

Processamento Eficiente de Consultas Analíticas Estendidas com Predicado de Similaridade em Spark

Guilherme Muzzi da Rocha¹, Cristina Dutra de Aguiar Ciferri¹

¹Departamento de Ciências de Computação – Universidade de São Paulo (USP)
Caixa Postal 668 – 13.560-970 – São Carlos – SP – Brasil

guilherme.muzzi.rocha@usp.br, cdac@icmc.usp.br

Abstract. *An image data warehousing extends a conventional data warehousing to also manipulate images represented by feature vectors and attributes for similarity search. A challenge that arises is the efficient processing of analytical queries extended with a similarity search predicate since these queries have a high computational cost. In this article we propose the BrOmnImg method, which efficiently solves this challenge in Spark. Compared to its closest method, BrOmnImg improved query processing up to 65.49%.*

Resumo. *Um data warehousing de imagens estende um data warehousing convencional para também manipular imagens representadas por vetores de características e atributos para pesquisa por similaridade. Um desafio que surge é o processamento de consultas analíticas estendidas com predicado de similaridade, desde que essas consultas possuem alto custo computacional. Neste artigo é proposto o método BrOmnImg, o qual soluciona eficientemente esse desafio usando o framework Spark. Comparado com o método mais próximo, BrOmnImg proveu ganhos de desempenho de até 65,49%.*

1. Introdução

Um *data warehousing* de imagens estende um *data warehousing* convencional para também manipular imagens [Teixeira et al. 2015]. O processo ETL (*extract, transform, load*) é estendido para realizar a extração das características intrínsecas das imagens, as quais são representadas por vetores de características e atributos para pesquisa por similaridade. Esses novos tipos de dados são armazenados em um banco de dados especialmente projetado para esse fim, o *data warehouse* (DW) de imagens. O processo OLAP (*on-line analytical processing*) também é expandido, de forma que ofereça suporte ao processamento de consultas analíticas estendidas com predicado de similaridade entre as imagens.

Assim, uma nova gama de consultas analíticas pode ser realizada, enriquecendo a tomada de decisão estratégica. Por exemplo, em uma organização médica voltada à análise de imagens de exames, pode-se determinar quantas imagens de câncer pulmonar são similares a uma dada imagem, considerando pacientes maiores do que 40 anos no estado de São Paulo nos últimos 3 anos [Teixeira et al. 2015]. No contexto da agricultura, medidas agrícolas referentes ao solo, fertilidade e umidade ao longo dos anos representam tipos de dados convencionais e os vetores de características de imagens armazenadas nesse contexto representam tipos de dados estendidos [Nguyen et al. 2017]. Na tomada de decisão, pode-se investigar quantas imagens são similaridades a uma determinada imagem do solo, considerando valores específicos para as medidas agrícolas.

Um desafio que surge é o processamento de consultas analíticas estendidas com predicado de similaridade. Essas consultas têm alto custo porque requerem o processamento conjunto de caras operações de junção-estrela, comuns em DWs armazenados segundo o esquema-estrela, com onerosas operações de cálculos de distância entre as imagens. Outro fato que acentua esse desafio é o gigantesco volume de dados manipulados. DWs convencionais são usualmente muito volumosos. Ademais, DWs de imagens armazenam um volume cada vez maior de imagens. Por exemplo, na área médica, diferentes hospitais, clínicas de saúde e laboratórios produzem várias imagens de exames por dia, as quais são compartilhadas visando a tomada de decisão mais robusta [Sebaa et al. 2018]. Na área agrícola, o uso de sensores possibilita a coleta massiva de dados, inclusive imagens, gerando informações valiosas no campo do agronegócio [Nguyen et al. 2017].

Neste artigo, é proposto o método *BrOmnImg*, o qual introduz uma solução eficiente para o desafio descrito. Para processar consultas analíticas estendidas com predicado de similaridade, *BrOmnImg* introduz as seguintes características. Ele integra as técnicas *broadcast join* [Brito et al. 2016] e *Omni* [Traina et al. 2007], as quais otimizam, separadamente, operações de junção-estrela sobre DWs convencionais e cálculo de distância em pesquisas por similaridade. Adicionalmente, para lidar com o gigantesco volume de dados, *BrOmnImg* é desenvolvido considerando o *framework* de processamento paralelo e distribuído Spark [Zaharia et al. 2010].

