem resultados que permitam tirar conclusões concretas acerca da validade de novas alterações introduzidas.

3.6.1 Eficiência

Em relação à eficiência, pretende-se medir da forma mais rigorosa possível os tempos de execução dos algoritmos e a variação no incremento do espaço ocupado pelas bases de dados. No caso concreto, do presente trabalho, dadas as cacterísticas distribuídas do sistema, são, ainda, analisados os tempos de propagação dos dados pela rede.

3.6.2 Eficácia

A análise de uma sistema em termos de qualidade de dados obtidos tem como base, a verificação da relevância dos documentos devolvidos face a uma interrogação.

A questão coloca-se em definir aquilo que é relevância. Será que o conceito de relevância é idêntico para toda a gente? Á partida, quem procura sabe aquilo que pretende, mas este pressuposto pode variar, quando alguém quer analisar o que outra pessoa pretende. Mais difícil, se torna quando é uma máquina a realizar essa tarefa. Daí a complexidade inerente às tecnologias da RI. Antes de qualquer técnica milagrosa para resolver o problema da procura de informação, o seu sucesso depende de quem formula a pesquisa e a forma como o faz. O utilizador deve, antes de mais, ser sensível na criação da sua interrogação e traduzir de forma clara o que pretende.

Este contexto, leva-nos a concluir que o melhor método para a avaliação de um sistema seria o próprio utilizador. É evidente, que este processo de avaliação torna-se dispendioso em termos de implementação, necessitando de um acréscimo de recursos logisticos, que nem sempre são fáceis de concretizar. Não só se torna complexa a aquisição das opiniões, como se torna impraticável a sua comparação, pela natureza divergente de conceitos sobre os temas de pesquisa por parte dos utilizadores.

Colocando de parte esta perspectiva, resta a avaliação através da simulação por computador. Para que isso seja possível é necessário:

1. Colecções de documentos, e um conjunto de interrogações pré-estabelecidas e para as quais são já conhecidos os documentos relevantes;
2. Uma medida de qualidade, baseada nos dados disponíveis nas colecções de documentos.

O processo de avaliação decorre, tendo em conta, para cada interrogação, quais os documentos relevantes devolvidos pelo sistema, que existem no conjunto de documentos relevantes já definidos para essa mesma interrogação.

3.6.3 Colecções de teste

As colecções de teste constituem um dos principais meios para a avaliação de sistemas de RI. Apesar de serem um método um pouco artifical, a sua utilização permite a comparação entre os vários sistemas, ou variações nos parâmetros das técnicas a um nível laboratorial. De modo algum, as colecções de teste traduzem condições reais dos sistemas. As dimensões dos documentos e das interrogações, podem não condizer com a realidade. Por outro lado, o conjunto de documentos relevantes que fazem parte das colecções de teste, são fruto do trabalho de um indivíduo (ou grupo) e, para o qual, não há garantias de que a sua opinião acerca da relevância dos documentos seja universal.

Para a utilização de uma colecção de teste, é importante conhecer as suas características intrínsecas, por forma a se obter a noção do tipo de documentos que se está a manipular. As características dimensionais permitem deduzir resultados

acerca do espaço poupado com as técnicas de indexação. A informação qualitativa, permite a a avaliação do sistema em termos de eficácia. Entre estas características, destacam-se:

1. O número total de documentos;
2. O tamanho dos documentos.
3. O assunto a que se refere;
4. O tamanho e a quantidade de interrogações;
5. A quantidade de documentos relevantes para cada interrogação e a sua identificação;

3.6.4 Precisão e totalidade

Uma utilização genérica de um sistema de pesquisa de informação tem em vista, por parte do utilizador, um conjunto limitado de documentos. Dentro desse conjunto, é possível identificar quais os documentos relevantes e não relevantes para o seu pedido. O objectivo de um sistema é maximizar a quantidade de documentos relevantes devolvidos dentro do conjunto total de documentos. Com base nestes dados podemos definir precisão e totalidade da seguinte forma:

	Relevantes	Não Relevantes	
Devolvidos	r	$n-r$	n
Não devolvidos	$R-r$	$N-n-R+r$	$N-n$
	R	$N-R$	N

Tabela 3.1: Declaração de variáveis para a definição de precisão e totalidade

Sendo P a precisão e T a totalidade a sua definição é feita da seguinte forma:

$$P = \frac{r}{n} \quad T = \frac{r}{R} \tag{3.6}$$

A precisão mede a qualidade da pesquisa, calculada pelo quociente entre o material útil e o número de documentos devolvidos. A totalidade mede a largura da pesquisa, ou seja, o quociente entre o material devolvido e o número de documentos considerados relevantes. Para cada número de documentos devolvidos existe um valor para a precisão e a totalidade. Ambas estão normalizadas entre 0 e 1, e, variam na razão inversa uma da outra. À medida que o número de documentos aumenta, a totalidade, obviamente, aumenta, e a precisão tende a descer, quando surgem documentos não relevantes. É usual apresentar a avaliação dos resultados de

uma interrogação em curvas de precisão-totalidade (P-T), em que cada ponto pode ser calculado estipulando um nível de referência, como por exemplo, o número de documentos devolvidos, ou então em valores standard de totalidade.

3.6.5 Métodos para o cálculo da curva média

As curvas P-T são calculadas em relação a cada uma das interrogações. Para se efectuar o cálculo do desempenho global do sistema, é necessário utilizar um esquema de interpolação das várias curvas. Das técnicas mais conhecidas destacam-se [vR79]:

- Micro-avaliação;
- Macro-avaliação.

A primeira baseia-se no cálculo do somatório das variáveis intervenientes nas fórmulas de precisão e totalidade em cada nível pretendido. Esta técnica é utilizada quando se pretende calcular os valores de precisão e totalidade para um determinado nível. Considerando S o conjunto de pedidos e λ o nível pretendido, podemos definir para o conjunto S:

$$\overline{R} = \sum_{s \in S} R_s \tag{3.7}$$

$$\overline{n}_\lambda = \sum_{s \in S} n_{\lambda s} \tag{3.8}$$

$$\overline{r}_\lambda = \sum_{s \in S} r_{\lambda s} \tag{3.9}$$

Podemos agora calcular os valores da totalidade e precisão no nível λ:

$$T_\lambda = \sum_{s \in S} \frac{\overline{r}_\lambda}{\overline{R}} \tag{3.10}$$

$$P_\lambda = \sum_{s \in S} \frac{\overline{r}_\lambda}{\overline{n}_\lambda} \tag{3.11}$$

A macro-avaliação é utilizada quando se pretende calcular os valores de precisão e totalidade em valores fixos de totalidade. Para executar este cálculo é necessário proceder à interpolação das curvas individuais, uma vez que, para cada interrogação para os mesmos níveis de cada valor fixo de totalidade, a precisão é diferente. O procedimento passa então por calcular para cada interrogação o valor da precisão em cada ponto fixo de totalidade, executando depois a média com o número de interrogações.

Assumindo que λ corresponde ao nível de documentos relevantes devolvidos (tal como no caso da micro-avaliação) e s é uma interrogação pertencente à totalidade das interrogações S, a precisão em cada valor fixo de totalidade é dado por:

$$G_s = (T_{\lambda s}, P_{\lambda s}) \tag{3.12}$$

$$P_s(T) = \{\sup P : T' \geq T \wedge (T', P) \in G_s\} \tag{3.13}$$

Para calcular a média em todos os pontos fixos:

$$\overline{P}(T) = \sum_{s \in S} \frac{P_s(T)}{|S|} \tag{3.14}$$

Por outras palavras, a precisão é definida num ponto fixo de totalidade com o valor máximo em cada par com totalidade maior que o nível pretendido.

Estes são alguns dos métodos utilizados para a avaliação de um sistema de RI. Quer por comparação dos valores de precisão e totalidade para um determinado λ quer por observação directa da curva média é possível efectuar a confrontação das técnicas de RI. No entanto, e apesar de se perder um pouco a noção da evolução do desempenho, é usual efectuar a média da precisão nos valores de totalidade iguais a 0,25, 0,5, 075 [Kee92], obtendo-se, assim, um único valor de fácil comparação. É com base neste valor e na apresentação das curvas de P-T que se fará a avaliação das técnicas apresentadas neste trabalho.

Capítulo 4

Sistema de Indexação e Aglomeração Distribuída (SInAD)

Este capítulo é dedicado à descrição da plataforma que foi desenvolvida conjuntamente com a investigação sobre a *classificação* automática e indexação de informação. É feita uma sucinta abordagem ao estado actual dos sistemas de RI, são descritas ainda as razões que levaram à criação de um sistema de raiz, é ainda feito o desenvolvimento que conduziu à selecção dos modelos conceptuais que foram considerados mais adequados para que fosse possível a melhor optimização para a concretização dos objectivos propostos, procedendo de seguida à descrição dos aspectos de implementação referentes à escolha das técnicas utilizadas e também ao nível de optimizações algorítmicas.

O sistema foi construído para a validação e experimentação das técnicas aqui apresentadas e teve em vista a optimização dos processos de cálculo. A sua utilização tem como base a criação de um banco de testes. Pretensões para tornar a aplicação um utilitário, apenas necessitaria de uma ligeira afinação da flexibilidade na modificação dos parâmetros das técnicas de RI, da criação de uma fachada de interface mais interactivo e da sua interligação a um navegador automático da `WWW`. Isto porque o objectivo aplicacional da ferramenta tem em vista a indexação de páginas da `WWW`, situação esta que foi já simulada com sucesso, requisitando assim apenas pequenas alterações a nível do código para o seu funcionamento perfeito.

4.1 Sistemas disponíveis

Existem actualmente à disponibilidade dos investigadores de RI, alguns programas que fornecem infraestruturas para o teste e experimentação das técnicas de RI. São exemplo o SMART [Buc85],o INQUERY [CCH92] e o OKAPI [RWHB+93]. As grandes vantagens na utilização deste tipo de sistemas são a flexibilidade, versatibilidade,

estabilidade, confiança e uniformização nos resultados.

O sistema SMART utiliza o modelo do espaço vectorial tradicional e guarda a informação dos documentos num ficheiro invertido, estando em vias de implementação o modelo de aglomeração, referenciado já no projecto inicial proposto por Salton [Sal71]. O INQUERY implementa um modelo probabilístico baseado em redes de inferência. Uma das características mais fortes neste sistema é a análise das interrogações, onde se torna possível a sua definição baseada em regras ou conceitos [BCC94]. O OKAPI usa também um modelo probabilístico.

4.2 Razões para a criação de uma plataforma própria

Apesar de todas estas vantagens, a criação de novas plataformas deve ser vista como uma forma de evolução na própria área de pesquisa. Para além de ser criado um projecto com uma perspectiva diferente, uma vez que serão, por certo, seguidos outros caminhos para alcançar os objectivos, podendo evitar-se certos hábitos e mesmo vícios na análise do problema.

A utilização de projectos já estabelecidos, pode também criar uma certa "preguiça" mental para análise de factores de índole intrínseca dos aspectos mais peculiares, como sejam optimizações algorítmicas ou outras técnicas de programação.

Um factor a ter em conta também, é o tempo dispendido na ánalise das outras plataformas, que nem sempre são acompanhadas da documentação mais elucidativa.

Mas o aspecto mais relevante para que se tenha decidido a implementação de um novo sistema está relacionado com a concretização dos objectivos do presente trabalho. Em primeiro lugar, pouca importância tem sido dada aos modelos de aglomeração, que constitui a base teórica do trabalho; em segundo lugar, a gestão da distribuição dos aglomerados em máquina ligadas a uma rede envolve um processo de gestão de recursos relacionados com a transferência de informação através da rede. Algum trabalho de pesquisa foi já realizado na distribuição de sistemas de RI, utilizando um modelo de simulação tendo como base o INQUERY [CM95]. Para além de se tratar de uma situação simulada, o INQUERY não incorpora os aspectos relacionados com a aglomeração.

Ainda, outra questão importante é a avaliação dos resultados obtidos. A quantidade de factores envolvidos num sistema de RI é tão grande e a sua inter-relação tão influente, que basta uma ligeira variação para que os resultados sofram alterações. Só na indexação é possível criar uma imensa combinação de técnicas, que só por si conduzem a variações nos resultados. Como exemplo, basta observar as técnicas usadas na análise lexical do documento. Pode-se chegar ao pormenor de verificar se uma palavra que comece com maiúscula poderá ser mais significativa; se palavras

hifenadas constituem a mesma palavra chave ou não. Tudo isto constitui uma série de considerações as quais podem ser colocadas em segundo plano, já que, partindo do princípio que estamos perante uma plataforma original, poder-se-ão desprezar os efeitos de factores colaterais que poderiam afectar a interpretação dos resultados do presente trabalho. Evitam-se assim desfazamentos na análise de valores obtidos, podendo-se assumir todos os resultados como absolutos, quando comparados entre si.

O que de facto se pretende verificar é o comportamento do desempenho do sistema perante a distribuição dos documentos numa rede, seguindo uma norma de associação entre eles. A criação de um sistema de raiz leva não só a uma melhor compreensão do funcionamento das técnicas de RI, como a um controle absoluto de todos os factores intervenientes no agrupamento de documentos e na sua distribuição. Pretende-se criar uma plataforma de experimentação que seja o bastante versátil e o suficiente eficaz e eficiente de modo que seja possível uma variação ampla nos testes da técnica aqui introduzida. Se bem que o tempo de implementação se tenha tornado crítico, a utilização de outras plataformas iria conduzir à análise de factores que assim se podem considerar desprezáveis, permitindo uma maior concentração nos objectivos propostos.

4.3 Considerações para a implementação de um sistema

A construção de um sistema de RI é uma tarefa que envolve um conjunto de decisões a vários níveis. Até aqui foi lançado o pano de fundo que envolve esta área, ou seja, a sua definição, modelos conceptuais, técnicas utilizadas e validação. Com base nesta exposição vamos agora deduzir as opções que permitem contruir um sistema viável.

Algumas das decisões foram tomadas por mera questão de facilidade de implementação, outras por adequação ao objectivo proposto e ainda as que de facto se mostraram mais eficientes. O acerto de determinados parâmetros foi feito com base na experimentação com o sistema aqui apresentado.

A implementação de sistemas de RI, realizando um estudo profundo de todos os aspectos que foram aqui referidos, é um processo que envolve um trabalho que iria ficar fora do âmbito de uma dissertação de mestrado. Por isso, alguns dos aspectos que não estejam directamente relacionados com a linha que orienta o tema do trabalho, obterão uma importância menos significativa. Assim irão ser focados aspectos relacionados com a distribuição de informação em diversos servidores e a avaliação do seu desempenho quando comparada com sistemas com os mesmos parâmetros em situação centralizada.

4.3.1 Técnicas utilizadas

Nesta secção serão descritos pormenores intrinsecamente relacionados com aspectos da RI, isto é, são esclarecidas as opções tomadas com base na teoria de RI, quais as técnicas utilizadas pela aplicação. A forma como certos aspectos são implementados, sofre algumas alterações a bem do desempenho, pelo que deixaremos esses pormenores de implementação para a secção 4.4.

