28TH BRAZILIAN SYMPOSIUM ON DATABASES

THESIS AND DISSERTATION WORKSHOP (WTDBD)

PROCEEDINGS

September 30th – October 3rd, 2013
Recife, Pernambuco, Brazil

Promotion

Brazilian Computer Society – SBC
SBC Special Interest Group on Databases

Organization

Universidade Federal de Pernambuco – UFPE

Realization

Centro de Informática (CIn)

Thesis and Dissertation Workshop Chairs

Karin Becker, UFRGS and André Santanchè, UNICAMP

Editorial

The Workshop of Thesis and Master Dissertations in Databases (WTDBD) is a traditional event co-located with the Brazilian Symposium on Databases. In its 12nd edition, WTDB takes place in the wonderful city of Recife in 2013. WTDBD provides a representative sample of the excellent research work in the area of Databases, which is under development at the graduate programs of Brazilian Universities. It is a great opportunity to gather professors and graduate students to discuss research, and it allows graduate students to receive from experienced researchers, constructive feedback about their on-going work. All submitted papers received three reviews. Additionally, during the Workshop, students of selected papers have the opportunity of presenting their work to at least 2 senior researchers. Hence, they can experience the challenge of presenting their work to an external committee, as well as receive feedback. This excellent program is completed with an invited talk of Prof. Mirella Moro, who will provide useful guidance and tips for writing and presenting scientific work.

Although we tried to be as inclusive as possible, the number of quality submissions exceeded the Workshop presentation slots. In terms of distribution, submitted papers refer to Graduate Programs of universities on the Northeast (6), Southeast (8) and South (5) of Brazil. Unfortunately not all submissions could be selected. A total of 15 works were accepted for submitting a final version of their papers, 14 of which will be presented at the workshop.

The WTDBD2013 chairs would like to thank the students and advisors that submitted their work to the workshop. Similarly, we are very grateful to the reviewers, who spent significant time to provide elaborate, detailed and constructive reviews, targeted at helping students to reach a successful completion of their graduate programs. We also thank all researchers who will take part in the jury during the workshop. Finally, the WTDBD chairs would like to thank the SBBD 2013 organization for the support and excellent collaboration in preparing this edition of the event.

We wish the community an excellent workshop and success in their works,

Karin Becker
SBBD 2013, Thesis and Dissertation Workshop (WTDBD) Chair

André Santanchè
SBBD 2013, Thesis and Dissertation Workshop (WTDBD) Chair

28TH BRAZILIAN SYMPOSIUM ON DATABASES

September 30th – October 3rd, 2013

Recife, Pernambuco, Brazil

Promotion

Brazilian Computer Society – SBC
SBC Special Interest Group on Databases

Organization

Universidade Federal de Pernambuco – UFPE

Realization

Centro de Informática (CIn)

SBBD Steering Committee

Marco A. Casanova, PUC-Rio (Chair)
Ana Carolina Salgado, CIn-UFPE
Cristina Dutra de Aguiar Ciferri, USP
José Palazzo Moreira de Oliveira, UFRGS
Mirella M. Moro, DCC-UFMG

SBBD 2013 Committee

Steering Committee Chair
Marco A. Casanova, PUC-Rio
Local Organization Chair
Bernadette Farias Lóscio, CIn-UFPE
Program Committee Chair
Cristina Dutra de Aguiar Ciferri, USP
Short Papers Chairs
Renato Fileto, UFSC and Marco Cristo, UFAM
Demos and Applications Chairs
Damires Souza, IFPB and Daniel Kaster, UEL
Thesis and Dissertation Workshop Chairs
Karin Becker, UFRGS and André Santanchè, UNICAMP
Tutorials Chair
Mirella M. Moro, DCC-UFMG
Lectures Chair
João Eduardo Ferreira, IME-USP

Local Organization Committee

Bernadette Farias Lóscio, CIn-UFPE (Chair)
Ana Carolina Salgado, CIn-UFPE
Agenor Marinho de Souza Neto, CIn-UFPE

Carlos Eduardo Santos Pires, DSC-UFCG
Danusa Ribeiro Cunha, CIn-UFPE
Priscilla Vieira, CIn-UFPE
Robson Fidalgo, CIn-UFPE

WTDBD 2013 Program Committee

Carmem Satie Hara, UFPR
Duncan Ruiz, PUCRS
Fabio Porto, LNCC
Javam C. Machado, UFC
João Eduardo Ferreira, USP
José Fernando Rodrigues Júnior, USP
Fernanda Baião, UNIRIO
Leandro Wives, UFRGS
Luciano Antônio Digiampietri, USP
Mirella M. Moro, UFMG
Ronaldo dos Santos Mello, UFSC
Sandra de Amo, UFU
Vanessa Braganholo, UFF
Vaninha Vieira dos Santos, UFBa

WTDBD 2013 Jury

Carmem Hara, UFPR
Damires Souza, IFPB
Duncan Ruiz, PUCRS
Fabio Porto, LNCC
Fernando Fonseca, UFPE
João Eduardo Ferreira, USP
Jose Fernando Rodrigues Jr., USP - São Carlos
Renata Galante, UFRGS
Renato Fileto, UFSC
Ronaldo Mello, UFSC
Vanessa Braganholo, UFF