O artigo está estruturado da seguinte forma. Trabalhos correlatos são descritos na seção 2. Conceitos são resumidos na seção 3. *BrOmnImg* e os testes de desempenho são descritos nas seções 4 e 5, respectivamente. Conclusões são feitas na seção 6.

2. Trabalhos Correlatos

Em [Teixeira et al. 2015] é proposto o processamento de consultas analíticas estendidas com predicado de similaridade. Esse trabalho usa a técnica *Omni*, porém não considera o processamento dessas consultas em ambientes computacionais paralelos e distribuídos. Em [Brito et al. 2016] é proposto o método SBJ que usa a técnica de *broadcast join* para processar a junção-estrela em Spark. SBJ, descrito na seção 3, provê melhor desempenho do que seus concorrentes, sendo considerado o método mais próximo ao *BrOmnImg*.

Operações de cálculos de distância em Spark são otimizadas em [Li et al. 2017] usando-se funções *hash* projetadas para diminuir a colisão de imagens similares. Em [Nguyen and Huh 2017] é usado o método de acesso métrico VP-tree para esse fim, considerando o *framework* MapReduce [Dean and Ghemawat 2008]. Diferentemente de *BrOmnImg*, esses trabalhos não consideram operações de junção-estrela. Outras limitações referem-se à complexidade de definição de funções *hash* apropriadas e ao fato de que a *Omni* provê melhor desempenho do que a VP-tree [Traina et al. 2007].

No melhor do conhecimento dos autores deste artigo, nenhum dos trabalhos correlatos considera conjuntamente todos os aspectos que o método proposto considera, ou seja, o processamento da junção-estrela e similaridade sobre DWs de imagens em Spark.

3. Fundamentação Teórica

Em DWs relacionais, os dados são armazenados segundo um esquema-estrela composto de uma tabela de fatos que se relaciona com várias tabelas de dimensão. Isso requer a

realização de caras operações de junção-estrela no processamento das consultas OLAP, nas quais são realizadas junções entre a tabela de fatos e cada uma das tabelas de dimensão envolvidas na consulta, bem como resolvidas condições de seleção e agrupamento.

Além das tabelas de dimensão convencionais, um DW de imagens também contém outras que armazenam as características intrínsecas das imagens: uma tabela *Vetor Características* e várias tabelas *Camada Perceptual* [Teixeira et al. 2015]. A tabela *Vetor Características* contém, para cada camada perceptual de cada imagem, um vetor de características gerado por um extrator de características. Por exemplo, podem ser extraídas características referentes às camadas perceptuais de cor, textura ou forma. Os vetores de características contêm representações numéricas das imagens, e são usualmente representados no espaço métrico. Na pesquisa por similaridade, uma função de distância é usada para medir a dissimilaridade entre duas imagens por meio de seus vetores de características, de forma que a função de distância torna-se menor à medida que as imagens são mais similares. Nesse sentido, cada tabela *Camada Perceptual i* contém as distâncias entre cada imagem armazenada e cada elemento representativo da camada perceptual *i*. Esses elementos representativos são imagens estrategicamente posicionadas no espaço métrico visando a redução na quantidade das onerosas operações de cálculo de distância realizadas na pesquisa por similaridade, e são gerados pela técnica Omni [Traina et al. 2007].

Em ambientes computacionais paralelos e distribuídos, a técnica de *broadcast join* chamada SBJ [Brito et al. 2016] resolve a junção-estrela sobre DWs convencionais da seguinte forma. Ela assume que as tabelas de dimensão são suficientemente pequenas para serem enviadas para todos os nós do *cluster* durante o processamento da consulta OLAP, e realiza todas as junções, em paralelo, localmente em cada nó.

4. O Método *BrOmnImg*

Nesta seção é proposto o método *BrOmnImg* (acrônimo para *Broadcast Omni for processing analytical Image queries*) para processar, em Spark, consultas analíticas estendidas com predicado de similaridade sobre um DW de imagens conforme definido na seção 3.