Estruturas de dados

As primeiras decisões para a implementação deste sistema, estão directamente relacionadas com o suporte dos dados que serão processados. A sua eficiência é fundamental para o sucesso da aplicação. A estrutura utilizada para a persistência da informação é um ficheiro invertido. Apesar da sua organização tão pouco natural, benefecia-se em larga escala de uma enorme eficiência, pois o acesso a um termo da colecção implica o acesso imediato a todos os documentos com esse termo. Na descrição dos algoritmos veremos em pormenor esta característica.

Indexação automática

A indexação automática dos documentos e interrogações foi um dos aspectos mais desprezados, apesar da consciência da sua enorme importância no desempenho dos sistemas. Foram considerados os métodos que simplificassem ao máximo a implementação e utilizados aqueles que estão globalmente estáveis e são bem aceites pela comunidade científica da área.

A análise léxica, foi simplificada a uma mera divisão dos termos separadas por caracteres não alfanuméricos.

O dicionário negativo utilizado consiste num conjunto de 571 palavras retiradas do sistema SMART.

Quanto ao processo de radicalização foi utilizado o *porter stemmer* [Por80] devido à sua tão vasta difusão.

A atribuição de pesos aos termos dos documentos foi feita com base na frequência com que cada termo surge no documento ou interrogação, sendo utilizada a função raiz quadrada para atenuação de frequências elevadas. Foi ainda testado experimentalmente a utilização de uma função logarítmica e apenas a frequência do termo, tendo a raíz quadrada obtido melhor sucesso na colecção estudada, quando comparada com variações que utilizam a função logarítmica.

4.3.2 Modelo conceptual

A decisão na escolha do modelo conceptual, não ofereceu muitas dúvidas, colocando o modelo do espaço vectorial (MEV) no topo das opções possíveis. A sua enorme difusão, assim como a sua simplicidade, oferecem uma base de trabalho que facilitam a implementação, sem pôr em causa o seu sucesso comprovado. Ainda a forte relação existente entre este modelo e o modelo de aglomeração tornou a sua escolha imprescindível.

Supondo que estamos perante um espaço eucludiano de dimensão n, vejamos a forma de representação dos documentos e das interrogações:

$$Q = (qft_1, qft_2, \ldots, qft_n), \quad D_j = (dft_{1j}, dft_{2j}, \ldots dft_{nj}) \tag{4.1}$$

em que qft_i e dft_ij representam a frequência do termo i de uma interrogação e de um documento j, respectivamente.

A atribuição de pesos a cada um deles é feita utilizando a função de atenuação raiz quadrada da frequência do termo, seguida da normalização do peso de cada um dos termos. Estes cálculos, são efectuados durante a inserção de documentos, por questões de optimização, como veremos na fase de implementação. A métrica de similaridade entra, ainda, em linha de conta com a IFD, justificada heuristicamente e comprovado o seu sucesso em várias estudos realizados.

A IDF é dada por:

$$IDF = \log \frac{N}{n_i} \tag{4.2}$$

A sua associação com os termos das interrogações está relacionada com o facto de que este cálculo não necessita efectivamente de percorrer todos os termos dos documentos, mas apenas os das interrogações, como veremos na implementação.

$$q_i = \sqrt{\frac{qft_i}{\sum_i qft_i}} IDF \quad d_{ij} = \sqrt{\frac{dft_{ij}}{\sum_j dft_{ij}}} \tag{4.3}$$

A métrica de similaridade será então dada por:

$$sim(Q, D_j) = \sum_{i=1}^{n} q_i d_{ij} \tag{4.4}$$

Esta configuração foi baseada no estudo feito em [SB88], que se revela bastante eficaz para a colecção que irá ser analisada no capítulo 5.

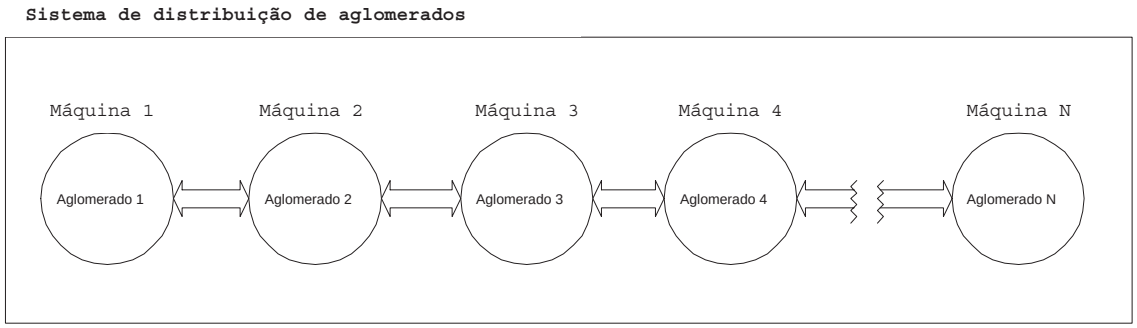

Figura 4.1: Distribuição de aglomerados numa rede de computadores

4.3.3 Conjugação do modelo de aglomeração

A teoria da *hipótese do aglomerado* é sem dúvida bastante pertinente. A possibilidade que nos propõe, de podermos classificar automaticamente as colecções de documentos, torna-a muito atraente, não só em termos de estética, ou seja, documentos associados entre si encontrarem-se virtualmente juntos, mas também em termos de desempenho do sistema, já que o número de comparações da interrogação com os documentos é reduzida substancialmente.

Quer o processo de decisão de inclusão de documentos num aglomerado específico do sistema, quer a devolução de documentos com base numa interrogação, são passíveis de individualização e, por isso, susceptíveis de serem processados independentemente e concorrentemente. Os métodos utilizados no modelo de aglomeração caracterizam-se pela sua carga computacional elevada. Os métodos hierárquicos, apesar de gozarem da vantagem da independência da ordem pela qual os documentos são indexados, a sua ordem de complexidade em termos de tempo para a aglomeração ($\mathcal{O}(N^2)$ [FBY92]) tornam-se inviáveis para colecções de grandes dimensões; ao passo que os métodos não hierárquicos já apresentam uma ordem de complexidade bem mais aceitável, ou seja, $\mathcal{O}(NM)$ em que N é o número de aglomerados e M o número de documentos, sendo $N << M$. Além disso, nos métodos hierárquicos é bem mais difícil a actualização da colecção com novos documentos, devido à necessidade do cálculo da matriz de similaridade que envolve a colecção na sua totalidade, enquanto que os métodos não hierárquicos têm um funcionamento algorítmico tipicamente iterativo e, por isso, adequado na manipulação de colecções dinâmicas, como é o caso da `WWW`.

Para tirar partido das vantagens do modelo de aglomeração e reduzir os problemas inerentes à sua escalabilidade, a solução é aumentar o poder computacional que processa os algoritmos de aglomeração e por consequência os de devolução de documentos. A forma de concretizar tal objectivo, sem ultrapassar as limitações tecnológicas actuais, é recorrer à distribuição concorrente dos algoritmos de aglomeração por diversas máquinas interligadas por uma rede.

Fazendo corresponder cada aglomerado a uma máquina da rede, em que cada uma é responsável pela execução dos algoritmos de aglomeração, podemos obter uma taxa de processamento muito superior, sem sobrecarregar demasiado uma máquina apenas. Por outro lado, qualquer interrogação que se encontre em situação de ser processada, será encaminhada para a máquina que mais se adequa à sua concretização, deixando assim, libertas as restantes para posteriores processamentos. A figura 4.1 visualiza tal distribuição. A existência de uma entidade controladora permite que cada novo documento que dá entrada no sistema para indexação seja, imediatamente, encaminhado para o computador que processa o aglomerado associado. A libertação de recursos desta máquina controladora permite que um novo documento seja de imediato processado e enviado para a máquina correspondente, criando assim uma forma de aumentar o poder computacional de todo o sistema, processando um maior número de documentos de cada vez. A decisão que a máquina controladora toma para enviar o documento para o aglomerado correcto, é baseada em informação previamente criada pelas outras máquinas, no momento de inserção de um novo documento, após o cálculo do representante do seu aglomerado.

A distribuição de aglomerados utilizando o método não hierárquico, para além de permitir um aumento de desempenho na indexação dos documentos de colecções predominantemente dinâmicas, isto é, que não se sabe à partida a quantidade ou dimensão dos seus documentos, vem também permitir uma distribuição do espaço ocupado pelos documentos indexados, pelas máquinas que constituiem o sistema, melhorando assim a eficiência na utilização de recursos, tirando ainda partido de todas as vantagens que advêm da *análise de aglomerados*. Por outro lado, a distribuição vem oferecer uma maior fiabilidade e disponibilidade ao sistema na sua globalidade, já que qualquer máquina de suporte aos aglomerados é um potencial dador/receptor de informação. Significa isto, que qualquer máquina tem capacidade de receber novos documentos e devolver informação perante uma interrogação. Ainda o facto de os documentos semelhantes serem agrupados na mesma máquina supõe-se que essa máquina só por si consiga responder ao "tema" que o utilizador procure. A disponibilidade advém da distribuição em si. Se uma máquina estiver fora de serviço é natural que os documentos aí contidos fiquem inacessíveis, mas o sistema continua o seu funcionamento sem perda de eficácia na devolução dos restantes documentos.

A manutenção da consistência do sistema implica um aumento do tráfego na rede que se manifesta um dos factores que mais contribui para colocar em risco o desempenho do sistema, para além do processo de aglomeração. A utilização de tráfego de rede é necessária para actualizar os representantes dos aglomerados na máquina controladora vindos das máquina que contêm os aglomerados, após a introdução de um novo documento. No momento das interrogações, é também necessário encaminhá-las para a respectiva máquina e esta, por sua vez, enviar o resultado dos documentos relevantes para o utilizador. A análise da influência deste factor é analisada no capítulo 5.

4.4 Implementação do SInAD

A implementação do SInAD foi realizada utilizando a linguagem C++ [Lip91, Str91, GOP90] sob a plataforma UNIX, tendo sido testado e utilizado em PCs com processador $Pentium^{TM}$ a 166 MHz, com 32 Mbytes de memória RAM e 2 Gbytes de disco, com o sistema operativo LINUX. Apesar de ter sido utilizada esta plataforma, o código é perfeitamente portável para outras, desde que se encontrem instaladas as bibliotecas de funções abaixo indicadas. O código é constituído por um conjunto de 37 ficheiros fonte, totalizando cerca de 20.000 linhas. O programa foi implementado de modo a tirar partido do paradigma de orientação para o objecto. Este paradigma não só facilita associação entre modelos reais e abstractos, como também permite uma análise e modificação de código melhorada, aumentando a rentabilidade de produção do software [Lip96]. Para a gestão das estruturas de dados internas (em memória central) foram utilizadas as potencialidades oferecidas pelos tipos abstractos de dados incluídos na biblioteca de C++ (`libg++`), tais como manipulação de conjuntos, vectores e strings. Para as estruturas externas (memória secundária) foi utilizada a biblioteca `libdb`, para implementar a gestão e manipulação de estruturas de árvores-B.

O SInAD é uma aplicação que pode funcionar tanto em modo centralizado como distribuído. A distribuição no SInAD está patente na aquisicão dos documentos e na devolução das interrogações e, ainda, na repartição dos documentos pelos aglomeração, assim como na difusão dos seus representantes.

Por uma questão de organização o SInAD é dividido em entidades funcionais e entidades gestoras. Se bem que uma entidade funcional tem também papel de gestão, pretende-se marcar a diferença entre taferas com níveis de abstração diferentes, ou seja, uma entidade gestora será uma entidade com tarefas mais particulares dentro do modulo funcional que se encontra.

Assim, consideram-se seis entidades funcionais, cujo relacionamento se organiza na figura 4.2:

- a geradora de aglomerados (`crepmake`);
- a gestora de aglomerados e documentos (`docman`);
- a distribuidora de documentos (`lbot`);
- a inquiridora de documentos para a `WWW` (`queryweb`);
- a distribuidora de interrogações para fins de análise de desempenho (`query`) e
- a avaliadora de resultados (`evaluate`).

Não sendo as duas últimas necessárias ao funcionamento do sistema, são indispensáveis para análise dos resultados. Existem ainda três entidades gestoras englobadas no `docman`: A gestora de mensagens, a gestora de documentos e a gestora

de representantes de aglomerados. Estas duas últimas são manipuladas por uma entidade que faz o processamento de acções vindas da gestora de mensagens e que têm como objectivo a manipulação dos ficheiros invertidos associados a cada um dos aglomerados (`docman`). A criação destas entidades é fundamental para o desempenho global do sistema, uma vez que o acesso aos ficheiros invertidos reveste-se de uma grande sensibilidade a esse facto, devido à pouca rapidez no acesso a memórias secundárias. Por outro lado, a biblioteca utilizada para a gestão de ficheiros (`libdb`) possui mecanismos de optimização de escrita e leitura através da criação de *caches* em memória RAM. Uma vez que o acesso aos ficheiros poderá ser sucessivamente frequente entre as actualizações de novos documentos e cálculo de resultados de novas interrogações, o fecho de descriptores de ficheiros envolvia a actualização da *cache* no disco, não se retirando, assim, vantagem na sua utilização, dada a lentidão do processo.

Para solucionar este problema foram, criados para cada um dos ficheiros invertidos, manipulados por cada uma das entidades `docman`, um processo, correspondente a uma entidade gestora, que se encarrega da gestão desses mesmos ficheiros. O tempo de vida destes processos coincide com o tempo de vida da entidade `docman`, minimizando-se, assim, os tempos de acesso que envolvem acções sobre os ficheiros invertidos.

`crepmake`

O `crepmake` serve de parametrização do sistema ao nível da configuração dos aglomerados. São enumerados os nomes das máquinas em que cada aglomerado irá residir, assim como, o porto de comunicação necessário para a transferência dos dados pela rede. Esta operação deve ser realizada em todas as máquinas que irão pertencer ao conjunto de aglomerados do sistema, por forma a que cada uma delas tenha conhecimento das restantes.

`docman`

O gestor de aglomerados e documentos constitui o núcleo do sistema, já que, implementa, praticamente, toda a funcionalidade do SInAD. É ele que processa os documentos vindouros, que serão acrescentados à base de dados e responde perante as interrogações dos utilizadores. É também, o `docman` que gere a criação e a distribuição dos representantes de aglomerados. O `docman` funciona com categoria de *daemon* e tem características de servidor e cliente simultaneamente. Como servidor, fica à escuta de mensagens vindas de outras máquinas, e como cliente envia mensagens, que permitem a comunicação com as outras que formam o conjunto de aglomerados.