Mirella M. Moro

Short bio:

Mirella M. Moro é professora adjunta do Departamento de Ciência da Computação (DCC) da Universidade Federal de Minas Gerais (UFMG). Possui doutorado em Ciência da Computação pela University of California in Riverside (2007), e graduação e mestrado em Ciência da Computação pela Universidade Federal do Rio Grande do Sul (UFRGS). É Diretora de Educação da SBC (Sociedade Brasileira de Computação), editora-chefe da revista eletrônica SBC Horizontes e membro do Education Council da ACM. Seus interesses de pesquisa estão na área de Banco de Dados, incluindo tópicos como processamento de consultas, disseminação e recomendação de conteúdo, redes sociais e XML.

Title:

Pós-Graduação em Bancos de Dados: Escrita, Apresentação e Além

Abstract:

Esta palestra aborda uma coletânea de dicas e uma visão geral do processo pelo qual *todos* os mestrandos e doutorandos passam no decorrer do seu curso. Além disso, questões importantes também são abordadas incluindo: como escrever textos e participar de eventos científicos; por que fazer mestrado/doutorado em bancos de dados; como se faz pesquisa; como começar e finalizar o curso. Mais importante ainda são os esclarecimentos sobre como a vida do estudante de pós-graduação muda durante o curso. É uma oportunidade única para ter acesso a uma coletânea de informações organizadas em apenas 50 minutos.

28TH BRAZILIAN SYMPOSIUM ON DATABASES

THESIS AND DISSERTATION WORKSHOP (WTDBD)

Table of Contents

An Adaptive Blocking Approach for Entity Matching with MapReduce 01
Demetrio Gomes Mestre, Carlos Eduardo Pires

OPIS: Um método para identificação e busca de páginas-objeto apoiado por realimentação de relevância e classificação de páginas web 08
Miriam Pizzatto Colpo, Edimar Manica, Renata Galante

Banco de Dados em Nuvem: Um Modelo para Garantia de Consistência dos Dados 15
Elyda Laisa S. X. Freitas, Fernando da Fonseca de Souza

Correlation between the quality of focused crawlers and the linguistic resources obtained from them 22
Bruno Rezende Laranjeira, Viviane P. Moreira, Aline Villavicencio

Análise da Evolução Temporal de Dados Complexos 29
Isis Caroline O. V. de Sousa, Renato Bueno

Efficient integrity checking for untrusted database systems 36
Anderson Luiz Silvério, Ronaldo dos Santos Mello, Ricardo Felipe Custódio

Avaliação da Qualidade em Linked Datasets: uma abordagem com foco nos requisitos da aplicação 43
Walter Travassos Sarinho, Bernadette Farias Lóscio, Damires Souza

Incorporando Dados Espaciais Vagos em Data Warehouses Geográficos: A Proposta do Tipo Abstrato de Dados VagueGeometry 50
Anderson Chaves Carniel, Ricardo Rodrigues Ciferri

ImageDW-index: Uma estratégia de indexação voltada ao processamento de imagens em data warehouses 57
Jefferson William Teixeira, Cristina Dutra de Aguiar Ciferri

Utilizando Regras baseadas no Contexto para Reescrever Consultas 64
Antônio Ezequiel de Mendonça, Ana Carolina Salgado, Damires Souza

Mineração de Preferências em Data Streams 71
Jaqueline Aparecida Jorge Papini, Sandra de Amo

Recomendação de Consultas de Banco de Dados utilizando Agrupamentos de Usuários. 78
Márcio de Carvalho Saraiva, Carlos Eduardo Pires, Leandro Balby Marinho

APPWM - Agrupamento Personalizado de Pontos em Web Maps usando um modelo multi-dimensional 85
Marcio Bigolin, Helena Grazziotin Ribeiro, Renata Galante

HSTB-index: A Hierarchical Spatio-Temporal Bitmap Indexing Technique 92
Cesar Joaquim Neto, Ricardo Rodrigues Ciferri, Marilde Terezinha Prado Santos

An Adaptive Blocking Approach for Entity Matching with MapReduce

Demetrio Gomes Mestre[1], **Prof. Dr. Carlos Eduardo Pires**[1]

[1]Programa de Pós-Graduação em Ciência da Computação
Universidade Federal do Campina Grande (UFCG)
Caixa Postal 10.106 – 58.429-900 – Campina Grande – PB – Brazil

demetriogm@gmail.com, cesp@dsc.ufcg.edu.br

Nível: Mestrado
Ano de ingresso no programa: 2012
Defesa da proposta: Dezembro de 2012
Época esperada de conclusão: Março de 2014

Abstract. *Cloud computing has proven to be a powerful ally to efficient parallel execution of data-intensive tasks such as Entity Matching (EM) in the era of Big Data. For this reason, studies about challenges and possible solutions of how EM can benefit from the cloud computing paradigm have become an important demand nowadays. In this context, we investigate how the MapReduce programming model can be used to perform efficient parallel EM using a variation of the Sorted Neighborhood Method (SNM) that uses a varying size window. We propose Distributed Duplicate Count Strategy (DDCS), an efficient MapReduce-based approach for this adaptive SNM, aiming to decrease even more the execution time of SNM.*

