Os efeitos do paralelismo e relações de *thesaurus* em uma ferramenta de busca em bases textuais

Renan Gomes Pereira[1]
Maria Fernanda Moura[2]
Luis Eduardo Gonzales[3]

Resumo: O objetivo deste trabalho é apresentar as funcionalidades de paralelismo e busca utilizando relações de *thesaurus* da IPreproc, que é uma ferramenta eficaz para realizar buscas em bases textuais. A ferramenta tem sido utilizada pelo software Compilação e Recuperação de Informação Técnico-científica e Indução ao Conhecimento (CRITIC@) para a realização da indexação incremental, análise e busca de documentos. Observou-se uma diminuição do tempo de execução utilizando paralelismo na ferramenta e um aumento do número de resultados retornados por uma consulta com o uso das relações *thesaurus*.

Palavras-chave: Apache Lucene, mineração de textos, máquina de busca, expansão de busca.

[1] Estudante de Engenharia de Computação da Universidade Estadual de Campinas (Unicamp), estagiário da Embrapa Informática Agropecuária, Campinas, SP.

[2] Estatística, doutora em Ciências Matemáticas e da Computação, pesquisadora da Embrapa Informática Agropecuária, Campinas, SP.

[3] Engenheiro da Computação, analista da Embrapa Informática Agropecuária, Campinas, SP.

Introdução

A IPreproc é uma ferramenta em Java, utilizando a biblioteca *open source* Apache Lucene 6.1.0 (THE APACHE SOFTWARE FOUNDATION, 2016). que realiza o pré-processamento de um conjunto de documentos de acordo com os parâmetros incluídos no seu arquivo de configuração (arquivo texto com marcadores XML). Ela foi desenvolvida para atender diversas necessidades de projetos de mineração de textos. Durante sua fase de desenvolvimento, buscou-se criar uma ferramenta bastante flexível e modular, de modo a ser facilmente expansível com a adição de novas funcionalidades. A ferramenta possui 3 módulos principais: *Indexer, Extractor e Searcher*. Esses módulos são responsáveis respectivamente pela indexação incremental de documentos, extração de Matrizes atributo-valor para suportar os processos de mineração de textos e busca. Uma das adaptações da ferramenta está incorporada ao software CRITIC@(MOURA et al., 2015).

Na ferramenta é possível definir uma série de filtros para busca e/ou mineração de textos para subconjuntos de *tags* de uma base de textos, composta por arquivos no formato XML. Deste modo, o usuário pode tratar os campos de dados dos arquivos de forma diferente. Por exemplo, nas *tags* resumo, título e descrição, pode-se utilizar o filtro de indexar apenas as palavras presentes em um arquivo de vocabulário controlado com termos sobre agricultura, ignorando os termos restantes. Nas demais *tags*, esse filtro pode ser desligado. Se a ferramenta não possuísse essa flexibilidade de escolha de quais *tags* utilizam quais filtros, ao indexar o campo de Autor, todos os nomes de autores que não estivessem no vocabulário seriam perdidos e não seria possível retornar esses autores nas consultas. Porém, também é interessante poder utilizar algum tipo de filtragem no campo de Autor. Por falta de padronização, muitas vezes os nomes dos autores dos documentos aparecem diferentes e uma simples filtragem, por exemplo não fazer distinção entre letras maiúsculas e minúsculas, pode aumentar consideravelmente a fração de documentos relevantes retornados de uma busca.

Para aumentar a performance da indexação incremental de documentos e da extração de Matrizes atributo-valor, as operações no índice são feitas de forma paralela, podendo utilizar todos os núcleos disponíveis na máquina em que a ferramenta está sendo executada. A IPreproc cria diversas *threads* que realizam as operações paralelamente no índice. Infelizmente o ganho de desempenho do processo de indexação é limitado pelo uso intensivo

de leitura do disco. Para a extração da Matriz-atributo valor a vantagem da paralelização é muito mais expressiva.