As consultas possuem dois tipos de predicado: convencional e de similaridade. O predicado convencional é composto por condições de seleção, sendo cada condição definida sobre um atributo armazenado em uma tabela de dimensão convencional. O predicado de similaridade é composto por uma operação de similaridade e pelas camadas perceptuais consideradas. Por exemplo, na consulta “Liste a quantidade de imagens similares a uma dada imagem, para pacientes do sexo feminino diagnosticadas com câncer de mama e as camadas perceptuais de cor e textura”, tem-se: (i) pacientes do sexo feminino e câncer de mama como condições de seleção; (ii) o uso da operação de *range query* para resolver a similaridade; e (iii) as camadas perceptuais de cor e textura.

BrOmnImg integra as técnicas *broadcast join* e *Omni* da seguinte forma (Figura 1). As condições de seleção são aplicadas sobre as tabelas de dimensão convencionais relacionadas (Figura 1a). Para cada tabela *k*, os dados filtrados são armazenados em uma estrutura *HashMapConvencional_k*. Com relação ao predicado de similaridade, primeiramente a operação de similaridade é aplicada sobre todas as tabelas *CamadaPerceptual_i* consideradas (Figura 1b). Para cada camada perceptual *i*, seus dados são filtrados utilizando a técnica Omni, gerando resultados candidatos armazenados nas estruturas *HashMapFiltragem_i*, as quais são transmitidas para todos os nós. Na

sequência, são realizadas as operações de cálculo de distância usando os resultados candidatos e a tabela *Vetor Características* para eliminar falsos positivos (Figura 1c). Os resultados gerados são aqueles que atendem ao predicado de similaridade, e são armazenados na estrutura *HashMapRefinamento*. Por fim, as estruturas *HashMapConvencional_k* e a estrutura *HashMapRefinamento* são transmitidas para todos os nós do *cluster* para a realização da junção-estrela estendida sobre a tabela de fatos (Figura 1d).

Spark é baseado em computação em memória e na abstração de RDDs (*resilient distributed dataset*) [Zaharia et al. 2010]. *BrOmniImg* armazena cada tabela do esquema-estrela como um RDD. Ele aplica os filtros convencionais e de imagem sobre os RDDs correspondentes usando a operação *filter*, e armazena os dados nas estruturas *hash map* por meio das operações *mapToPair* e *collect*. Adicionalmente, *BrOmniImg* realiza a junção-estrela estendida aplicando as operações *filter*, *mapToPair* e *reduceByKey*.

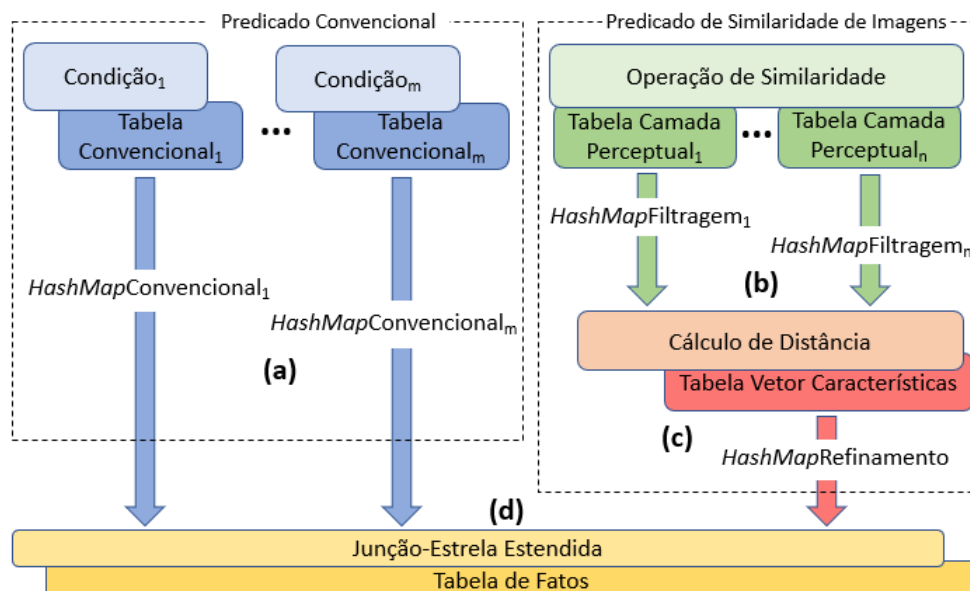


Figura 1. Visão geral do método proposto.