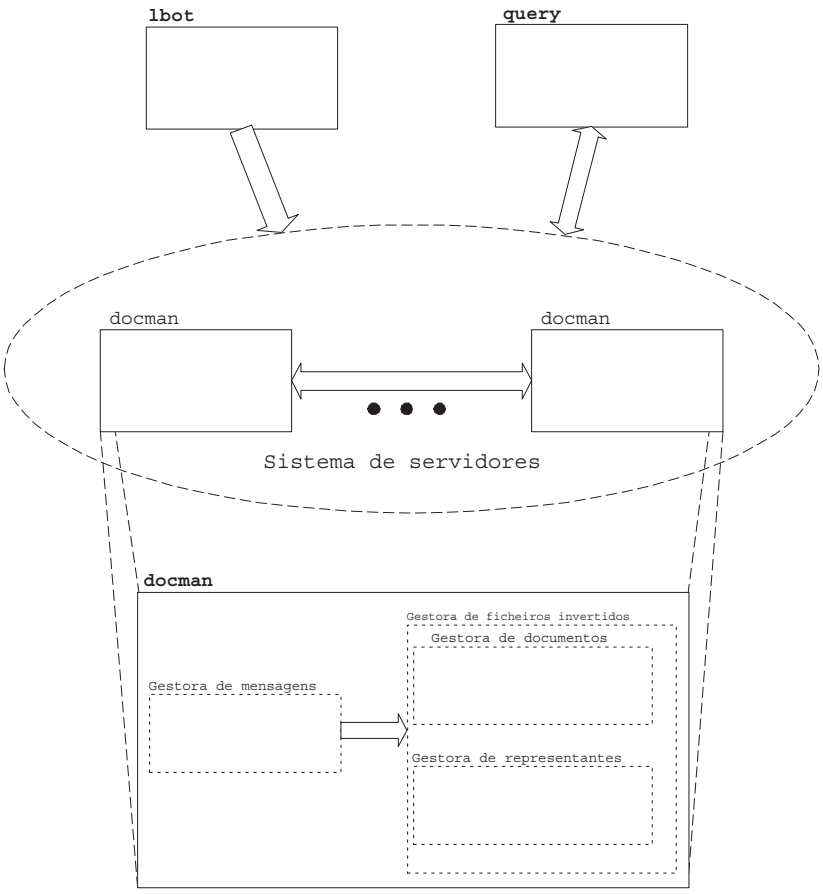

Figura 4.2: Diagrama das entidades do SInAD

`lbot`

O `lbot` foi criado no sentido de providenciar o fornecimento de documentos das colecções de testes aqui analisadas. A informação é enviada em texto original para o `docman`, onde é feito o seu processamento. Em termos práticos o `lbot`, será substituído por um *robot*, que se encarregará de percorrer a `WWW` automaticamente, e enviar a informação para um servidor especificado.

`queryweb`

O `queryweb` trata-se de uma CGI (*Common Gateway Interface*) que permite a interligação aos servidores do sistema através de uma página da `WWW`. Esta página contém um formulário onde é possível introduzir a *query* que o utilizador pretende pesquizar, sendo-lhe devolvidos os resultados dessa mesma pesquisa numa listagem de documentos indexados por ordem decrescente de relevância, com base nos critérios utilizados pelo sistema.

`query`

O `query` é um programa de teste que acciona o sistema perante um conjunto de interrogações pertencentes às colecções de teste. As respostas às interrogações são devolvidos do `docman` para o `query` e os resultados armazenados em ficheiros para posterior análise. Qualquer um dos servidor está apto a responder a pedidos do `query`. Numa situação real, este programa é substituído, por um interface mais amigável, que permita uma interacção mais dinâmica com o utilizador.

`evaluate`

A análise do desempenho do sistema é feita com o programa `evaluate`. Os resultados são calculados em termos de uma tabela de Precisão/Totalidade com intervalos de 0.1 de totalidade, tomando os dados obtidos com o `query`.

4.4.1 Organização das estruturas de dados

Antes de se prosseguir para uma descrição do funcionamento do SInAD, resta descrever como foram organizadas as estruturas de dados internas que o constituem. Não se pretende de forma alguma, enumerar exaustivamente todas as estruturas utilizadas e definidas no sistema, mas, sim, dar a entender o suporte que faz a manipulação dos dados e a interacção entre eles.

É conveniente distinguir entre o suporte de dados que é manipulado internamente para a concretização das operações sobre aglomerados e documentos; e a estrutura externa de armazenamento persistente da informação. Uma vez que foi adoptada uma abordagem orientada para o objecto, foi possível tirar partido da abstração e encapsulamento de dados, tornando o processo de implementação mais coerente e modular. A figura 4.3 apresenta a relação hierárquica entre os objectos considerados mais importantes. Outras estruturas foram utilizadas, no entanto, pode-se desprezar a sua existência para a compreensão do funcionamento do sistema neste contexto.

Classe `ObjectRW`

A `ObjectRW` implementa toda a funcionalidade necessária para se efectuar a leitura e escrita dos objectos do sistema através de um canal de dados. Esta classe é aproveitada para transferir dados entre entidades remotas, através de um *socket* e, também entre as entidades locais através de uma *pipe*, sendo o meio de comunicação criado na classe que herda a `ObjectRW`, ou seja a classe `Socket` e a classe `InvMan`. O tipo de objectos possíveis de ser transmitidos são:

- `char`;

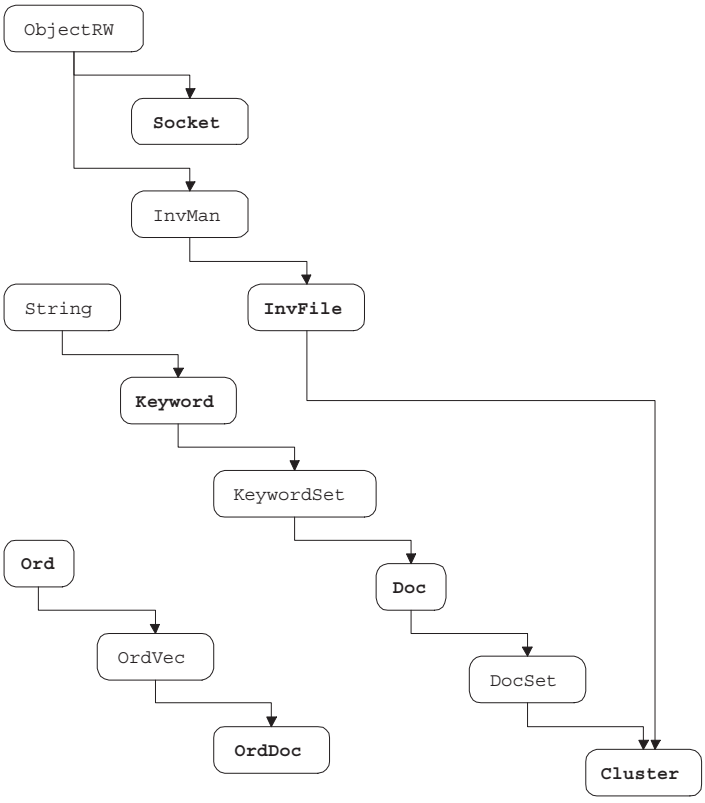

Figura 4.3: Relacionamento entre os objectos do SInAD

- `float`;
- `Keyword`;
- `Doc`;
- `Ord`;
- `OrdDoc` e
- `ODBT`.

A criação deste objecto veio trazer consideráveis vantagens em termos de abstração na troca de informação entre as entidades, devido, nomeadamente, à complexidade de objectos como o `Doc` e o `OrdDoc`, que serão descritos mais à frente.

Classe `Socket`

A `Socket` é a classe que inicializa todo o processo de aquisição e devolução de dados. São implementadas todas as operações relacionadas com a transferência de dados

entre clientes e servidores. Dado que esta classe tira partido das potencialidades da `ObjectRW`, fica, então, apta a trocar qualquer tipo de dado entre as instâncias definidas a partir dela. Para além disso, é esta classe que processa as mensagens que chegam a partir dos clientes, que também são enviadas por ela. A implementação desta classe, conseguiu concentrar os aspectos da comunicação entre as entidades envolvidas, de forma a obter uma maior abstração em camadas de código superiores.

Classe `InvMan`

A troca de informação entre as estruturas internas e externas, é feita através do interface criado por esta classe. Basicamente, são construídas duas *pipes* que irão permitir a comunicação entre o processo que aceita as mensagens (classe `Socket`) e o processo que se encarrega da leitura e escrita na estrutura de dados persistente.

Classe `InvFile`

É nesta classe que é feita a implementação do ficheiro invertido. A cada um dos aglomerados está associado um ficheiro invertido, que contêm os termos dos documentos que pertencem a esse aglomerado.

Por uma questão de pura eficiência, as operações efectuadas sobre o ficheiro invertido são feitas através de um processo que actua exclusivamente sobre esse ficheiro, daí a necessidade da *pipe* de comunicação. Esta decisão, teve sérias repercursões a nível da eficiência do sistema, já que permitiu um melhoramento no desempenho do acesso ao ficheiro invertido. Assim, são evitadas constantes aberturas e encerramentos dos descritores dos ficheiros, que consomem bastante tempo de execução, devido à tranferência de dados da memória para o disco. Por outro lado, a biblioteca que foi utilizada para implementar o ficheiro invertido (`libdb`), possui um formato em árvores-B com aproveitamento de cache em memória central. Cada vez que o ficheiro é fechado, a cache é despejada e, por isso, deixa de se tirar partido dela em acessos subsequentes.

Classe `Keyword`

A classe `Keyword` permite associar um termo a um peso. A `Keyword` é utilizada para representar os documentos, as interrogações e os representantes dos aglomerados.

Classe `Doc`

Esta é uma das classes com maior diversidade na sua utilização. É composta pela definição de um conjunto de `Keyword`s a partir da classe `KeywordSet`, que por sua vez é derivada de uma classe da biblioteca `libg++`, onde está implementada uma classe

de conjuntos genéricos. Além disso, contém também o nome do documento (título), um identificador único de documentos e outra informação adicional temporária.

Classe `Cluster`

Esta classe representa um conjunto de `Doc`, implementado pela classe `DocSet`, derivada, também, de uma classe da biblioteca `libg++`. Existe ainda informação acerca do nome do aglomerado, que corresponde ao nome da máquina definido pelo `crepmake`. A cada uma destas classes está associada uma classe `InvFile`, de forma a permitir o armazenamento persistente. Quando um documento é adicionado a um aglomerado, ele é imediatamente actualizado no ficheiro invertido.

Para manter uma referência dos documentos que se encontram num aglomerado, é armazenada uma instância de `Doc`, apenas com a informação do nome e do identificador único, assim é possível aceder rapidamente aos documentos do aglomerado poupando o máximo de memória possível. Se bem, que estas referências em memórias seriam desnecessárias, a utilidade, é puramente para controle, pois grande parte das operações que envolvem acesso a documentos são feitas tirando partido do acesso directo ao ficheiro invertido, isto é, o acesso é feito por termo.

Classe `Ord`

Esta classe é utilizada para associar a um documento um peso, após a sua comparação com uma interrogação.

Classe `OrdDoc`

Esta é a classe que suporta um vector de `Ord`s, construído a partir da ordenação de um conjunto de representantes de aglomerados ou documentos face a uma interrogação. A implementação do vector é feita utilizando uma classe da `libg++`, sendo o interface feito pela `OrdVec`.

4.4.2 Comunicação entre as entidades

Após a discussão das entidades funcionais existentes e as estruturas de dados mais importantes, vamos de seguida esclarecer o funcionamento das entidades gestoras e a forma como é feita a comunicação entre estas entidades e as entidades funcionais.

Das entidades funcionais referidas, podem excluir-se o `crepmake` e o `evaluate`, uma vez que estes tiram partido da informação residente no disco para a parametrização do sistema e avaliação de resultados, respectivamente, e por isso, não são considerados nesta secção.

Figura 4.4: Estrutura das mensagens processados pelo `docman`

A comunicação entre as entidades funcionais, é feita através do conjunto de mensagens apresentado na figura 4.4. O relacionamento destas entidades pode ser revisto na figura 4.2. O facto de as entidades funcionais poderem residir em localizações remotas, isto é, em máquinas diferentes, a comunicação é feita através da utilização de `Unix Sockets`.

A entidade que se encarrega do processamento das mensagens é a gestora de mensagens inserida dentro da `docman`, que pode também actuar como cliente, tal como a `lbot` e a `query`. A gestora de mensagens encarrega-se de processar todas as mensagens que desencadeiam actividades na `docman`:

1. Receber um novo documento da `lbot`;
2. Receber um documento de outra `docman`;
3. Receber um representante de um aglomerado de outra entidade `docman`;
4. Receber o texto de uma interrogação da entidade `query`;
5. Receber uma interrogação de outra entidade `docman`.

Em resposta a uma interrogação entre as entidades `docman` (ponto 5) ou para a entidade `query` (ponto 4), o sistema responde com uma mensagem do tipo apresentado na figura 4.5, contendo uma estrutura `OrdDoc`.

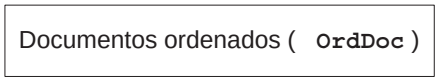

Figura 4.5: Mensagem de resposta a uma interrogação

A comunicação com as entidades gestoras dos ficheiros invertidos é feita através do envio de mensagens específicas a cada acção a efectuar no ficheiro correspondente, utilizando, para isso, uma *pipe*, como método de comunicação entre processos.

4.4.3 Aquisição de novos documentos

Um dos processos fundamentais no funcionamento do SInAD é a aquisição de novos documentos. É aqui que as bases de dados são actualizadas e onde são feitas as decisões de encaminhamento de documentos para os aglomerados respectivos, quando a aplicação funciona em modo distribuído. Os documentos são recebidos pela entidade `docman` e enviados pela `lbot`, através da rede. A informação enviada consiste no título do documento, que para uma aplicação do SInAD na `WWW` pode ser a URL de uma página, e o texto original desse documento. O `docman` encarrega-se de executar a indexação do documento conforme o apresentado na secção 4.3.1, incluindo a afectação da função de atenuação (raíz quadrada) e a sua posterior normalização, melhorando-se, assim, o desempenho no cálculo da função de similaridade. De seguida, o documento indexado é confrontada com os representantes de cada um dos aglomerados, sendo o documento adicionado no aglomerado que obteve melhor classificação na função de seriação. Quando o SInAD é executado em modo distribuído, o documento é enviado para o servidor que contém o aglomerado referido. Após a adição do documento ao respectivo aglomerado é calculado o seu representante, sendo distribuído por todos os servidores que fazem parte do sistema.

Para melhorar o desempenho na aquisição de novos documentos, foi incorporado um sistema de semáforos, de forma a permitir o processamento controlado dos documentos. Como o cálculo do representante dos aglomerados é um processo que envolve um enorme gasto de poder computacional, sendo esta, inclusivamente, a razão principal para a distribuição dos aglomerados, não é conveniente que os documentos sejam enviados para os servidores de aglomerados indiscriminadamente. Com o sistema de semáforos é possível controlar o número de representantes que são calculados simultaneamente, evitando uma sobrecarga do processador que calcula os representantes, optimizando o número de documentos indexados por unidade de tempo.