Keywords: *MapReduce, Entity Matching, Adaptive Window, Sorted Neighborhood Method.*

1. Introduction

Distributed computing has received a lot of attention lately to perform high data-intensive tasks. Extensive powerful distributed hardware and service infrastructures capable of processing millions of these tasks are available around the world and have being used by industry to streamline its heavy data processing. To make efficient use of these distributed infrastructures, the MapReduce (MR) programming model [Dean and Ghemawat 2008] emerges as a major alternative since it can efficiently perform the distributed data-intensive tasks through map and reduce operations, can scale parallel shared-nothing data-processing and is broadly available in many distributions, such as Hadoop [1].

Entity Matching (EM) (also known as entity resolution, deduplication, or record linkage) is such a data-intensive and performance critical task that demands studies on how it can benefit from cloud computing. EM is applied to determine all entities referring to the same real world object given a set of data sources [Kopcke and Rahm 2010]. For example, in master-data-management applications[2], a system has to identify that the names "Jon S. Stark", "Stark, Jon" and "Jon Snow Stark" are potentially referring to the same person. Thus, the task has critical importance for data cleaning and integration.

Perform EM processes is challenging nowadays. Besides the need of applying matching techniques on the Cartesian product of all input entities which leads to a computational cost in the order of $O(n^2)$, there is an increasing trend of applications being expected to deal with vast amounts of data that usually do not fit in the main memory of one machine. This means that the application of such approach is ineffective for large datasets. One way to minimize the workload caused by the Cartesian product execution and to maintain the match quality is reducing the search space by applying blocking techniques. Such techniques work by partitioning the input data into blocks of similar entities and restricting the EM to entities of the same block [Baxter et al. 2003]. For instance, it is sufficient to compare entities of the same manufacturer when matching product offers.

The Sorted Neighborhood Method (SNM) is one of the most popular blocking approaches [Hernández and Stolfo 1995]. It sorts all entities using an appropriate blocking key, e.g., the first three letters of the entity name, and only compares entities within a predefined distance window w. The SNM approach thus reduces the complexity to $O(n \cdot w)$ for the actual matching. Figure 1 shows an execution example of SNM for a window size w = 3. The input set S consists of n = 9 entities (from a to i) and all the entities are sorted according to their blocking key K (1, 2, or 3). Initially, the window includes the first three entities (a, d, b) and generates three pairs of comparisons $[(a, d), (a, b), (d, b)]$. Later, the window is slided down (one entity) to cover the entities d, b, e and two more pairs of comparisons are generated $[(d, e), (b, e)]$. The sliding process is repeated until the window reaches the last three entities (c, g, i). Note that the number of comparisons generated is $(n - w/2) \cdot (w - 1)$.

However, the SNM presents a critic performance disadvantage due to the fixed and difficult to configure window size: if it is selected too small, some duplicates might be missed. On the other hand, a too large window results in many unnecessary comparisons. Note that if effectiveness is most relevant, the ideal window size is equal to the size of

[1] https://hadoop.apache.org

[2] A set of tools that consistently defines and manages the master data (i.e. non-transactional data entities) of an organization.

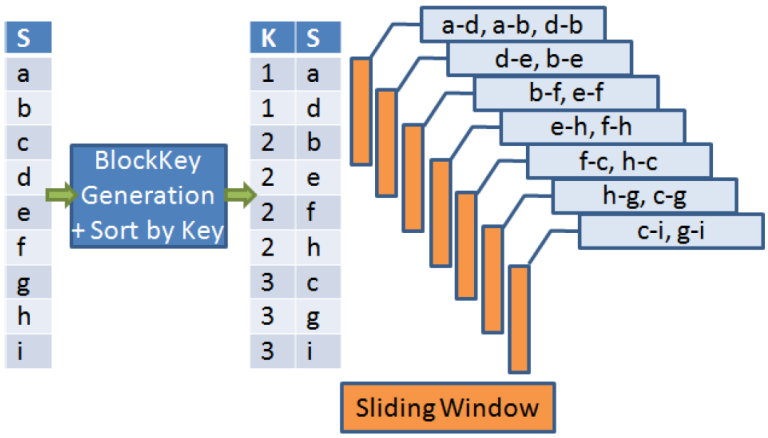


Figure 1. Execution example of the sorted neighborhood method with window size w = 3 (adapted from [Kolb et al. 2012b]).

the largest duplicate sequence in the dataset. Thus, it is not uncommon the intervention of a data specialist to solve this tradeoff (small/large window size). To overcome this disadvantage, the authors of [Draisbach et al. 2012] proposed an efficient SNM variation named as Duplicate Count Strategy or simply **DCS** that follows the idea of increasing the window size in regions of high similarity and decreasing the window size in regions of lower similarity. They also proved that their adaptive SNM overcomes the traditional SNM in performance terms by given at least the same matching results with a significant reduction in the number of comparisons.