5. Testes de Desempenho

Dados. O DW de imagens da área médica foi povoado com dados gerados pela ferramenta *ImgDW Generator* [Rocha and Ciferri 2018]. Foram projetadas 2 tabelas de dimensão convencional: *Pacientes*, com 2 milhões de tuplas, e *Descrição Exame*, com 20 milhões de tuplas. Foram projetadas 2 tabelas *Camada Perceptual*: *Histograma Cores* e *Haralick Variância*, cada qual contendo as distâncias entre cada imagem e 3 elementos representativos gerados pela Omni. Foram inseridas 20 milhões de tuplas na tabela de fatos *Exame*, em *Histograma Cores*, em *Haralick Variância* e na tabela *Vetor Características*.

Consultas. As consultas analíticas estendidas com predicado de similaridade foram definidas considerando até 3 condições de seleção do predicado convencional (sexo *feminino*, diagnóstico de *câncer* e parte do corpo sendo *mama*) e 1 predicado de similaridade especificado pela operação *range query* [Traina et al. 2007] com um raio de abrangência de 20% do diâmetro do conjunto de dados, considerando cada camada perceptual envolvida.

Configurações. As configurações consideraram a seletividade do predicado convencional e a dimensionalidade das camadas perceptuais. Para o predicado convencional, foram definidas consultas: (i) sem predicado - *SemConv*; (ii) com predicado de baixa seletividade (33,33%) envolvendo os pacientes do sexo feminino - *BaixaSel*; e (iii) com predicado de alta seletividade (0,08%) envolvendo os pacientes do sexo feminino com câncer de mama - *AltaSel*. Para o predicado de similaridade, foram definidas consultas sobre: (i) a camada perceptual de alta dimensionalidade *Histograma Cores* (256 dimensões) - *His*; a camada perceptual de baixa dimensionalidade *Haralick Variância* (4 dimensões) - *HV*; e (iii) essas duas camadas conjuntamente - *His/HV*. No total, foram geradas 9 configurações, conforme mostrado na Tabela 1. Para cada configuração, o número de imagens que atenderam aos predicados convencional e de similaridade foram: *C1*: 224.060; *C2*: 125.460; *C3*: 5.620; *C4*: 74.800; *C5*: 42.000; *C6*: 1.800; *C7*: 160; *C8*: 180; *C9*: 20.

Tabela 1. Configurações definidas para os testes de desempenho

	His	HV	His/HV
SemConv	(C1) SemConvHis	(C2) SemConvHV	(C3) SemConvHis/HV
BaixaSel	(C4) BaixaSelHis	(C5) BaixaSelHV	(C6) BaixaSelHis/HV
AltaSel	(C7) AltaSelHis	(C8) AltaSelHV	(C9) AltaSelHis/HV

Execução. Foi usado um *cluster* com 5 nós, cada qual com, no mínimo, 3GB de RAM. *BrOmniImg* foi comparado com SBJ, que representa o trabalho correlato mais próximo da literatura. Cada consulta foi executada 5 vezes, sendo obtida a média das execuções.

Resultados. Na Figura 2 são ilustrados os resultados obtidos, bem como o desvio padrão. Para as configurações que possuem, pelo menos, uma camada perceptual de alta dimensionalidade (*C1*, *C3*, *C4*, *C6*, *C7*, *C9*), *BrOmniImg* proveu ganhos de desempenho que variaram de 60,01% a 65,49% quando comparado com SBJ. Isso está relacionado à complexidade dos cálculos de distância da técnica Omni, sendo dependente da dimensionalidade do conjunto de dados e do número de elementos representativos. A diferença entre o número de dimensões de *Histograma Cores* (256) e o número de elementos representativos correspondente (3) impactou positivamente no desempenho de *BrOmniImg*. Para as configurações que possuem apenas a camada perceptual de baixa dimensionalidade (*C2*, *C5*, *C8*), *BrOmniImg* empatou com SBJ ou proveu ganhos de desempenho de até 10,50%. Nessas configurações, a diferença entre o número de dimensões de *Haralick Variância* (4) e o número de elementos representativos correspondente (3) foi muito pequena. Positivamente, mesmo *BrOmniImg* sendo mais complexo do que SBJ por integrar a técnica Omni para processar o predicado de similaridade, ou ele empatou ou provê melhores resultados.