Ordenação dos representantes dos aglomerados

Sendo os representantes o *centroid* do aglomerado, este é tomado como um documento, e, por conseguinte, passível de ser comparado através da métrica de ordenação referida na secção 4.3.3. Quando o sistema se depara com aglomerados que não contêm nenhum documento, torna-se necessária a consideração de um parâmetro que apenas inclua um documento num novo aglomerado, caso esse parâmetro não seja ultrapassado. Este factor é de extrema importância devido ao modelo de aglomeração utilizado ter uma forte dependência da ordem pela qual são inseridos os documentos. Com base neste parâmetro, só é utilizado um novo aglomerado quando não houver um aglomerado com documentos, que ultrapasse esse limite. Assim, é possível moderar a inclusão de um documento num aglomerado de forma a permitir uma melhor dispersão da colecção pelos aglomerados. Veremos a influência deste factor no capítulo 5.

Uma questão muito pertinente na ordenação dos representantes e também verificada na ordenação dos documentos, é a optimização que é efectuada para a computação da fórmula 4.4. Para além de já se ter verificado que os documentos são normalizados antes do cálculo da métrica, o que permite um melhoramento no momento deste cálculo, não se torna necessário efectuar as n operações ali descritas, mas sim, apenas o número de termos contido na interrogação. Esta simplificação, naturalmente óbvia, só é atingivel devido á organização dos documentos de forma invertida, o que permite a procura na base de dados dos documentos que contêm apenas os termos que se encontram na interrogação.

Na comparação do documento com os representantes dos aglomerados não foi considerado o peso IFD, para além de não fazer muito sentido, verificou-se que quando era utilizado os documentos concentravam-se num número de aglomerados muito reduzido.

Cálculo dos representantes dos aglomerados

O cálculo do representante dos aglomerados é realizado no servidor em cujo aglomerado o documento foi inserido. A dimensão dos representantes é um factor que pode ser determinante tanto para a eficiência como para a eficácia do sistema. Um representante ideal seria aquele que contivesse todos os termos e respectivos pesos calculados do aglomerado respectivo, contudo, devido à necessidade em trasferir os representantes pela rede, este processo pode tornar-se deficiente em termos de tempo e espaço. Para controlar este factor foi introduzido um parâmetro que indica quais os termos com peso superior a esse parâmetro são aproveitados para o representante. Desta forma, consegue-se criar um representante com os termos mais relevantes dentro do aglomerado. Naturalmente, o acerto deste parâmetro é feito por experimentação e a sua influência será analisada no capítulo 5.

4.4.4 Consulta ao sistema

A pesquisa de documentos relevantes indexados no sistema é feita a partir das entidades `query` ou `queryweb`. Após a recepção da mensagem pela `docman`, é feita a indexação do texto original, onde estão englobados os processos referidos na secção 4.3.1, seguida da sua normalização. O passo que se segue, consiste na decisão de qual dos aglomerados é mais promissor de conter mais e "melhores" documentos. Esta seriação é feita de forma idêntica à inserção de um novo documento. À máquina que contém o melhor aglomerado, é enviada a mensagem `DISTQUERY`, a que este responde com a seriação dos documentos contidos neste aglomerado enviando-os de volta para a máquina que enviou a mensagem, que por sua vez os devolve para o cliente que efectuou a interrogação.

Nesta tarefa uma vez mais, é feita a seriação de representantes de aglomerados de forma semelhante à da tarefa de aquisição de novos documentos. Quanto à seriação dos documentos, há que salientar a inclusão do factor IFD, que como veremos aumenta substancialmente o desempenho do sistema.

Capítulo 5

Experimentação com a colecção *Cranfield*

Nos capítulos anteriores, foi analisada a teoria que envolve a RI, e lançados todos os dados necessários para compreensão da variação que se pretende introduzir no âmbito dos sistemas de RI. Foi também explicado com pormenor os aspectos relacionados com a implementação da ferramenta que, para além de permitir a colocação dos objectivos pretendidos em prática, constitui uma bancada de experimentação que conduzirá à extração das conclusões acerca da validade da técnica aqui apresentada e suas variações.

Para se poder efectuar uma comparação precisa foi feito uma série de experimentações com sucessivas modificações de parâmetros ou acréscimo de outros factores. Começou-se por avaliar a técnica base e a partir dessa extrapolar os resultados para as restantes experiências e, assim, retirar as conclusões.

A experimentação é baseada num conjunto de documentos, e um conjunto de interrogações pré-definido, a que a cada uma, está associado um conjunto de documento considerados relevantes. Todas estas informações encontram-se reunidas naquilo que é chamado de colecções de teste. Este capítulo foca a colecção *Cranfield* e o conjunto de experimentações efectuados com ela.

O estudo das várias experiência foca essecialmente 2 aspectos fundamentais:

- A eficácia do sistema medida atráves de curvas P-T, sendo apresentada a média de precisão nos 9 pontos de totalidade em que a precisão é calculada, para facilidade de comparação.
- A eficiência do sistema medida em segundos como sendo o tempo de resposta a cada uma das interrogações.

Um outro aspecto ainda focado, se bem que, não sendo crucial para o desempenho do sistema, mostra o tempo que cada documento demora a ser nele introduzido. Este

factor, tornou-se de alguma forma condicionante na execução das experiências dado o elevado tempo de indexação que as colecções levavam. Por fim, note-se que em modo de aglomeração o tempo de inserção de um documento pode disparar para valores bastante críticos, devido ao cálculo dos representantes dos aglomerados. Por outro lado é de esperar, um aumento destes tempos, na inserção, assim como na resposta a interrogações, devido ao tempo de propagação desses mesmos aglomerados, quando em modo distribuído.

5.1 A colecção *Cranfield*

A colecção *Cranfield* é um conjunto de resumos de artigos na área da engenharia aeronáutica.

As razões pelas quais a colecção *Cranfield* foi escolhida baseiam-se:

1. na quantidade de documentos, que é relativamente pequena, e por isso, prática quando se pretende efectuar uma quantidade de testes elevada;
2. na quantidade de interrogações que fica acima da média de outras colecções, permitindo uma simulação mais aproximada da realidade; e
3. nos bons resultados obtidos, quando comparados com outras colecções, utilizando a atribuição de pesos aqui apresentada.

A colecção é composta por 1397 documentos e 225 interrogações. O vocabulário da colecção, ou seja, o número de termos diferentes após a utilização do dicionário negativo e o algoritmo de radicalização já referido, é de 5311, a média de termos por documento é 55,76 e a média de termos numa interrogação é 8,94.

Podem-se resumir as reduções percentuais no número de termos devido à sua igualdade, à eliminação pelo dicionário negativo e pelo processo de radicalização, em relação ao número de termos dos documentos originais, na tabela 5.1.

	Redução (%)
Por igualdade	47,97
Pelo dicionário negativo	67,42
Por radicalização	69,51

Tabela 5.1: Percentagens de redução do número de termos em relação ao documento original.

O que constitui uma redução total bastante significativa de cerca de 70%. Um exemplo de um documento e de uma interrogação original encontram-se demonstrados nas figuras 5.1 e 5.2.

```
.I 5
.T
one-dimensional transient heat conduction into a
double-layer slab subjected to a linear heat input for
a small time internal .
.A
wasserman,b.
.B
j. ae. scs. 24, 1957, 924.
.W
one-dimensional transient heat conduction into a
double-layer slab subjected to a linear heat input for
a small time internal .
analytic solutions are presented for the transient heat
conduction in composite slabs exposed at one surface
to a triangular heat rate . this type of heating rate
may occur, for example, during aerodynamic heating
.
```

Figura 5.1: Exemplo de documento da colecção *Cranfield*.

```
.I 006
.W
what theoretical and experimental
guides do we have as to turbulent
couette flow behaviour .
```

Figura 5.2: Exemplo de uma interrogação da colecção *Cranfield*.

5.2 Experimentação base

Para que se possa fazer uma sequência lógica na modificação dos parâmetros e na introdução da técnica aqui apresentada e, uma vez que ela é baseada na conjugação de outras já existentes, começou-se por analisar a experiência base que consiste na utilização do modelo do espaço vectorial simples, sem qualquer tipo de aglomeração e num ambiente centralizado ao nível da execução dos algoritmos de inserção de documentos. Todos os parâmetros relacionados com a RI, são utilizadas as configurações referidas no capítulo 4.

O envio de novos documentos é feito pela entidade `lbot` para a entidade `docman` de forma remota. Este facto tem a ver com a preparação da aplicação como um servidor independente, de forma a possibilitar o seu funcionamente com a indexação da `WWW` e como banco de teste, simultaneamente. Para simplificar o processo de comparação de técnicas, este tempo de propagação não é considerado. Em contrapartida, o tempo que as respostas às interrogações levam a chegar da `docman` à entidade `query` já é considerado.

As experiências subsequentes têm como padrão os resultados extraídos desta ex-

periência. As razões para esta decisão apontam para o facto de o MEV ser tomado como obtendo bons resultados, e por ser um modelo simples a nível de implementação [Hul94].

Dado que não está a ser avaliado o desempenho do sistema por alteração de factores que afectam a parte de indexação, é mais uma boa razão para a realização de uma experiência base, de forma a colocar em questão apenas os factores relacionados com a aglomeração e distribuição. Como já foi referido na secção 4.3.1 optou-se por utilizar um modelo simples de indexação que, mesmo assim, conseguiram obter-se resultados bastante satisfatórios quando comparados com outras experiências [VF95].

5.2.1 Eficiência

A avaliação do sistema em termos de eficiência divide-se em duas partes:

1. O tempo de inserção dos documentos na base de dados;
2. O tempo de resposta às interrogações.

Apesar de a primeira não ser vulgarmente considerada neste tipo de experiências, uma vez que o tempo de inserção dos documentos não afecta directamente o desempenho nas respostas ao utilizador, no entanto, uma vez que o sistema aqui descrito utiliza um modelo de aglomeração, é de se considerar tal medida, já que este é um dos pontos sensíveis neste tipo de modelo.

A análise dos tempos de resposta é de importância vital, pois é precisamente neste campo que se pretende dar alguma contribuição a partir dos resultados deste trabalho.

É de salientar, que todos estes valores, carecem, no entanto, de exactidão absoluta. Por um lado, todos os tempos medidos são adulterados por código de *logging* e *debug*, por outro, os tempos de propagação de informação na rede, são medidos entre o envio da informação e a recepção de uma mensagem de reconhecimento, o que não traduz correctamente o tempo de propagação efectivo. Uma solução seria a utilização de um esquema que medisse a média de tempos de ida e volta do mesmo tipo e quantidade de informação. Este esquema, não foi, no entanto, utilizado, devido ao aumento de entropia no sistema, o que conduziria a experimentações demasiado demoradas. Quanto à medição do espaço ocupado, também ela sofre de falta de rigor, já que, a biblioteca utilizada para a gestão dos ficheiros, implementa uma *cache* que, optimiza a escrita em blocos de dados de tamanho uniforme, perdendo-se ligeiramente a noção da quantidade de dados envolvida em cada iteração que se pretenda medir. Para acrescentar a tudo isto, deve ainda notar-se, o equipamento modesto em que as experiências foram realizadas.

Resumindo, apesar das pequenas faltas de rigor, pretende-se efectivamente, obter um plano de fundo, que permita a comparação relativa entre as várias técnicas experimentadas.

Inserção dos documentos

Para se ficar com uma ideia dos tempos de inserção de novos documentos, apresenta-se na figura 5.3 o gráfico dos tempos em segundos que cada documento demora a ser inserido na base de dados.

Por obervação do gráfico, facilmente se conclui que este não expressa uma boa medida para analisar a evolução dos tempos de inserção devido aos picos que se verificam com alguma frequência. Depois de terem sido feitas mais algumas experiências neste sentido, nomeadamente a decomposição mais pormenorizada dos tempos em cada fase da inserção de documentos, verificou-se que estes picos são justificados na sua maior parte pelo esvaziamento da *cache* que é manipulada pela biblioteca de funções utilizadas para o processamento dos ficheiros invertidos (`libdb`). Por outro lado, e devido ao facto de esta biblioteca não incluir um sistema de bloqueio de leitura e escrita, foi necessária a sua implementação a um nível mais alargado que corresponde a um bloqueio por cada operação realizada pela entidade gestora de ficheiros invertidos quando são manipulados os dados destes ficheiros. Logicamente, este bloqueio não deverá ser tão eficiente como se fosse implementado dentro da própria `libdb`.

Para assegurar uma comunicação viável entre esta entidade e a gestora de mensagens foi também necessária a criação de um sistema de bloqueio que não permitisse a mistura das mensagens que eram processadas e assegurar a atomicidade das operações.

Com base nisto, optou-se por colocar de parte a apresentação deste tipo de gráfico nas experiência seguintes e, utilizar o valor da média dos tempos de todos os documentos. Esta medida, não só facilita a comparação entre as experiências como torna mais clara a leitura dos resultados.

Tempo Médio (s)
0,1127

Tabela 5.2: Tempo médio de inserção de documentos para a colecção *Cranfield* sem aglomeração.

Dado que o gráfico dos tempos se mostra de difícil visualização, fica de alguma forma complicado realizar um estudo sobre a escalabilidade do sistema. Uma vez que grande parte do tempo é utilizado na escrita dos dados em disco e as operações são de $\mathcal{O}(\log(n))$, podemos interpolar os pontos com um função logarítmica e assim

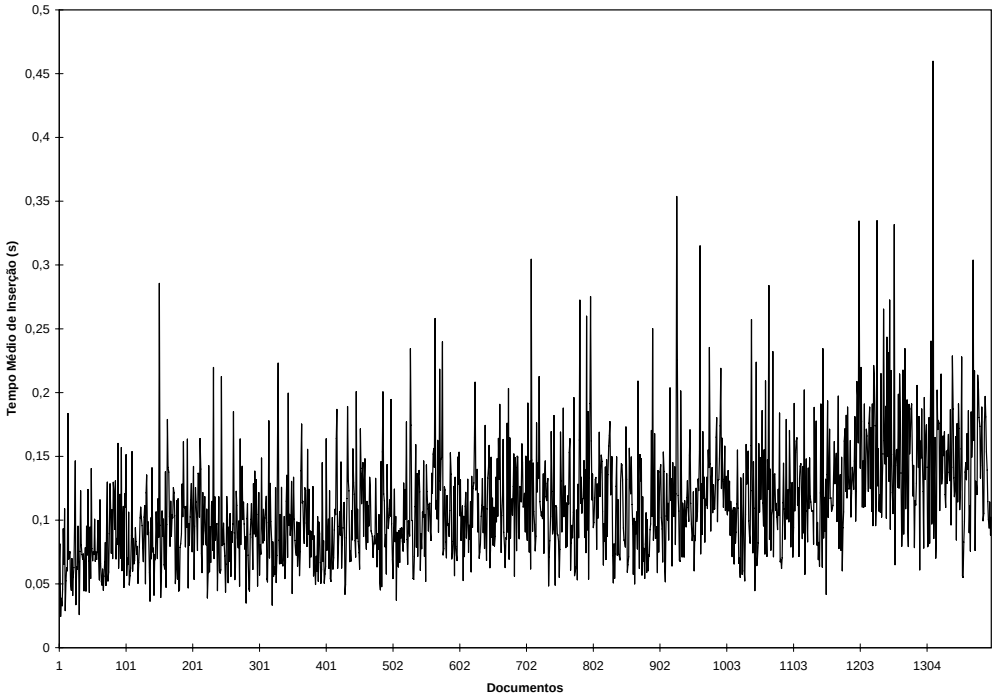

Figura 5.3: Tempos de inserção de documentos da colecção *Cranfield*.

estimar os tempos para quantidades de documentos superiores às das colecções de teste. Obviamente que, não se trata de um estudo rigoroso, mas a utilização prática deste tipo de sistemas aponta para bases de dados de dimensões astronómicas, como é o caso da indexação das páginas da `WWW`.