Even with significant advances in the SNM design, EM remains a critical task in terms of performance when applied to large datasets. Thus, this work aims to propose a MapReduce-based approach capable of combining the efficiency gain achieved by the **DCS** method with the benefit of efficient parallelization of data-intensive tasks in cloud infrastructures to decrease even more the execution time of EM tasks performed with the SNM (briefly, combine the best of the two worlds).

2. MapReduce and Entity Matching

MapReduce is a programming model designed for parallel data-intensive computing in shared-nothing clusters with a large number of nodes [Dean and Ghemawat 2008]. The key idea relies on data partitioning and storage in a distributed file system (DFS). Entities are represented by (key, value) pairs. The computation is expressed with two user-defined functions:

$$map : (key_{in}, value_{in}) \rightarrow list(key_{tmp}, value_{tmp})$$

$$reduce : (key_{tmp}, list(value_{tmp})) \rightarrow list(key_{out}, value_{out})$$

Each of these functions can be executed in parallel on disjoint partitions of the input data. For each input key-value pair, the map function is called and outputs a temporary key-value pair that will be used in a shuffle phase to sort the pairs by their keys and send them to the reduce function. Unlike the map function, the reduce function is called every time a temporary key occurs as map output. However, within one reduce function only the corresponding values $list(value_{tmp})$ of a certain key_{tmp} can be accessed. A MR

cluster consists of a set of nodes that run a fixed number of map and reduce jobs. For each MR job execution, the number of map tasks (m) and reduce tasks (r) is specified. The framework-specific scheduling mechanism ensures that after a task has finished, another task is automatically assigned to the released process.

Although there are several frameworks that implement the MapReduce programing model, in the scientific community, Hadoop is the most popular implementation of this paradigm. We are therefore implementing and evaluating our approach with Hadoop.

Parallel EM implementation using blocking approaches with MR can be done without major difficulties. In a simple way, denoted as **Basic** [Kolb et al. 2012a], the map process defines the blocking key for each input entity and outputs a key-value pair (blockingKey, entity). Thereafter, the default hash partitioning in the shuffle phase can use the blocking key to assign the key-value pairs to the proper reduce task. The reduce process is responsible for performing the entity matching computation for each block. An evaluation of the **Basic** approach showed a poor performance due to the data skewness caused by varying size of blocks [Kolb et al. 2012a]. The data skewness problem occurs when the match work of large blocks of entities is assigned to a single reduce task. It can lead to situations in which the execution time may be dominated by a few reduce tasks and thus enable serious memory and load balancing problems when processing too large blocks. Therefore, concerns about lack of memory and load imbalances become necessary.

3. Adaptive Windows for Entity Matching with MapReduce

Given the importance of researching approaches to enhance the Adaptive SNM performance in the context of distributed computing, the goal of this dissertation is to propose an adaptive SNM approach based on the MapReduce model to solve the hard EM task.

To achieve the goal, we divided the work in three phases (with the first two already implemented). In the first phase, we focused on the model development to enable the fully parallelization of the **DCS** method without worrying about the lack of memory or load imbalances in the cloud infrastructure. In the second phase, we extended the model generated in the first phase to address the problem of high memory resources consumption due to the entities replication in the map phase. The third phase is future work and will be discussed in Section 6.

The adaptive SNM MR-based model, named as Distributed Duplicate Count Strategy or simply **DDCS**, proposed in the second phase is defined as the scheme of Figure 2. Two MR processes were used, both based on the corresponding number of tasks and number of input partitions (input dataset). The first process (analysis) consists in a preprocessing step capable of collecting information about the partitions allowed to be replicated and that must be sent to a specific reduce task of the second MR process. This information is formed from the specification of the target reduce task followed by the key index of each partition stored in an array, denoted as Partition Allocation Matrix (PAM), that will be used by the second MR process. The PAM is requested during the execution of the map phase in the second MR process to automatically allocate the partitions to the proper reduce tasks. Furthermore, the PAM enables an efficient redistribution of the entities to the reduce phase, which in turn performs the adaptive sliding window and the entities pairs matching.

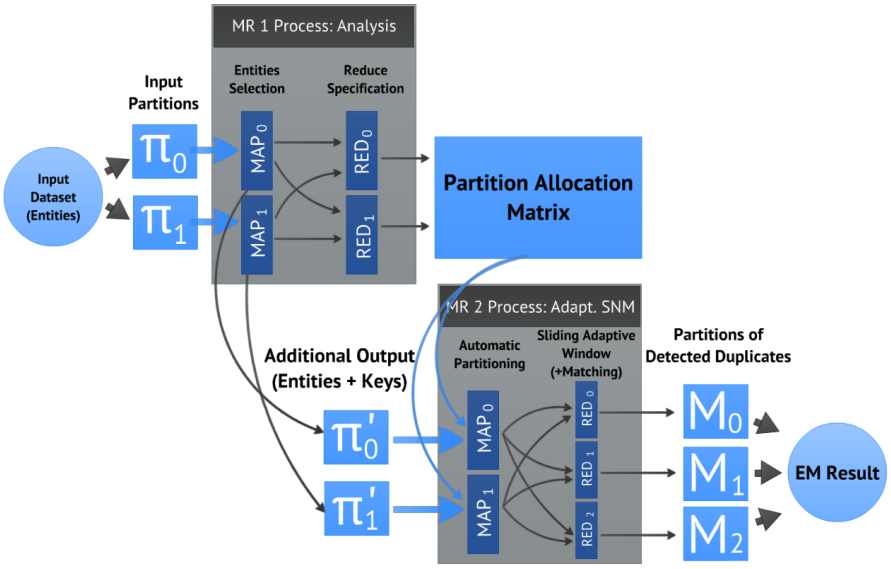


Figure 2. Schematic overview of the Adaptive SNM with MapReduce.