O módulo de busca possui a opção para utilizar um *thesaurus* para expandir um termo de busca e buscar por termos semelhante. A ferramenta lê um arquivo XML com as relações dos termos no *thesaurus*. Essas relações podem ser de related terms (termos relacionados), *narrower terms* (termos específicos) ou *broader terms* (termos mais abrangentes). Durante uma busca, se o termo inserido pelo usuário combina com alguma das relações, essa relação é inserida no termo de busca com um peso dependendo do tipo de relação. O peso de cada tipo de relação é escolhido pelo usuário no arquivo de configuração da IPreproc. Muitos dicionários *thesaurus* também apresentam informações sobre os sinônimos dos arquivos. Na IPreproc os sinônimos de um termo devem ser inseridos na fase de indexação dos documentos utilizando o filtro de sinônimos, também especificável no arquivo de configuração. Com isso a busca por sinônimos é muito mais eficiente do que se esse processo fosse feito durante a fase de expansão da busca.

Este trabalho apresenta os ganhos de performance com o paralelismo e exemplos de buscas utilizando a expansão de busca por *thesaurus*.

Materiais e Métodos

A metodologia empregada foi dividida em duas fases: i) Paralelismo; ii) buscas com thesaurus. Ambas as fases foram realizadas em uma máquina Intel® Core™ i7-4702MQ CPU @ 2.20GHz × 8 rodando no Ubuntu 16.04 64-bit, com 8 GB RAM.

i) Paralelismo: para testar a indexação, foi indexada a base de metadados do repositório Alice (EMBRAPA, 2016) no formato Open Archives Initiative (OAI) (OPEN ARCHIVES INITIATIVE, 2016) com 66 mil documentos. As *tags* indexadas foram: descrição, assunto e título. Foram medidos os tempos de execução da ferramenta na indexação do repositório de duas abordagens diferentes (utilizando 1 núcleo e 8 núcleos). Como os documentos ficam na cache da memória RAM após uma execução, o sistema não realiza leitura de disco na execução seguinte. Deste modo, ao final de 5 dessas 10 execuções, a máquina foi reiniciada para realizar medidas de tempo de execução desconsiderando o efeito da cache nos experimentos. Por fim, mediu-se 10 vezes o tempo de extração de uma matriz atributo-valor do índice gerado.

Nesse caso não foi preciso reiniciar a máquina porque a matriz é calculada novamente a cada execução, não sofrendo com os efeitos da cache da memória. Contudo, esse processo ainda depende de operações de leitura e escrita no disco, diminuindo o ganho de desempenho do paralelismo.

ii) Buscas com thesaurus: utilizando a mesma base dos experimentos com o paralelismo, foi utilizado um vocabulário com 71.000 termos com regras obtidas dos vocabulários Thesagro (BRASIL, 2016) e do The National Agricultural Library's Agricultural Thesaurus (ESTADOS UNIDOS, 2016) para auxiliar nas buscas da IPreproc. Diversas consultas, que possuem relações nos dicionários utilizados, foram realizadas sobre os índices e foi analisado o número de resultados retornados. Os pesos escolhidos para os *broader, narrower* e *related terms* foram 0.3, 0.5 e 0.4 respectivamente. Esses pesos foram escolhidos arbitrariamente pois o objetivo deste trabalho é apenas medir o número de resultados retornados por uma consulta e não a ordem de relevância desses resultados.

Resultados e Discussão

i) Paralelismo: os resultados obtidos na indexação dos 66 mil metadados do repositório Alice estão apresentados na Tabela 1. O tempo de execução do caso 2 foi 26.14% menor do que do caso 1, enquanto a diminuição do tempo do caso 4 em relação ao caso 3 foi de 68.57%. Essa diferença se deve pelo fato de que o processo de indexação depende muito da leitura dos arquivos gravados no disco rígido da máquina. A leitura desses arquivos é responsável pela maior parte do tempo gasto durante a execução do programa. Como a leitura do disco é sequencial, apenas uma thread por vez pode realizar a leitura do disco. Portanto, as maiores responsáveis pela diminuição de tempo dos experimentos com paralelismo foram as operações de processamento. Nos casos 3 e 4 em que a máquina não foi reiniciada após cada execução da ferramenta, os documentos estavam na cache da máquina e os ganhos da abordagem paralela foram muito mais evidentes, pois dependiam muito mais do processamento. Apesar de ainda depender da leitura e escrita do disco rígido, o processo de extração de matrizes utiliza mais o processamento da máquina. Neste caso, a diminuição

do tempo de execução utilizando a abordagem paralela foi de 55.38% (Tabela 2) maior do que no caso da indexação.

ii) Buscas com *thesaurus*: O resultado obtido para 4 consultas é mostrado a seguir na Tabela 2. Nas quatro consultas consideradas, houve um aumento significativo do número de resultados retornados.