Resposta às interrogações

Tal como a medida anterior, o tempo de resposta às interrogações apresenta oscilações bastante incómodas para se obter uma estimativa do tempo com base na observação do gráfico (Figura 5.4). A grande parte das variações bruscas podem ser justificadas pela variação do número de termos presentes na interrogação, no entanto, após uma análise mais pormenorizada entre o comprimento da interrogação e o tempo de resposta levou a concluir que não existe uma relação constante. A razão deste facto é justificada pelos motivos referidos no ponto anterior.

Qualquer tentativa para se concretizar uma estimativa do tempo de resposta em função do número de documentos da colecção torna-se praticamente incalculável, pois estes tempos são medidos para uma quantidade fixa de documentos, sendo esta a única forma de tirar partido das informações disponíveis das colecções de teste.

Mesmo assim, torna-se fundamental avaliar este factor e dispôr de uma medida de comparação para as experiências seguintes. Vai-se, por isso, apresentar o tempo

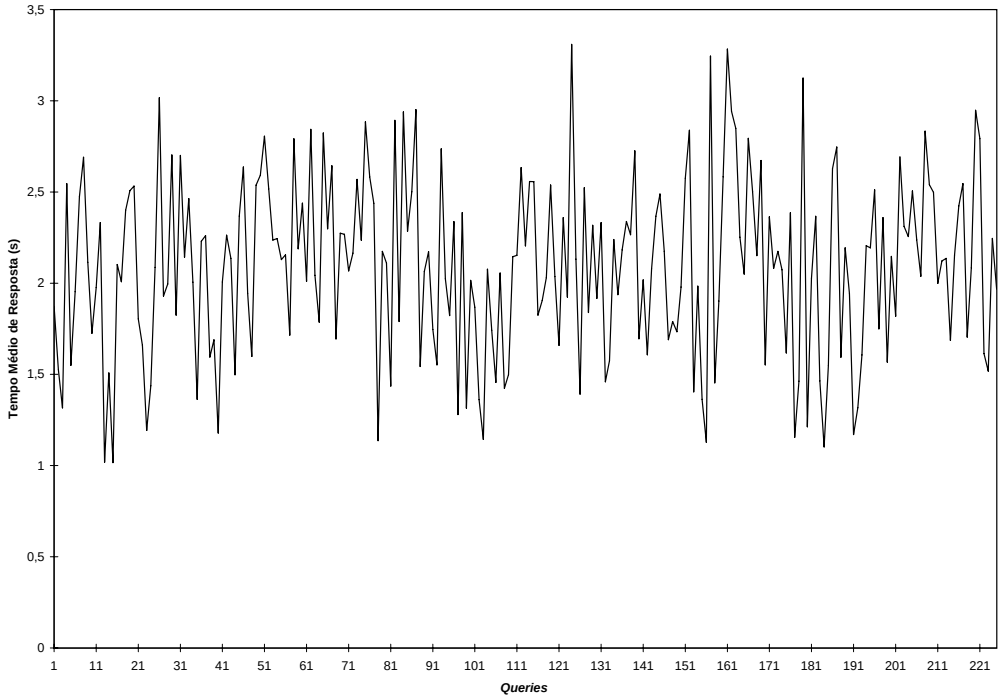

Figura 5.4: Tempos de resposta às interrogações.

médio de todas as interrogações realizadas ao sistema.

Tempo Médio (s)
2,0916

Tabela 5.3: Tempo médio de resposta às interrogações para a colecção *Cranfield* sem aglomeração.

Resta acrescentar, que estes tempos englobam o tempo de propagação das respostas para a entidade que as formula.

5.2.2 Eficácia

Para a avaliação da eficácia do sistema foi utilizado o método de macro-avaliação referido na secção 3.6.5. Os dados resultantes das experiências em conjugação com este método permitiram o traçado das curvas P-T. A figura 5.5 apresenta o resultado para a esta colecção quando não existe qualquer tipo de aglomeração.

Optou-se por apresentar a curva P-T em pontos fixos de totalidade de 0,1 a 0,9 em intervalos de 0,1, uma vez que, não se verificou um consenso nas experiências observadas por outros autores [Sal78, SWY75, AaJPCCL75].

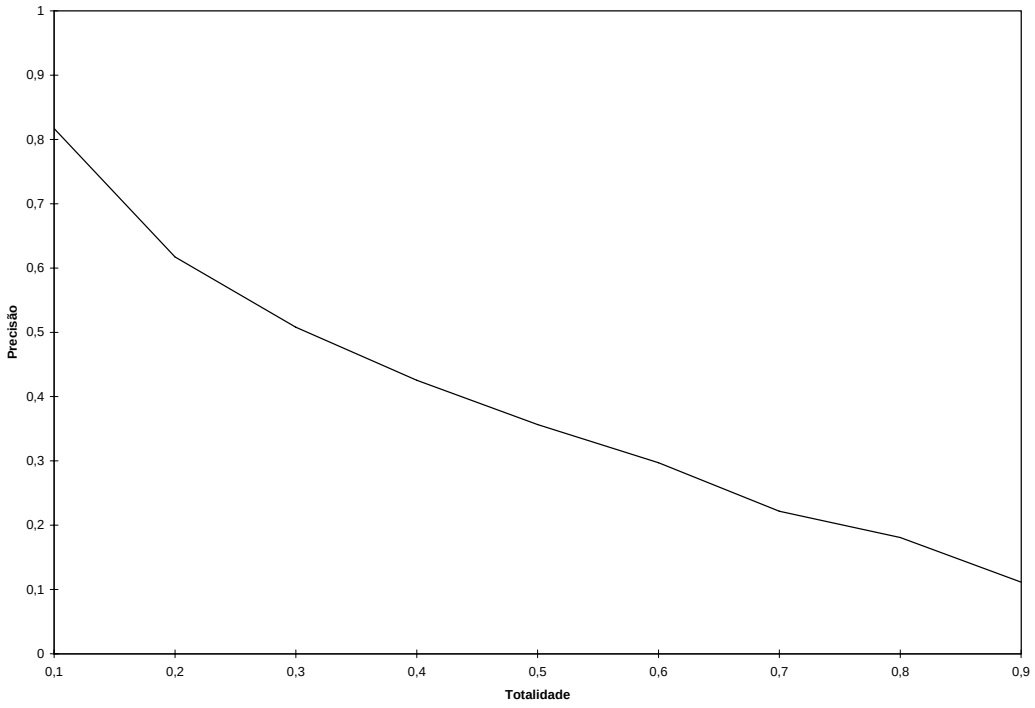

Figura 5.5: Curva P-T para a colecção *Cranfield* sem aglomeração.

Para ser possível uma melhor comparação de desempenho entre as várias experiências foi calculada a média de precisão para os 9 pontos de totalidade. Para este caso temos:

$$\overline{P} = 0,392795$$

5.3 Experimentação com a aglomeração

De seguida irão ser apresentados os resultados obtidos quando se varia o número de aglomerados no sistema. Convém referir que a quantidade de aglomerados no modelo de aglomeração não hierárquico é definida à partida, ou seja, o número mantém-se fixo não havendo a possibilidade de alteração após a indexação, sendo utilizada para a sua definição a entidade `crepmake`.

Neste ponto, os testes serão feitos em modo local, quer isto dizer, que se pretende fixar apenas o efeito da aglomeração sem olhar para pormenores de distribuição. Mais uma vez, apenas entre a `lbot` e a `docman` é que existe uma passagem de documentos remotamente, isto para manter a coerência entre as várias experiências.

Antes de se prosseguir para a descrição dos resultados convém salientar um dos aspectos que mais dificultou o desenvolvimento deste trabalho: a distribuição dos

documentos nos aglomerados depende da ordem pela qual estes são inseridos. O que significa que se poderia obter resultados diferentes se os documentos fossem inseridos por outra ordem. De qualquer forma, e dado que a RI joga no campo das probabilidades, há que segurar em algo palpável que nos permita dar uma ideia do comportamento para uma situação em particular. Mesmo assim, note-se que o método de aglomeração é baseado em pressupostos matemáticos tanto no cálculo do representante dos aglomerados como no cálculo da sua proximidade com os documentos que são inseridos, e por isso, merecem alguma validade.

Como já seria de esperar, a aglomeração de documentos provoca uma ligeira degradação no desempenho do sistema em termos de precisão [vR79]. É, mesmo assim, contando com esta situação, que se pretende analisar quais os custos a perder nesta vertente e quais os benefícios a ganhar em termos de tempo de resposta às interrogações.

A figura 5.6 apresenta a comparação dos tempos de inserção e resposta com a média da precisão.

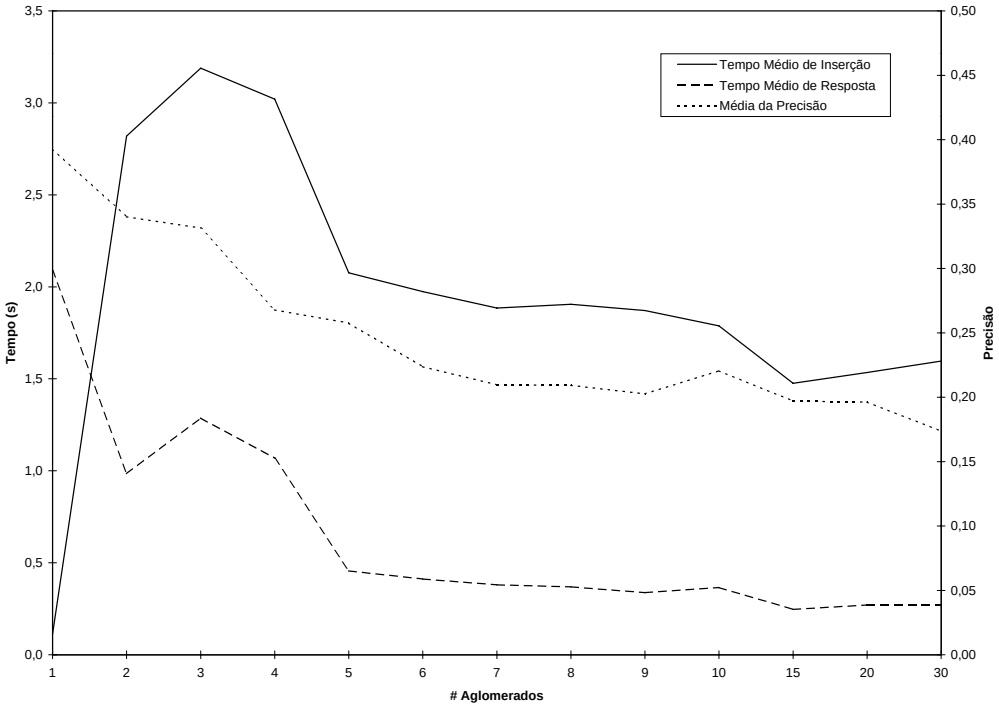

Figura 5.6: Comparação dos desempenhos para variações no número de aglomerados.

Poder-se-ia ainda apresentar as curvas P-T para todas as experiências realizadas variando o número de aglomerados, no entanto, devido às ligeiras variações verificadas, o gráfico tornar-se-ia pouco claro e, por isso, optou-se não o apresentar.

Foram feitos testes com um número restrito de aglomerados (um máximo de 30)

pois verificou-se alguma estabilidade nos tempos de inserção e resposta, a partir de certo ponto (5 aglomerados). A acrescentar ainda que, a linha da média da precisão começa a ficar em valores relativamente baixos e a "potência" computacional não justificaria os resultados em ambiente distribuído. Por outras palavras, não adianta ter 50 máquinas ocupadas senão se obtêm resultados satisfatórios.

A primeira abcissa do gráfico (ponto 1) corresponde ao desempenho do sistema sem aglomeração, ou seja, a experimentação base já analisada na seccão 5.2.

Após uma análise cuidada ao gráfico há que salientar três aspectos fundamentais:

- o aumento drástico do tempo de inserção quando se inicia a aglomeração;
- a queda do tempo de resposta na mesma situação;
- a descida na precisão.

Estes resultados vão de encontro com aquilo que era esperado tanto ao nível dos tempos, como a nível da precisão.

A subida subsequente destes tempos (situação com 3 aglomerados) são justificadas por uma distribuição não uniforme dos documentos, isto é, um dos aglomerados ficou com um número muito elevado de documentos. Este factor foi um dos maiores dilemas do presente trabalho, sendo o seu controle muito difícil de se concretizar sem um conhecimento prévio do conjunto de documentos em questão. Acerca disto falaremos mais à frente. As oscilações que se verificam posteriormente justificam-se também por este factor.

De uma forma geral, as linhas dos tempos tendem a diminuir, assim como a precisão de uma forma mais suavizada por todos os pontos estudados. Por observação do gráfico, conclui-se com alguma facilidade que os pontos mais promissores são a divisão com 2 e com 5 aglomerados. O primeiro devido a uma redução de quase 53% no tempo de resposta e por uma quebra, se bem que um pouco significativa, mas inevitável, na precisão. Em relação ao ponto 5, a nova quebra no tempo de resposta e um valor, ainda, aceitável para a precisão, fazem dele uma boa configuração para esta colecção.

5.4 Controle da distribuição de documentos pelos aglomerados

O modelo de aglomeração não-hierárquica pode ser caracterizado por possuir uma visibilidade muito restrita, ao contrário, dos modelos hierárquicos que entram em linha de conta com todos os documentos e, por isso, extremamente pesados em termos computacionais e mal adequados para aplicações em que existe uma inserção incremental de documentos (ver capítulo 3). Esta falta de visibilidade pode conduzir

a resultados indesejados, assim como, a não distribuição uniforme dos documentos pelos aglomerados. Este facto, verifica-se com a simples alteração do número de aglomerados existentes no sistema. Como este é iniciado a partir do nada, ou seja, não existe uma definição prévia do representante dos aglomerados, a atribuição do primeiro documento a um aglomerado é de extrema importância.

À medida que os documentos vão dando entrada no sistema, são atribuidos ao primeiro aglomerado vazio. Ora, este pressuposto, pode não gerar a distribuição ideal, uma vez que, um documento que vai ser introduzido, quando existam aglomerados vazios, pode ter uma afinidade maior para um aglomerado que já contenha documentos. Com base nisto, foi introduzido um parâmetro que controla a inserção de documentos em aglomerados vazios, designada por limite de bloqueio, este parâmetro, impede que um documento seja atribuído a um aglomerado vazio se houver outros aglomerados (seus representantes) com uma proximidade maior com o documento.