4. Experimental Evaluation

At the present time, we are evaluating **DDCS** against **RepSN** [Kolb et al. 2012b], a MR-based approach for the traditional SNM (fixed window). We are running **RepSN** from the Dedoop[3] aplication provided by the authors of [Kolb et al. 2012b]. The evaluation is regarding two performance-critical factors: the number of configured map and reduce tasks and the number of available nodes in the cloud environment. In each experiment we evaluate the algorithms behavior when dealing with the resources consumption caused by the use of several map and reduce tasks and how they can scale with the number of available nodes.

We are running our experiments on a 10-node HP Pavilion P7-1130 cluster. Each node has one Intel I5 processor with four cores, 4 GB of RAM, and one 1TB of hard disk. Thus the cluster consists of 40 cores and 10 disks. On each node, we installed the Windows 7 64 bits, JAVA 1.6, cygwin and Hadoop 0.20.2. We are utilizing two real-world datasets. The first dataset DS1 is a sample of the Ask[4] database that contains about 214,000 question records. The second dataset DS2, DBLP[5], contains about 1.46 million publication records. For both datasets, the first three letters of the questions or publication title, respectively, form the default blocking key. Since our work focus on performance (execution time), any attribute could have been used to form the default blocking key. Two entities are compared by computing the Jaccard similarity of their comparing attributes and those pairs with a similarity ≥ 0.85 are regarded as matches.

The partial results show that, using DS1 and up to five nodes, although **DDCS** generates 10% more map outputs than **RepSN**, **DDCS** outperforms **RepSN** around 10 to 17% in terms of execution time. Using DS1 and more than five nodes, **DDCS** still generates 10% more map outputs than **RepSN**, but **DDCS** outperforms **RepSN** around 20 to 30% in terms of execution time. The partial result seems to be promising and will be

[3]http://dbs.uni-leipzig.de/howto_dedoop
[4]http://ask.com
[5]http://www.informatik.uni-trier.de/ ley/db/

deeper investigated.

5. Related Work

EM is a very studied research topic. Many approaches have been proposed and evaluated as described in a recent survey [Kopcke and Rahm 2010]. However there are only a few approaches that consider parallel entity matching. The first steps in order to evaluate the parallel Cartesian product of two sources is described in [Kim and Lee 2007]. [Kirsten et al. 2010] proposes a generic model for parallel entity matching based on general partitioning strategies that take memory and load balancing requirements into account.

In this context, when we deal with MapReduce-based large-scale Entity Matching, two well-known data management problems must be treated: load balancing and skew handling. MR has been criticized for having overlooked the skew issue [DeWitt and StoneBraker].

[Okcan and Riedewald 2011] applied a static load balancing mechanism, but it is not suitable due to arbitrary join assumptions. The authors employ a previous analysis phase to determine the datasets' characteristics (using sampling) and thereafter avoid the evaluation of the Cartesian product. This approach focus on data skew handled in the map process output, which leads to an overhead in the map phase and large amount of map output.

MapReduce has already been employed for EM (e.g., [Wang et al. 2010]) but only one mechanism of near duplicate detection by the PPjoin paradigm adapted to the MapReduce framework can be found. [Kolb et al. 2012a, Mestre and Pires 2013] study load balancing and skew handling mechanisms to MapReduce-based EM for the standard blocking approach. [Vernica et al. 2010] shows another approach for parallel processing entity matching on a cloud infrastructure. This study explains how a single token-based string similarity function performs with MR. This approach suffers from load imbalances because some reduce tasks process more comparisons than the others.

[Kolb et al. 2012b] study load balancing for traditional Sorted Neighborhood Method (SNM). SNM follows a different blocking approach (fixed window size) that is by design less vulnerable to skewed data. However, its fixed window size design is the reason of the serious disadvantage mentioned earlier in Section 1.

6. Partial Conclusion and Future Work

In this work, we have introduced DDCS, an adaptive Sorted Neighborhood Method based on the MapReduce model that addresses the time-consuming Entity Matching problem. DDCS intends to decrease even more the execution time of the Sorted Neighborhood Method in the cloud by combining a sophisticated strategy of adaptive window with the already renowned MapReduce model. As future work, we will extend DDCS (generated in the second phase, Section 3) to address the load balancing problem (all nodes must have similar working time) by sending an approximate number of comparisons to each node and thus improve the execution time of our method.