6. Conclusão

Neste artigo é proposto o método *BrOmniImg*, voltado ao processamento eficiente de consultas analíticas estendidas com predicado de similaridade sobre *data warehouses* de imagens. *BrOmniImg* mostra a viabilidade de integração das técnicas de *broadcast join* e Omni em Spark. Comparado com o método mais próximo, *BrOmniImg* garantiu melhora de desempenho expressiva para predicados de similaridade definidos sobre camadas perceptuais de alta dimensionalidade. Trabalhos futuros incluem a realização de novos testes de desempenho considerando outras consultas e diferentes operações de similaridade.

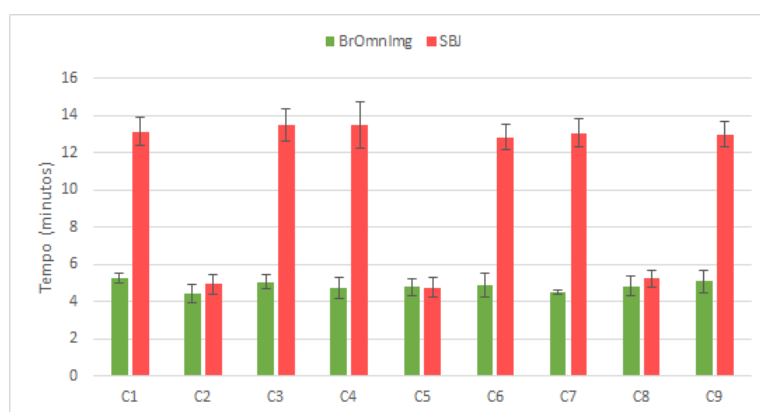


Figura 2. Testes de desempenho comparando *BrOmnImg* com *SBJ*.

Agradecimentos. Trabalho sendo desenvolvido com recursos financeiros da FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo), processos 2018/10607-3 e 2018/22277-8, e do CNPq.

Referências

- Brito, J. J., Mosqueiro, T., Ciferri, R. R., and Ciferri, C. D. A. (2016). Faster cloud star joins with reduced disk spill and network communication. *Procedia Computer Science*, 80:74 – 85.
- Dean, J. and Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113.
- Li, D., Zhang, W., Shen, S., and Zhang, Y. (2017). SES-LSH: Shuffle-efficient locality sensitive hashing for distributed similarity search. In *ICWS 2017*, pages 822–827.
- Nguyen, T. D. T. and Huh, E.-N. (2017). An efficient similar image search framework for large-scale data on cloud. In *IMCOM 2017*, pages 65:1–65:8.
- Nguyen, V.-Q., Ngoc, N., and Kim, K. (2017). Design of a platform for collecting and analyzing agricultural big data. *Journal of Digital Contents Society*, 18:149–158.
- Rocha, G. M. and Ciferri, C. D. A. (2018). ImgDW generator: a tool for generating data for medical image data warehouses. In *SBB 2018 Proc. Companion*, pages 23–28.
- Sebaa, A., Chikh, F., Nouicer, A., and Tari, A. (2018). Medical big data warehouse: Architecture and system design, a case study: Improving healthcare resources distribution. *Journal of Medical Systems*, 42(4):59.
- Teixeira, J. W., Annibal, L. P., Felipe, J. C., Ciferri, R. R., and Ciferri, C. D. A. (2015). A similarity-based data warehousing environment for medical images. *Computers in Biology and Medicine*, 66:190 – 208.
- Traina, C., Filho, R. F. S., Traina, A. J. M., Vieira, M. R., and Faloutsos, C. (2007). The Omni-family of all-purpose access methods: a simple and effective way to make similarity search more efficient. *The VLDB Journal*, 16(4):483–505.
- Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., and Stoica, I. (2010). Spark: Cluster computing with working sets. In *USENIX HotCloud 2010*.