A figura 5.7 apresenta os tempos médios de inserção dos documentos, os tempos médios de resposta às interrogações e a média da precisão para a colecção *Cranfield*, quando é indexada com cinco aglomerados variando o limite de bloqueio entre $0,1$ e 1. Este intervalo foi delimitado entre estes valores devido a que a função de similaridade entre os documentos e os representantes dos aglomerados ser normalizada. O valor 0 não necessita de ser analisado pois o resultado é o mesmo do que não haver aglomeração, ou seja, são todos incluídos num único aglomerado. O valor 1 corresponde à situação em que não se entra em linha de conta com o limite de bloqueio, ou seja, a introdução dos documentos é feita sempre em aglomerados vazios quando existam.

A escolha de 5 aglomerados foi feita com base nos resultados obtidos na secção 5.3, dado que foi considerada uma das melhores opções.

Como se pode observar, apenas para limite de bloqueio igual a $0,1$ é que houve uma deteorização dos tempos, isto devido a que para valores tão baixos deste parâmetro os documentos são inseridos quase exclusivamente em um aglomerado, obtendo-se, assim, resultados muito pobres em termos de tempos, sem melhoramento significativo da precisão.

Para os restantes valores, a primeira conclusão a retirar é um ligeiro incremento na média da precisão nos pontos 0,3 e 0,4, não se verificando descidas muito acentuadas nas medidas apresentadas.

De uma forma genérica podemos afirmar que a utilização do limite de bloqueio com valores não próximos de 0 (zero), não põe em causa o desempenho do sistema, podemos, sim, tirar partido disso. Neste caso em particular, a estabilidade nos factores tempo e precisão, levam a concluir que a colecção ficou bem distribuída sem a utilização do parâmetro em questão, razão pela qual se notam poucas variações nas linhas quando se varia o limite de bloqueio.

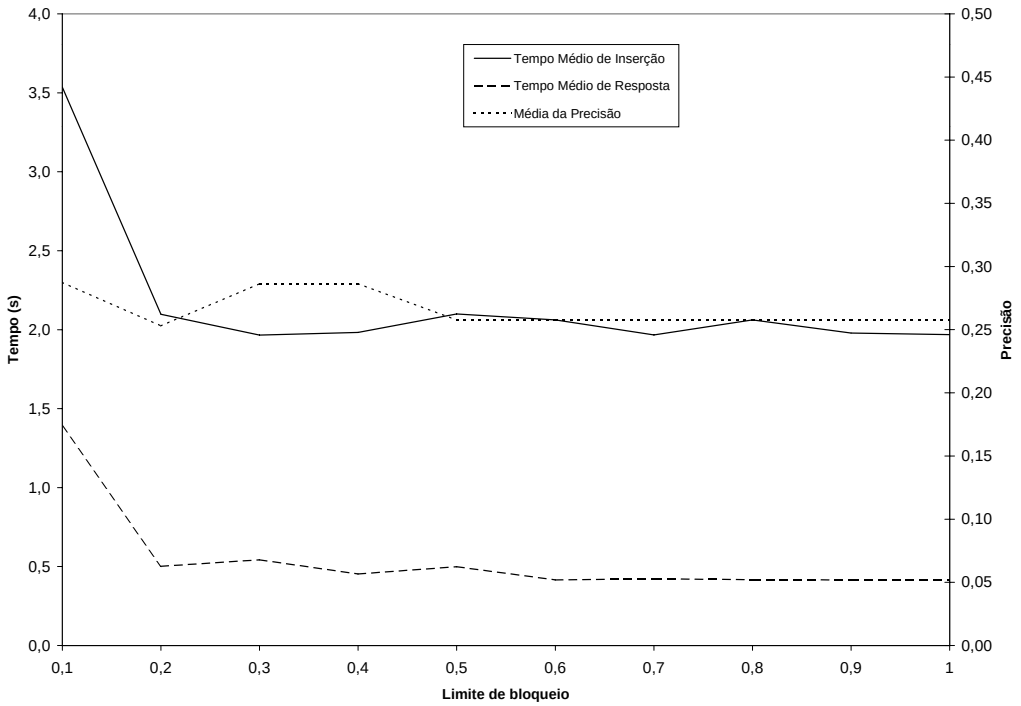

Figura 5.7: Comparação dos desempenhos para variações no limite de bloqueio com 5 aglomerados na colecção *Cranfield*.

Para o confirmar, vejamos o gráfico que apresenta a distribuição dos documentos pelos aglomerados na figura 5.8. Os números que se encontram na legenda representam uma forma de distinguir o aglomerado.

Do gráfico podemos confirmar a concentração de documentos num aglomerado para $0,1$ no limite de bloqueio, assim como, a existência de 2 aglomerados sem documentos. Em 0,2 ainda existe um aglomerado com um número reduzido de documentos. Para 0,3 e 0,4, como já foi visto consegue-se um ligeiro melhoramento na precisão, devido a uma melhor distribuição, certamente, por um aumento do número de documentos nos aglomerados 1 e 2, se comparados com os aglomerados 4 e 5 nos valores do limite de bloqueio superiores ou iguais a 0,5.

Como já seria de esperar, o acerto deste parâmetro não é possível de concretizar sem a experimentação. A entrada do primeiro documento que é inserido num aglomerado e a dependência da ordem de inserção afecta de tal forma os resultados que podemos ter as variações na distribuição dos documentos que a figura 5.8 visualiza.

De forma a se tentar chegar a uma conclusão mais precisa sobre o limite de bloqueio, veja-se na figura 5.9 o mesmo estudo para uma configuração com 4 aglomerados, já que apresenta, sem utilização deste parâmetro uma distribuição pouco uniforme. Repara-se no tempo de inserção que consegue atingir valores mais elava-

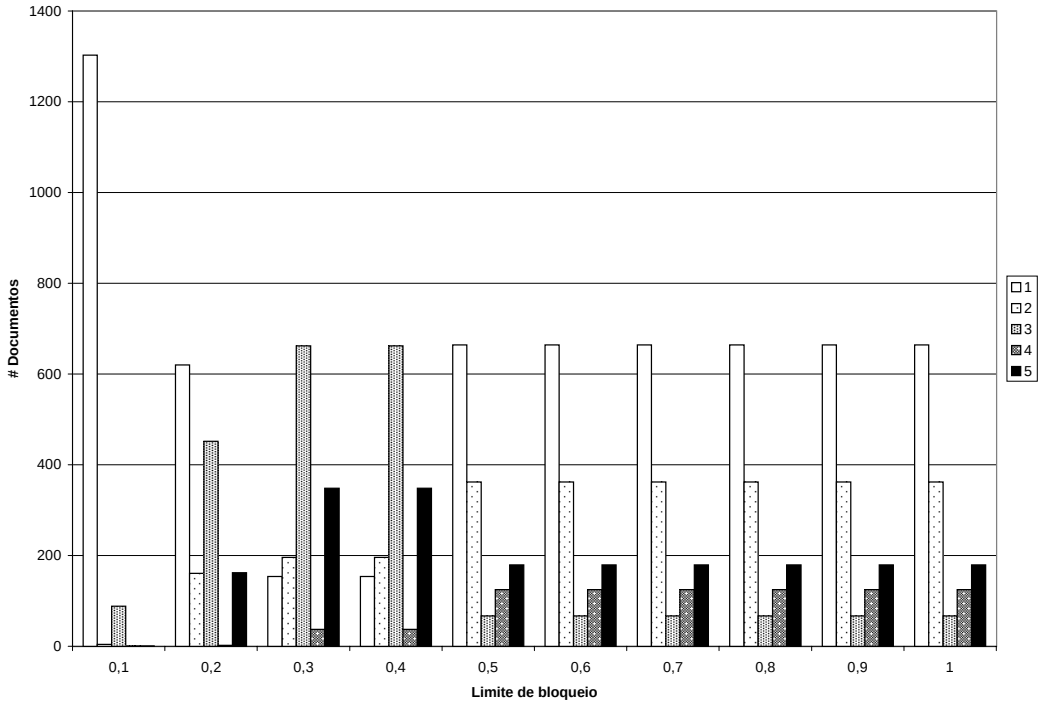

Figura 5.8: Distribuição dos documentos pelos aglomerados para a colecção *Cranfield* com 5 aglomerados.

dos do que uma distribuição para 5 aglomerados (ver figura 5.6).

Mais uma vez, verifica-se que à medida que o valor do limite de bloqueio diminui há um ponto (0,5) em que se obtém um melhoramento nos tempos de inserção e de resposta. A precisão não sofre, no entanto, grande degradação. A partir deste ponto a avaliação torna-se um pouco imprevisível, uma vez que há um aumento na média da precisão no ponto 0,2.

Para complementar estes resultados veja-se a distribuição dos documentos pelos aglomerados na figura 5.10.

Os dois casos apresentados revelam resultados um pouco imprevisíveis a partir de um certo valor do limite de bloqueio, não sendo possível o estabelecimento preciso de uma relação entre o limite de bloqueio e o desempenho do sistema. De qualquer forma, este limite pode ser interpretado como um complemento que, não sendo de fácil ajuste não prejudica significativamente o sistema se não forem usados valores baixos. Pode mesmo, como é o caso dos 4 aglomerados, conseguir-se uma diminuição acentuada dos tempos de inserção e de resposta, o que se manifesta bastante positivo. Esta diferença pode ser mais significativa quando a ditribuição não é muito uniforme sem o limite de bloqueio.

Em tom de finalização deste conjunto de experiências, referira-se que a oscilação

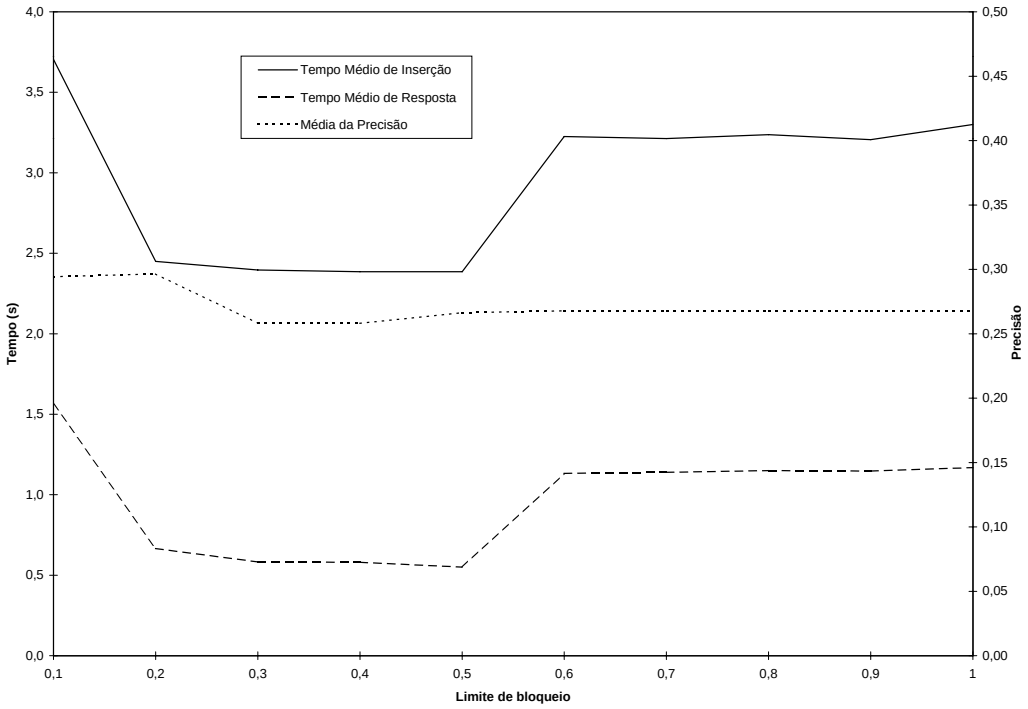

Figura 5.9: Comparação dos desempenhos para variações no limite de bloqueio com 4 aglomerados na colecção *Cranfield*.

do desempenho é provocado pela distribuição dos documentos nos aglomerados. Se um aglomerado alberga muitos documentos, a precisão aumenta, assim como os tempos de inserção e resposta, ficando o sistema menos eficiente.

Por último, para justificar a diferença nos tempos de inserção entre a aglomeração e a MEV simples, veja-se na tabela 5.4, a distribuição percentual média dos tempos de inserção dos documentos pelas operações mais significativas realizadas em cada inserção em relação ao seu tempo total, utilizando a configuração que tem vindo a ser abordada.

	percentagem (%)
Indexação	0,73
Ordenação dos aglomerados	2,10
Geração do representante	46,53
Adição do representante ao ficheiro invertido	48,26
Adição do documento ao ficheiro invertido	2,38

Tabela 5.4: Percentagens da distribuição de tempo pelas operações realizadas na inserção de um documento.

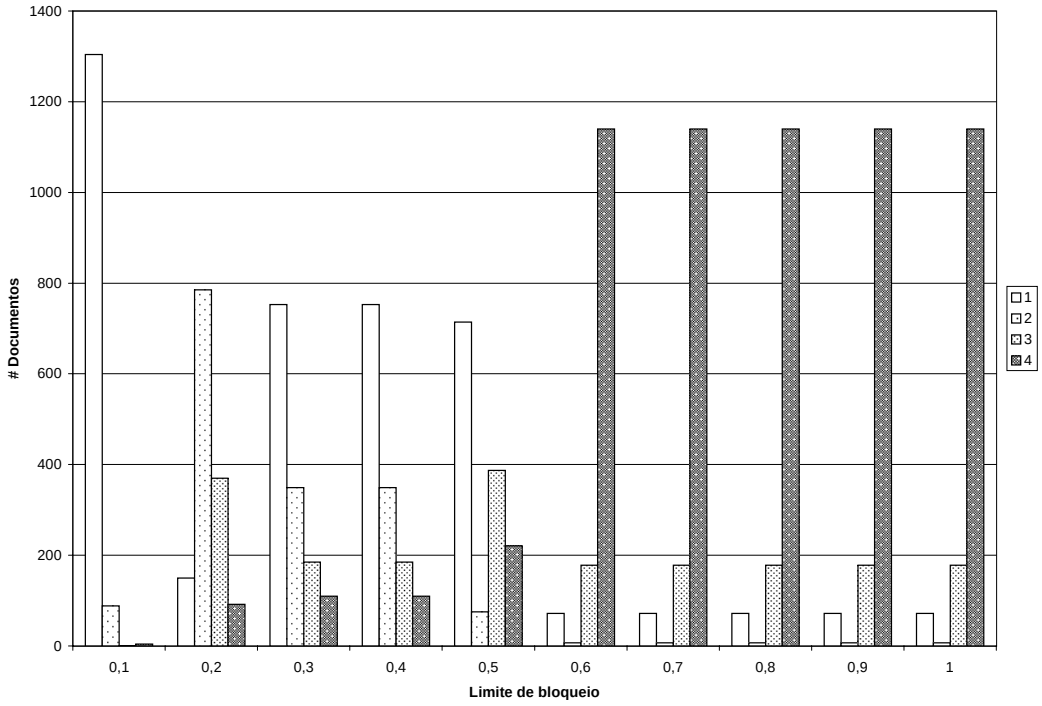

Figura 5.10: Distribuição dos documentos pelos aglomerados para a colecção *Cranfield* com 4 aglomerados.