References

Baxter, R., Christen, P., and Churches, T. (2003). A comparison of fast blocking methods for record linkage. In *ACM SIGKDD '03 Workshop on Data Cleaning, Record Linkage, and Object Consolidation*, pages 25–27.

Dean, J. and Ghemawat, S. (2008). Mapreduce: simplified data processing on large clusters. *Commun. ACM*, 51(1):107–113.

DeWitt, D. and StoneBraker, M. Mapreduce: A major step backwards, 2008, http://homes.cs.washington.edu/ billhowe/mapreduce_a_major_step_backwards.html.

Draisbach, U., Naumann, F., Szott, S., and Wonneberg, O. (2012). Adaptive windows for duplicate detection. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*, ICDE '12, pages 1073–1083, Washington, DC, USA. IEEE Computer Society.

Hernández, M. A. and Stolfo, S. J. (1995). The merge/purge problem for large databases. *SIGMOD Rec.*, 24(2):127–138.

Kim, H.-s. and Lee, D. (2007). Parallel linkage. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, CIKM '07, pages 283–292, New York, NY, USA. ACM.

Kirsten, T., Kolb, L., Hartung, M., Gross, A., Köpcke, H., and Rahm, E. (2010). Data partitioning for parallel entity matching. *CoRR*, abs/1006.5309.

Kolb, L., Thor, A., and Rahm, E. (2012a). Load balancing for mapreduce-based entity resolution. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*, ICDE '12, pages 618–629, Washington, DC, USA. IEEE Computer Society.

Kolb, L., Thor, A., and Rahm, E. (2012b). Multi-pass sorted neighborhood blocking with mapreduce. *Comput. Sci.*, 27(1):45–63.

Kopcke, H. and Rahm, E. (2010). Frameworks for entity matching: A comparison. *Data Knowl. Eng.*, 69(2):197–210.

Mestre, D. G. and Pires, C. E. (2013). Improving load balancing for mapreduce-based entity matching. In *Proceedings of the XVIII IEEE symposium on Computers and Communications*, ISCC '13. IEEE Computer Society.

Okcan, A. and Riedewald, M. (2011). Processing theta-joins using mapreduce. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, SIGMOD '11, pages 949–960, New York, NY, USA. ACM.

Vernica, R., Carey, M. J., and Li, C. (2010). Efficient parallel set-similarity joins using mapreduce. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, SIGMOD '10, pages 495–506, New York, NY, USA. ACM.

Wang, C., Wang, J., Lin, X., Wang, W., Wang, H., Li, H., Tian, W., Xu, J., and Li, R. (2010). Mapdupreducer: detecting near duplicates over massive datasets. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, SIGMOD '10, pages 1119–1122, New York, NY, USA. ACM.

OPIS: Um método para identificação e busca de páginas-objeto apoiado por realimentação de relevância e classificação de páginas web*

Miriam Pizzatto Colpo[1], Edimar Manica (Colaborador)[1], Renata Galante (Orientadora)[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

{mpcolpo,edimar.manica,galante}@inf.ufrgs.br

Nível: Mestrado

Programa: Programa de Pós-graduação em Computação (PPGC) da Universidade Federal do Rio Grande do Sul (UFRGS)

Ingresso: Março/2012

Época esperada para conclusão: Março/2014

Etapas concluídas: Créditos (2012), Proposta (Outubro/2012) e Seminário de Andamento (Maio/2013)

Etapas futuras: Submissão de Artigos (Julho/2013 – Março/2014) e Defesa da Dissertação (Março/2014)

Abstract. *This paper proposes a new method for identifying and searching object pages named OPIS (acronyms to* ***O****bject* ***P****age* ***I****dentifying and* ***S****earching). Object pages are pages that represent exactly one inherent real-world object on the web. The purpose of OPIS is to address the search for these real-world objects pages, since the General Search Engines (GSEs) cannot answer satisfactorily this type of search today. The kernel of our method is to adopt feedback relevance and machine learning techniques in the task of content-based pages classification. OPIS, when integrated into a GSE, enables the filtering of object pages, in which only pages classified as object pages are retrieved by user keyword queries instead of all pages that contain those words. Preliminary experiments show that OPIS improved on average 37% of the precision in 20 (p@20) of the results retrieved when compared with a GSE.*

Palavras-chave: páginas-objeto, busca-objeto, classificação de páginas *web.*

* Este trabalho é parcialmente financiado pelo Instituto Nacional de Pesquisa da Web, pelo CNPq e pela CAPES.

1. Introdução

Os motores de busca convencional da web (do inglês, *General Search Engines* – GSEs) são programas que visam recuperar informações da web e apresentá-las, de forma organizada e eficiente, aos usuários [Baeza-Yates e Ribeiro-Neto 2011]. Um GSE, basicamente, recebe um conjunto de palavras-chave e, analisando apenas o texto não estruturado, gera uma lista de páginas que contêm essas palavras.