Tabela 1. Resultados do experimento de paralelismo na indexação incremental.

#	Núcleos	Reiniciou	Tempo médio	Diminuição do tempo com paralelismo
1	1	Sim	6min11s	-
2	8	Sim	4min34s	-26.14% em relação a 1
3	1	Não	35s	-
4	8	Não	11s	-68.57% em relação a 3

Tabela 2. Resultados do experimento de paralelismo na extração de Matrizes atributo-valor.

#	Núcleos	Tempo médio	Diminuição do tempo com paralelismo
1	1	1min5s	-
2	8	29s	-55.38% em relação a 1

Tabela 3. Buscas com o thesaurus.

Consulta inserida	Consulta efetuada	Resultados sem *theraurus*	Resultados com *theraurus*
air	(air OR "gases"^0.3 OR "airshed"^0.4 OR "soil air"^0.4)	90	414
abacaxi	(abacaxi OR "fruta tropical"^0.3 OR "ananas comosus"^0.4 OR "bromelina"^0.4)	181	332
candy	(candy OR "sweets"^0.3 OR "chocolate"^0.4 OR "desserts"^0.4)	1	12
chocolate	(chocolate OR "cocoa products"^0.3 OR "alimento preparado"^0.3 OR "white chocolate"^0.5 OR "milk chocolate"^0.5 OR "candy"^0.4 OR "conching"^0.4 OR "tempering"^0.4 OR "flavorings"^0.4 OR "cacau"^0.4 OR "chocolate liquor"^0.4)	10	82

Considerações Finais

A funcionalidade de paralelismo se mostrou bastante eficiente nos testes realizados. Mesmo com a maior parte do processo de indexação ser inerentemente sequencial, devido à leitura dos arquivos no disco rígido da máquina, houve uma boa diminuição no tempo de execução ao utilizar todos os núcleos da máquina. No caso da extração de matrizes atributo-valor a vantagem do paralelismo foi mais expressiva, diminuindo o tempo de execução da ferramenta em mais de 50%.

A expansão dos termos de busca utilizando relações de dicionários *thesaurus* mostrou-se eficaz no que promete, aumentando o número de resultados obtidos.

Referências

BRASIL. Ministério da Agricultura, Pecuária e Abastecimento. **O Thesagro é o único thesaurus brasileiro especializado em literatura agrícola utilizado para indexação e recuperação dos documento**. Disponível em: <http://snida.agricultura.gov.br:81/binagri/html/Cen_Thes1.html>. Acesso em: 15 ago. 2016.

EMBRAPA. **Repositório Acesso Livre à Informação Científica da Embrapa (Alice)**. Disponível em: <https://www.alice.cnptia.embrapa.br>. Acesso em: 15 ago. 2016.

ESTADOS UNIDOS. Department of Agriculture. **Thesaurus**. Disponível em: <http://agclass.nal.usda.gov/agt.shtml>. Acesso em: 18 ago. 2016.

MOURA, M. F.; TARARAM, G.; SILVA, L. A.; GONZALES, L. E.; TAKEMURA, C. M.; REZENDE, S. O.; MARCACINI, R. M.; SANTOS, F. F. dos; EVANGELISTA, S. R. M. CRITIC 1.0: ambiente web para busca e análise da informação utilizada ou produzida pela Rede AgroHidro. In: SEMINÁRIO DA REDE AGROHIDRO, 3.; WORKSHOP DO PROJETO OS IMPACTOS DA AGRICULTURA E DAS MUDANÇAS CLIMÁTICAS NOS RECURSOS HÍDRICOS, 1., 2015, Corumbá. **Água na agricultura**: desafios frente às mudanças climáticas e de uso da terra: resumos. Brasília, DF: Embrapa, 2015. p. 30.

OPEN ARCHIVES INITIATIVE. Disponível em: <https://www.openarchives.org/>. Acesso em: 15 ago. 2016.

THE APACHE SOFTWARE FOUNDATION. **Apache Lucene Core**. Disponível em: <https://lucene.apache.org/core/>. Acesso em: 15 ago. 2016.