Como se pode observar os tempos de inserção tão elevados quando comparados com o MEV simples, devem-se maioritariamente à geração do representante e ao seu armazenamento no ficheiro invertido.

5.5 Experimentação da aglomeração distribuída

Depois do estudo feito sobre a aglomeração de documentos, resta verificar o segundo objectivo deste trabalho - a distribuição dos aglomerados.

Um dos maiores problemas confrontados nesta fase foi a criação de condições semelhantes em modo distribuído, em relação ao modo local, de forma a se proceder a uma comparação plausível nos resultados obtidos. A principal preocupação foi em assegurar que este modo, não alterasse a distribuição dos documentos pelos aglomerados, pois, como já foi verificado, pode alterar-se razoavelmente o desempenho consoante a distribuição de documentos pelos aglomerados.

Com base nas considerações referidas e, tendo em conta que o sistema distribuído ideal é aquele que consegue manipular vários documentos ou interrogações de forma a serem processados simultaneamente e, assim, tirar partido do enredo computacio-

nal, foi, no entanto, necessário introduzir os documentos um de cada vez. Como consequência, não serão avaliados os tempos de inserção, pois serão, logicamente, semelhantes, acrescidos dos tempos de propagação pela rede dos documentos e dos aglomerados.

A precisão, tal como se pretendia, foi semelhante em relação ao modo local para as experiências realizadas.

É em relação às interrogações, que o modo distribuído beneficiou o sistema. Para o comprovar realizou-se um conjunto de quatro experiências, utilizando as interrogações que acompanham o pacote da colecção *Cranfield*:

1. Uma sequência de interrogações realizadas uma de cada vez;
2. Três sequências das mesmas interrogações, realizadas simultaneamente.
3. Cinco sequências das mesmas interrogações, realizadas simultaneamente;
4. Seis sequências das mesmas interrogações, realizadas simultaneamente;
5. Seis sequências das mesmas interrogações, realizadas simultaneamente, sendo três feitas pela ordem inversa, intercaladamente.

A experiência 1 tem como objectivo pôr em evidência o facto de um sistema distribuído não ter vantagens, se não houver tarefas simultâneas. A experiência 2, 3 e 4 para mostrar a evolução com o aumento da carga de interrogações e, por último, a experiência 4, para acentuar a diferença entre os dois modos, quando são feitas interrogações diferentes simultaneamente.

Em termos de configuração, criou-se uma distribuição por 4 aglomerados, com o limite de bloqueio igual a 0,5. As decisões para o primeiro parâmetro estão relacionadas com as limitações ao nível do equipamento informático disponível. Para o limite de bloqueio, achou-se interessante colocar em evidência a sua utilização, uma vez que se obteve um bom comportamento em modo local.

Com base nisto, traçou-se o gráfico apresentado na figura 5.11.

A análise dos resultados, mostra que, com a experiência 1, de facto, se não existir processamento concorrente o desempenho em modo distribuído agrava-se devido à propagação das respostas. Refira-se que, em modo distribuído, uma resposta antes de ser encaminhada para o utilizador (entidade `query`), passa pela entidade `docman` à qual foi feita a interrogação. Naturalmente, esta decisão prejudica o tempo de resposta total, mas foi feita precisamente para colocar em evidência o tempo dispendido na propagação.

Nas restantes experiências, nota-se, claramente, um avanço do desempenho do modo distribuído, sendo mais significativo para a experiência 5, já que existe uma

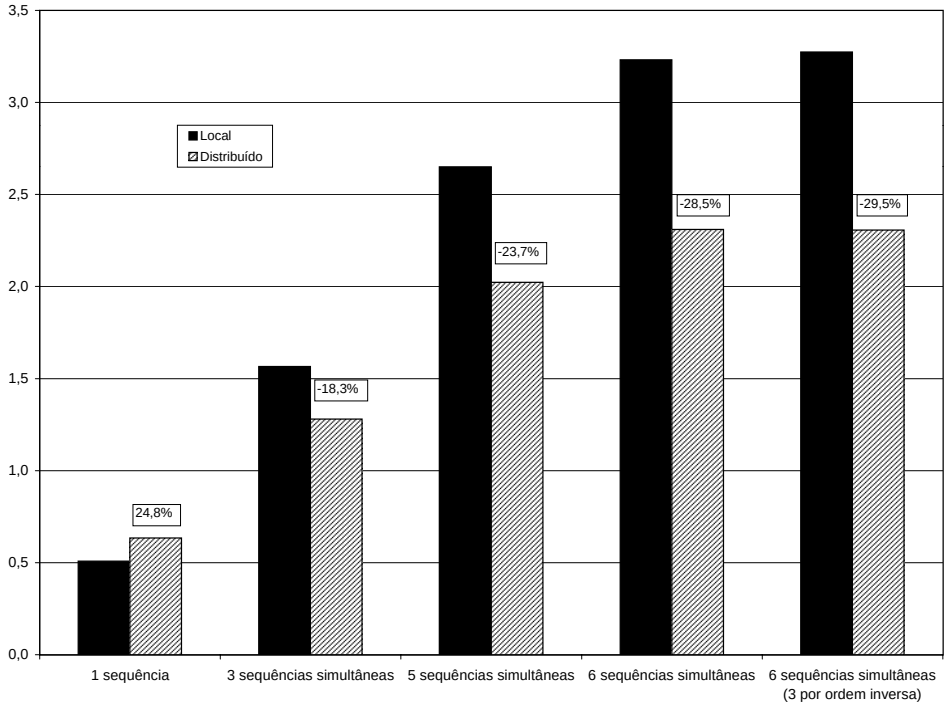

Figura 5.11: Variação de tempos de resposta em modo local e distribuído com 5 naipes de interrogações para a colecção *Cranfield* com 4 aglomerados e limite de bloqueio igual a 0,5.

maior aleatoriedade na distribuição das interrogações pelos servidores que contêm aglomerados.

A distribuição do espaço ocupado por cada aglomerado é, logicamente, dependente da distribuição dos documentos pelos aglomerados, acrescido do espaço ocupado pelos representantes dos restantes aglomerados. Este acréscimo provoca, em termos gerais, um aumento de 205,71% em relação ao espaço ocupado por um sistema sem aglomeração, mas consegue-se criar em cada máquina a distribuição de espaço total apresentada na figura 5.12.

Estes valores percentuais são de uma configuração de 4 aglomerados distribuídos para a colecção *Cranfield* com limite de bloqueio igual a 0,5, e são relativos ao espaço total ocupado pelos documentos, incluindo o espaço dos representantes.

Se se comparar o espaço ocupado em cada máquina com o espaço ocupado pelo sistema sem aglomeração, obtem-se a distribuição representada na figura 5.13.

O que indica claramente que apesar da duplicação do espaço em termos gerais, cada máquina fica menos sobrecarregada quando comparada com o sistema centralizado.

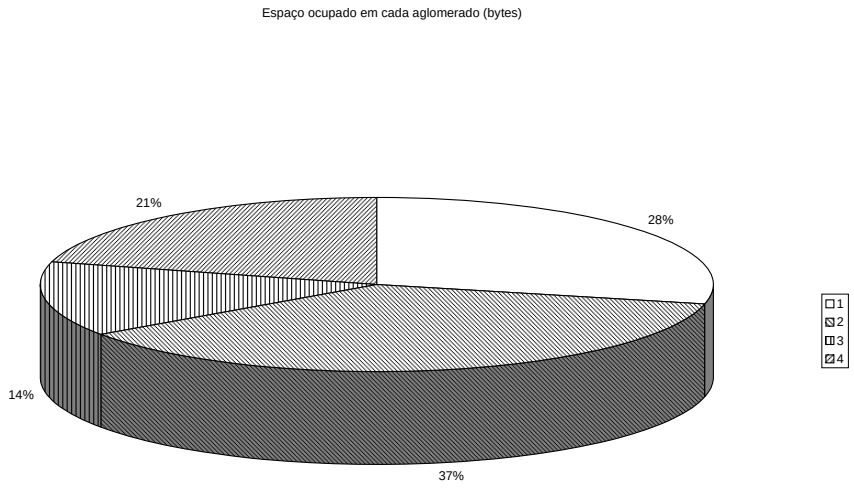

Figura 5.12: Percentagens de distribuição do espaço ocupado nas máquinas para a colecção *Cranfield* com 4 aglomerados e limite de bloqueio igual a 0,5.

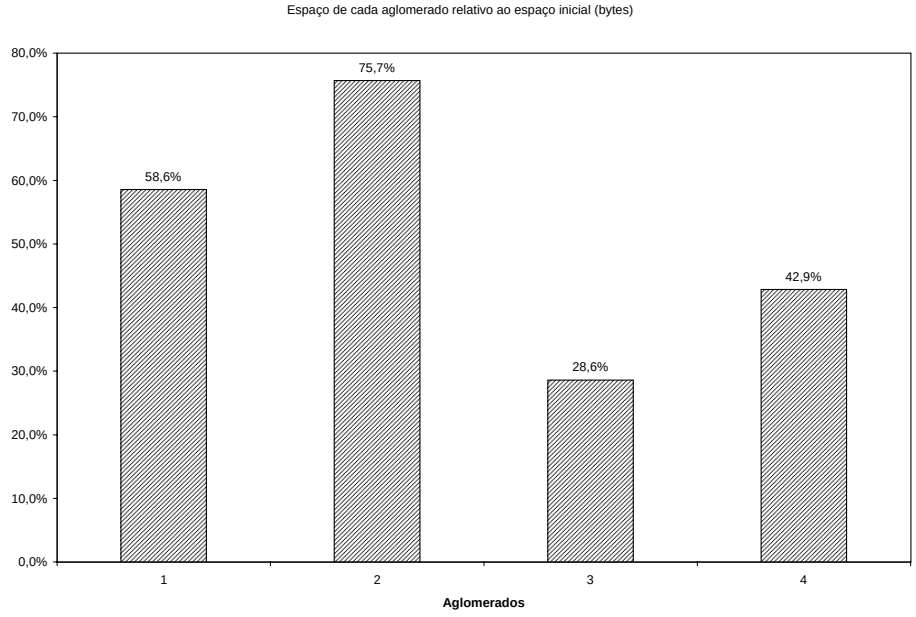

Figura 5.13: Percentagens de distribuição do espaço ocupado nas máquinas para a colecção *Cranfield* com 4 aglomerados e limite de bloqueio igual a 0,5 comparado com o espaço ocupado sem aglomeração.

Capítulo 6

Conclusões e trabalho futuro

6.1 Conclusões

A criação de uma plataforma de trabalho que permita a análise e o desenvolvimento de técnicas da RI utilizando um modelo de aglomeração não hierárquico, assumiu-se como objectivo primário, recorde-se, desta dissertação (ver secção 1.5). Uma vez que o processo de aglomeração associa documentos potencialmente relevantes para as mesmas interrogações no mesmo aglomerado, então, a informação que o sistema necessita para responder eficazmente a uma interrogação, depois de encaminhada para um aglomerado, encontrar-se-ia exclusivamente nesse aglomerado. Esta ideia, sugeriu que se efectuasse uma distribuição dos aglomerados por um conjunto de máquinas interligadas em rede, permitindo uma distribuição da carga computacional e do espaço ocupado pelos documentos.

6.1.1 SInAD

A criação de uma plataforma própria de RI, permitiu comprovar o que dela se esperava. Apesar do tempo consumido na implementação ter sido maior do que o previsto, a comparação de resultados ficou facilitada e uniformizada, tendo sido possível um isolamento bastante satisfatório dos parâmetros que se pretendiam analisar. Embora, desprezando aspectos importantes da teoria relacionada com a RI, tais como a indexação, obtiveram-se resultados bastante satisfatórios para a colecção testada com a utilização do MEV simples.

A implementação que se fez do SInAD ficou áquem das expectativas de se tornar numa aplicação de uso genérico. Se bem que este se encontre preparado para o funcionamento com casos reais, não ficou concluída a preparação de um interface versátil que permita a configuração fácil do sistema por um utilizador comum. Em relação à aplicação do sistema com a `WWW`, esta foi de certa forma boicotada, devido a

problemas no software de navegação (`libwww` [lib97]), que não foi possível de colocar em funcionamento pleno.

Um dos aspectos que mais entraves colocou à obtenção de um melhor desempenho do SInAD foi, sem dúvida, a utilização da biblioteca `libdb`, que não suporta a manipulação concorrente de dados. A utilização de mecanismos de exclusão mútua baseados nos tradicionais trincos como garante de uma leitura e escrita de dados coerente, prejudicou a eficiência do sistema.

Por outro lado, e infelizmente, o suporte à programação baseada em fios de execução[1] não estava suficientemente estabilizado em Linux, aquando do desenvolvimentodo SInAD, o que poderia ter dado um desempenho na execução mais satisfatório.

Adicionalmente, as máquinas utilizadas nas experiências não primam pela velocidade, sendo o seu desempenho facilmente afectado por outras aplicações. Se bem que se estabeleceu uma sequência de experiências que permite uma comparação relativa de tempos, estes não devem ser tomados como indicadores definitivos.

6.1.2 Modelo de Aglomeração

As características do modelo de aglomeração não hierárquico fazem dele uma técnica muito imprevisível. A dependência da ordem não permite que sejam feitas generalizações muito coerentes.

Verificou-se que não compensava uma divisão da colecção em muitos aglomerados, uma vez que, para além da tendência decrescente da precisão, existe uma relativa estabilização dos tempos de inserção e resposta.

Convém salientar que o algoritmo de aglomeração foi adoptado fielmente, isto é, a única informação que cada aglomerado tem acerca dos outros é o representante, o que influencia seriamente a inserção dos documentos, podendo ser criadas distribuições de documentos não uniformes pelos aglomerados. A inclusão do limite de bloqueio, veio, no entanto, contribuir para atenuação das diferenças no número de documentos em cada aglomerado, não prejudicando significativamente o desempenho do sistema.

A dependência da ordem pela qual os documentos são inseridos, dificultou a avaliação do sistema, uma vez que, é suficiente uma alteração num parâmetro para criar distribuições diferentes e, consequentemente, desempenhos diferentes o que impossibilitava uma comparação coerente de resultados para as mesmas condições de parâmetros.

Uma crítica aplicável à presente implementação do SInAD é a de que os tempos médios de inserção de documentos podem atingir valores bastante superiores aos do MEV simples, na ordem dos 2 segundos. Este tempo é maioritariamente gasto (ver

[1] Do inglês *threads*.

tabela 5.4):

1. no cálculo do representante;
2. na escrita do representante no ficheiro invertido.