`Objetos da web` são unidades de dados sobre as quais informações da web são coletadas, indexadas e ordenadas. Esses objetos são conceitos usualmente reconhecidos (como autores ou conferências), relevantes a um domínio de aplicação e que podem ser representados por um conjunto de atributos, os quais dependem do domínio do objeto [Nie et al. 2007]. `Páginas-objeto` são páginas que descrevem um único objeto inerente na web. Isso significa que páginas que listam diversos objetos não são consideradas páginas-objeto por não representarem um objeto em particular. A busca por páginas-objeto é feita através de consultas restringidas por atributos de domínio e pode ser chamada de `busca-objeto` [Pham et al. 2010]. Um exemplo desse tipo de consulta é "*professor de banco de dados da UFRGS*", que restringe a área e a instituição de atuação de um objeto professor e tem como objetivo recuperar páginas (pessoais, institucionais, de currículo, etc.) que descrevam esse objeto.

Embora os GSEs consigam atender à maioria das consultas realizadas atualmente, eles se mostram inadequados para recuperar páginas-objeto [Pham et al. 2010]. Dentre as limitações do processo de busca convencional que podem estar relacionadas a esse problema, encontra-se a ambiguidade das palavras-chave. Por exemplo, mesmo que o objetivo de uma busca com a palavra-chave "*Paris*" seja encontrar páginas relacionadas à cidade capital da França, muitas páginas relacionadas ao primeiro nome de *"Paris Hilton"* serão retornadas [Miklós 2010].

Este artigo propõe um novo método para a identificação e a busca de páginas-objeto, denominado OPIS (acrônimo para ***O**bject **P**age **I**dentifying and **S**earching*), que adota realimentação de relevância e técnicas de pré-processamento de texto e de aprendizagem de máquina na classificação baseada em conteúdo de páginas web. O OPIS envolve a integração de um classificador a um motor de busca, de modo que os resultados recuperados pelo motor de busca sejam filtrados, permitindo que somente as páginas identificadas (classificadas) como páginas-objeto sejam apresentadas aos usuários. A principal contribuição deste método é a melhoria na precisão de buscas-objeto, permitindo que usuários finais encontrem resultados que melhor atendam a suas necessidades de informação.

O OPIS foi avaliado através de experimentos preliminares no domínio real de pesquisadores. Os resultados mostram que o OPIS superou o motor de busca convencional com aumentos de precisão de 112% (0,144 vs. 0,068) nos primeiros cinco resultados e de 37% (0.157 vs. 0.115) nos primeiros 20.

O restante desse artigo está organizado da seguinte forma. Na Seção 2 são apresentados trabalhos relacionados. Na Seção 3, o OPIS é especificado detalhadamente. Na Seção 4 é apresentada a implementação e a experimentação do OPIS no domínio de pesquisadores. Na Seção 5, o artigo é concluído e direções futuras são apontadas.

2. Trabalhos Relacionados

Muitos esforços têm sido feitos para melhorar os resultados recuperados pelos GSEs. A maioria deles considera que os motores de busca são independentes de domínio e, em geral, usam a mesma função de *ranking* para todas as páginas, ou seja, os GSEs não consideram as particularidades de cada domínio. O OPIS propõe uma nova solução nesse sentido. A seguir, são apresentados e comparados os principais trabalhos relacionados à busca-objeto e tópicos relacionados.

Motores de busca vertical [Ji et al. 2009][Lee et al. 2011][Luo 2009], que são abordagens para criar motores de busca específicos a determinados domínios, relacionam-se à busca-objeto por também restringirem a busca à um domínio específico. Além disso, outros trabalhos [Bennett et al. 2010][Geng et al. 2009][Pham et al. 2010] usam funções de *ranking* específicas para recuperar páginas relacionadas a domínios específicos. Em geral, esses trabalhos usam técnicas de processamento de texto e aprendizagem de máquina para extrair o conteúdo das páginas e, com base nisso: aprender um determinado domínio e filtrar apenas páginas consideradas pertencentes a esse domínio no processo de coleta; ou determinar as características do domínio a serem usadas em uma função de *ranking* específica. O OPIS se difere desses trabalhos à medida que não deseja aprender apenas o tópico das páginas, mas também seu tipo funcional (se a página é ou não uma página-objeto).

Blanco et al. (2008) propõem um método para coletar automaticamente páginas da web que publicam dados relacionados às instâncias de entidades conceituais com um esquema implícito. Esse método assume que o usuário fornece exemplos de páginas de entidades a partir de sites distintos e percorre cada um desses sites procurando por páginas que apresentam *templates* e caminhos similares aos respectivos exemplos. Esse trabalho difere do OPIS por focar na coleta de páginas de entidades com *templates* similares, enquanto o OPIS busca identificar páginas-objeto, sem considerar *templates* específicos, para melhorar a busca-objeto. Também com relação à coleta de páginas, Assis (2008) propõe um coletor focado para tópicos de interesse que possam ser representados por características de gênero e de conteúdo. Quando o usuário deseja buscar por páginas de planos de ensino de disciplinas de banco de dados, por exemplo, um conjunto de características (termos) que descreva o gênero (planos de ensino) e outro que descreva o conteúdo (banco de dados) devem ser informados por um usuário especializado, de modo que o coletor possa analisar cada página através da sua similaridade com os termos de ambos os aspectos. No OPIS, o conteúdo e o gênero das páginas não são considerados separadamente, o que reduz o nível de especialidade do usuário, uma vez que ele não precisa discernir entre esses dois aspectos e nem selecionar termos manualmente para caracterizá-los.