O ponto 2 pode ser de alguma forma optimizado utilizando uma técnica de redução no número de termos do representante com base no estabelecimento de um limite de inclusão para o peso de cada termo. Este parâmetro cria, no entanto, novas distribuições dos documentos pelos aglomerados, originando variações nas medidas de desempenho, se bem que ligeiras, mas oscilatórias para uma variação linear desse parâmetro, o que torna a comparação de resultados impraticável.

O ponto 1 é uma consequência já esperada e depende da qualidade na distribuição dos documentos pelos aglomerados. Quanto melhor for a distribuição, isto é, quanto mais equilibrada, menor será o tempo de inserção de novos documentos, devido ao cálculo do representante, que necessita de percorrer todos os documentos.

Um aspecto que poderia de alguma forma melhorar estes tempos seria executar o cálculo do representante apenas entre um intervalo de documentos introduzidos. Apesar de não ter sido verificado este factor o seu contributo seria por certo satisfatório, se se tivesse em conta um período de estabilização do aglomerado, ou seja, numa fase inicial em que o aglomerado tenha poucos documentos, este processo não deveria ser colocado em prática, já o representante não reflectiria de alguma forma o conteúdo do seu aglomerado.

6.1.3 Distribuição

É precisamente com a distribuição dos aglomerados por várias máquinas que se tentou dar algum alento ao modelo de aglomeração. Não foram apresentados resultados mais extensos acerca deste factor porque houve uma maior preocupação na contabilização individual dos tempos de inserção de cada documento e não no cômputo geral de todos os documentos. Naturalmente, o modo distribuído pode processar um número máximo de documentos igual ao número de máquinas que cooperam, simultaneamente, no sistema, simultanemente, o que garante um aumento de eficiência.

Ao nível dos tempos de resposta conseguiram-se decréscimos na ordem dos 30% relativamente à versão centralizada (ver figura 5.11). Ficou também comprovada a existência óbvia de um aumento no tempo de resposta devido aos atrasos dos representantes na rede, quando as interrogações são processamente sequencialmente, havendo uma compensação no tempo quando são processadas várias interrogações ao mesmo tempo.

A optimização dos tempo de propagação, todavia, foi ainda ponderada utilizando

o limite de inclusão de termos do representante mas surgiam problemas na avaliação como já foi referido.

De uma forma geral pode-se afirmar que a distribuição veio garantir os objectivos propostos ao nível da eficiência e ao nível da disponibilidade de recursos. O processamento concorrente de documentos e interrogações permitiu um aumento de desempenho nos processos de inserção de documentos e resposta às interrogações. Por outro lado, conseguiu-se uma distribuição efectiva do espaço ocupado pelas bases de dados. Embora exista um aumento bastante significativo do espaço global ocupado, ele é distribuído pelas máquinas existente no sistema. Este incremento para quase o dobro não é, de forma alguma, satisfatório, mas tendo em conta que não foi feita qualquer tentativa de redução dos termos dos representantes é, no mínimo, previsível. A existência de mais do que uma máquina envolvida no sistema permitiu um aumento da disponibilidade global do mesmo, uma vez que cada servidor contém a informação de todos os representantes e por isso é capaz de encaminhar correctamente uma interrogação.

Resumindo, poderemos afirmar que o modelo de aglomeração não faz sentido se não houver uma distribuição efectiva dos recursos utilizados pelo sistema. O método de aglomeração não hierárquico sofre de lacunas de visibilidade global sobre toda a colecção de documentos, podendo ser geradas distribuições não uniformes de documentos pelos aglomerados. O limite de bloqueio auxiliou a resolução deste problema, beneficiando assim o tempo de inserção de documentos. Conseguiu-se, de um modo geral, optimizar o tempo de resposta às interrogações, sem prejudicar em demasia a eficácia do sistema. Comparando uma configuração de 5 aglomerados com limite de bloqueio igual a 0,5 com uma configuração sem aglomeração houve uma redução de 32% na média da precisão.

O SInAD permite a recolha de recursos de informação tal como o motor de pesquisa convencional, mas ao contrário destes, distribui as bases de dados por um conjunto de máquinas, aumentando, assim, a disponibilidade do sistema e o poder computacional global no tratamento de interrogações em simultâneo. A divisão dos documentos pelos aglomerados permite a identificação da noção de *classificação*, uma vez que, segundo a *hipótese do aglomerado* (secção 3.5), os documentos associados terão uma tendência a ser relevantes para a mesma interrogação, sendo desta forma devolvidos ao utilizador os documentos que mais se relacionam entre si.

Todo este cenário, apesar das desvantagens apontadas, apoia a delegação de resumos da informação em ambientes distribuídos para entidades que interagem com o utilizador que formula as interrogações, tal como o fazem os representantes dos aglomerados aqui referidos. Esta ideia está bem patente nos serviços de directoria, onde há passagem de testemunhos de informação de uns servidores para outros, como por exemplo o `whois++` [WFS95, DFW95, FSW95] que acumula a informação nos *centroids* ou o HARVEST [BDH+95] nas estruturas `SOIF`. O modelo aqui apresentado poderá dar uma nova visão na forma em como os recursos são distribuídos na rede

de servidores, utilizando uma métrica de próximidade de recursos e não apenas por opção de quem gere o sistema.

6.2 Trabalho futuro

É possível estabelecer algum trabalho futuro baseado nas conclusões retiradas e nos problemas debatidos.

Em primeiro lugar, nunca são de menosprezar os efeitos positivos que uma boa indexação pode provocar no desempenho do sistema. Seria, portanto, um ponto a aperfeiçoar, uma vez que não houve grande preocupação relativamente a este aspecto.

Em termos de implementação, seria necessário resolver os problemas que resultaram da utilização da biblioteca `libdb` de forma a incrementar o seu desempenho nos acessos concorrentes.

Para se conseguir tirar melhor partido de uma boa distribuição pelas máquinas é necessário que o método de aglomeração não hierárquico seja mais refinado. A passagem de outro tipo de informação entre aglomerados, para além do representante, é uma possibilidade a explorar, de forma a se conseguir obter uma panorâmica mais global do sistema. Informação de qualidade de serviço é também uma informação que poderá tornar-se útil.

Possivelmente, a utilização de um método híbrido, que conjugue o método hierárquico sem trazer consigo as desvantagens dos elevados tempos no cálculo da matriz de similaridade, trará alguns progressos a este nível.

Bibliografia

[AaJPCCL75] James Allan, Lisa Ballesteros ad James P. Callan, W. Bruce Croft, and Zhihong Lu. Recent experiments with inquery. *Fourth TREC Retrieval Conference (TREC-4)*, pages 613–620, 1975.

[BC87] N. J. Belkin and W. B. Croft. Retrieval techniques. In M.E. Williams, editor, *Annual Review of Information Science and Technology*, pages 109–145. Elsevier Science Publishers, 1987.

[BCC94] John Broglio, James P. Callan, and W. Bruce Croft. Inquery system overview. Technical report, University of Massachusetts, 1994.

[BDH+95] C Mic. Bowman, Peter B. Danzig, Darren R. Hardy, Udi Manber, Michael F. Schwartz, and Duane P. Wessels. Harvest: A scalable, customizable discovery and access system. Technical report, Department of Computer Science, Universidade de Colorado, 1995.

[BLC95] T. Berners-Lee and D. Connolly. Hypertext Markup Language 2.0, Novembro 1995. RFC 1866.

[BLCL+94] T. Berners-Lee, R. Cailliau, A. Luotonen, H. Frystyk Nielsen, and A. Secret. The world-wide web. *Communication of the ACM*, 1994.

[BLMM94] Tim Berners-Lee, L. Masinter, and M. McCahill. Uniform Resource Locators (URL), Dezembro 1994. RFC 1738.

[Buc85] Chris Buckley. Implementation of the smart information retrieval system. Technical report, Universidade de Cornell, 1985.

[Bur95] Robert Burgin. The retrieval effectiveness of five clustering algorithms as a function of indexing exhaustivity. *Journal of the American Society for Information Science*, 46(8):562–572, 1995.

[CCH92] J. P. Callan, W. B. Croft, and S. M. Harding. The inquery retrieval system. In *Third Internacional Conference on Database and Expert Systems Applications*, pages 78–83. Valencia, Spain, 1992.

[CM95] Brendon Cahoon and Kathryn McKinley. Performance Analysis of Distributed Information Retrieval Architectures. Technical report, Department of Computer Science, University of Massachusetts, Amherst, MA 01003, USA, Junho 1995.

[CP92] Doug Cutting and Jan Pedersen. Optimizations for dynamic inverted index maintenance. In *13th International Conference on Research and Development in Information Retrieval*, pages 318–329, 1992.

[Cro95] W. Bruce Croft. What do people want from information retrieval. *D-Lib Magazine*, Novembro 1995. URL:http://www.dlib.org/dlib/november95/11croft.html.

[DFM95] Leslie L. Daigle, Patrik Faltstrom, and Michael Mealling. Uniform resource names, iso oids and dns, Novembro 1995. Internet Draft.

[DFW95] Peter Deutsch, Patrick Faltstrom, and Cris Weider. Architecture of the WHOIS++ service, Janeiro 1995. Internet Draft.

[FB93] Norbert Fuhr and Chris Buckley. Optimizing document indexing and search term weighting based on probabilistic models. In *The First Text Retrieval Conference (TREC-1)*, pages 89–100. NIST Special Publication, 1993.

[FBY92] William B. Frakes and Ricardo Baeza-Yates. *Information Retrieval - Data Structures & Algorithms*. Prentice Hall, 1992.

[FGM+97] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, and T. Berners-Lee. Hypertext Transfer Protocol - HTTP/1.1, Janeiro 1997. RFC 2068.

[FSW95] P. Faltstrom, R. Schoultz, and C. Weider. How to interact with a WHOIS++ mesh, Março 1995. Internet Draft.

[GOP90] Keith E. Gorlen, Sanford M. Orlow, and Perry S. Plexico. *Data Abstraction and Object-Oriented Programming in C++*. John Wiley & Sons, 1990.

[GT94] Gregory Grefenstette and Pasi Tapanainen. What is a Word, What is a sentence? Problems of Tokenization. Technical report, Rank Xerox Research Centre, Grenoble Laboratory, 38240 Meylan, France, Abril 1994.

[HG96] David A. Hull and Gregory Grefenstette. A Detailed Analysis of English Stemming Algorithms. Technical report, Rank Xerox research Centre, 6 chemim de Maupertuis, 38240 Meylan France, Janeiro 1996.

[HK95] Martin Hamilton and Jon Knight. A simple discovery protocol. Technical report, Universidade de Loughborough, Março 1995.

[Hul94] David A. Hull. *Information Retrieval Using Statistical Classification.* PhD thesis, Stanford University, Novembro 1994.

[JC93] Yufeng Jing and W. Bruce Croft. An association thesaurus for information retrieval. Technical report, University of Massachusetts at Amherst, 1993.

[JRSW95] Gareth Jones, Alexander M. Robertson, Chawchat Santimetvirul, and Peter Willett. Non-hierarchic document clustering using a genetic algorithm. Technical report, Universidade de Sheffield, 1995.

[Kee92] E. Michael Keen. Presenting results of experimental retrieval comparisons. *Information Processing & Management*, 28(4):491–502, 1992.

[Lam86] Leslie Lamport. *LaTeX- A Document Preparation System.* Addison-Wesley Publishing, 1986.

[lib97] Libwww - the w3c sample code library, 1997. URL:http://www.w3.org/Library/.

[Lip91] Stanley B. Lippman. *C++ Primer.* Addison-Wesley Publishing, 1991.

[Lip96] Stanley B. Lippman. *Inside the C++ object model.* Addison Wesley Publishing, 1996.

[Luh58] H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, pages 159–165, 1958.

[Man97] R. Manmatha. Multimedia indexing and retrieval at the center for intelligent information retrieval. In *Symposium on Document Image Understanding Technology*, 1997.

[MBK91] Yoëlle S. Maarek, Daniel M. Berry, and Gail Kaiser. An information retrieval approach for automatically constructing software libraries. In *IEEE Transactions on Software Engineering*, volume 17, pages 800–813. IEEE, Agosto 1991.

[MW93] Udi Manber and Sun Wu. Glimpse: A tool to search through entire file systems. Technical report, Universidade de Arizona, 1993.

[Por80] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[RDM95] Jr. Ron Daniel and Michael Mealling. URC Scenarios and Requirements, Março 1995. Internet Draft.

[RL94] Ellen Riloff and Wendy Lehnert. Information Extraction as a basis for High-Precision Text Classification. In ACM, editor, *ACM Transactions on Information Systems*, volume 12, pages 296–333. Department of Computer Science, University of Massachusetts, Amherst, MA 01003, July 1994.

[RWHB+93] S. E. Robertson, S. Walker, M. Hancock-Beaulieu, A. Gull, and M. Lau. Okapi at trec. In *The First Text REtrieval Conference (TREC-1)*. NIST, 1993.

[Sal71] G. Salton. The smart retrieval system. Technical report, Universidade de Cornell, 1971.

[Sal78] Gerard Salton. Mathematics and information retrieval. Technical report, Universidade de Cornell, 1978.

[SB88] Gerard Salton and Chris Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[Slo97] Edward G. Slottow. Engineering a global resolution service. Master's thesis, Mit, Junho 1997.

[SM94] K. Sollins and L. Masinter. Functional Requirements for Uniform Resource Names, Dezembro 1994. RFC 1737.

[Sol97] Karen R. Sollins. Architectural principles of uniform resource name resolution, Setembro 1997. Internet Draft.

[Str91] Bjarne Stroustrup. *The C++ Programming Language*. Addison-Wesley Publishing, 1991.

[SWY75] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *ACM Communications*, 18(11):613–620, 1975.

[Tan92] Andrew S. Tanenbaum. *Modern Operating Systems*. Prentice Hall, 1992.

[Tan96] Andrew S. Tanenbaum. *Computer Networks*. Prentice-Hall International, 3rd edition, 1996.

[VF95] Charles L. Viles and James C. French. Dissemination of collection wide information in a distributed information retrieval system. In *Proceedings of the 18th International Conference on Research and Development in Information Retrieval*, 1995.

[Voo86] Ellen M. Voorhees. Implementing agglomerative hierarquic clustering algorithms for use in document retrieval. Technical report, Universidade de Cornell, Julho 1986.

[vR79] C.J. van Rijsbergen. *Information Retrieval*. Butterworths, 2nd edition, 1979.

[WD94] Chris Weider and Peter Deutsch. A Vision of an Integrated Internet Information Service, Dezembro 1994. RFC 1727.

[WFS95] Chris Weider, Jim Fullton, and Simon Spero. Architecture of the WHOIS++ Index Service, Junho 1995. Internet Draft.

[Wiz97] Network Wizards. Intenet domain survey, Julho 1997. URL:http://www.nw.com/.

[WM92] Sun Wu and Udi Manber. Agrep - a fast approximate pattern-matching tool. Technical report, Universidade de Arizona, 1992.

[WS79] Harry Wu and Gerard Salton. A comparison of search term weighting: Term relevance vs. inverse document frequency. Technical report, Universidade de Cornell, 1979.