Para Pham et al. (2010), cuja proposta esta mais próxima do OPIS, o problema de busca-objeto se assemelha ao de aprendizagem de *ranking*, em que o principal objetivo é aprender uma função de *ranking* através de uma função de aprendizagem, com base em um conjunto de características relevantes. A solução proposta consiste em desenvolver diversos motores de busca vertical para suportar a busca por páginas-objeto em diferentes domínios. Para isso, uma função de *ranking* deve ser aprendida para cada

domínio específico. O desenvolvedor[1] deve submeter consultas por palavras-chave e anotar um conjunto de treinamento a partir das páginas recuperadas. Esse conjunto de treinamento tem suas características extraídas automaticamente e usadas em uma função de aprendizagem. O OPIS também adota o conteúdo das páginas para melhorar a busca-objeto através da classificação funcional (páginas-objeto ou não) das páginas. Porém, ele não considera a busca-objeto como um problema de aprendizagem de *ranking*. Ao invés disso, o processo de *ranking* fica a cargo do motor de busca ao qual acoplamos o método. Isso torna desnecessária a análise das informações estruturadas embutidas nas páginas, durante o processo de busca, para casá-las com as características que integram a função de *ranking* aprendida (por exemplo, "a palavra professor aparece no título").

3. OPIS: *Object Page Identifying and Searching*

O OPIS (acrônimo para ***O**bject **P**age **I**dentifying and **S**earching*) é um método que visa a identificação e a busca de páginas-objeto. OPIS permite que somente páginas identificadas como páginas-objeto, para um determinado domínio, sejam recuperadas pelas consultas dos usuários, o que torna os resultados de buscas-objeto mais precisos e, dessa forma, mais adequados às necessidades dos usuários. O método caracteriza-se por adotar realimentação de relevância e técnicas de pré-processamento de texto e de aprendizagem de máquina na construção, baseada no conteúdo de páginas web, de um classificador, que será responsável pela identificação das páginas-objeto. Esse classificador é, então, integrado a um motor de busca, adicionando uma etapa de filtragem (classificação) ao processo de busca convencional.

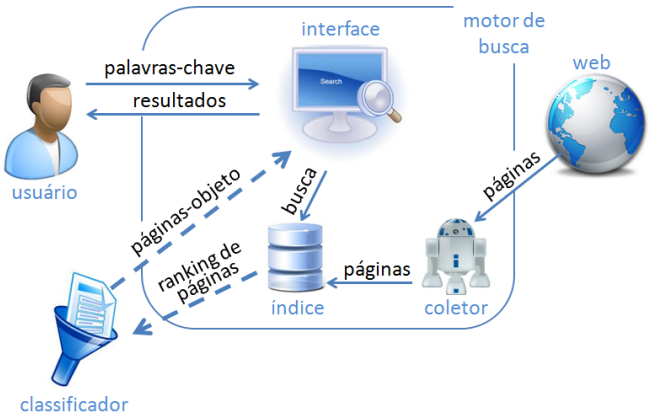


Figura 1. Visão geral do OPIS.

Na Figura 1 é apresentada uma visão geral do OPIS, mostrando a integração das atividades de identificação e busca. Note que o usuário submete, através da interface, uma consulta por palavras-chave. Essa consulta é executada sobre o índice e obtém como resposta o *ranking* das páginas nas quais os termos da consulta foram encontrados, seguindo, até então, o processo de busca convencional. A diferença introduzida pelo OPIS (ilustrada com seguimentos tracejados) esta no fato de que esse *ranking* de páginas não é apresentado diretamente ao usuário, passando antes por uma

[1] Pessoa responsável por guiar o treinamento da função de *ranking* para um domínio específico, permitindo que usuários possam, então, submeter consultas relacionadas a esse domínio.

atividade de filtragem adicional, na qual essas páginas passam por um processo de classificação e somente as classificadas como páginas-objeto são apresentadas para o usuário.

O classificador é o cerne do OPIS, pois é o responsável pela tarefa de identificação das páginas-objeto, da qual depende a filtragem e, dessa forma, os resultados da busca. A construção de um classificador depende, além da escolha e parametrização de um algoritmo de classificação, de um conjunto de páginas de treinamento do domínio de interesse, que é a base do processo de aprendizagem do algoritmo. Para obter esse conjunto, o OPIS faz uso de Realimentação de Relevância [Baeza-Yates e Ribeiro-Neto 2011], na qual o usuário avalia a relevância de um conjunto de páginas e este passa a ser considerado como base do treinamento. O classificador produzido está intrinsecamente relacionado com o usuário em questão, uma vez que será este quem definirá a base de treinamento a ser usada na geração do modelo de classificação. Dessa forma, a corretude da coleção de treinamento e, consequentemente, do classificador, está condicionada à correta avaliação do usuário quanto ao que é ou não uma página-objeto, devendo, assim, esse usuário ter conhecimento prévio sobre o domínio a ser treinado.

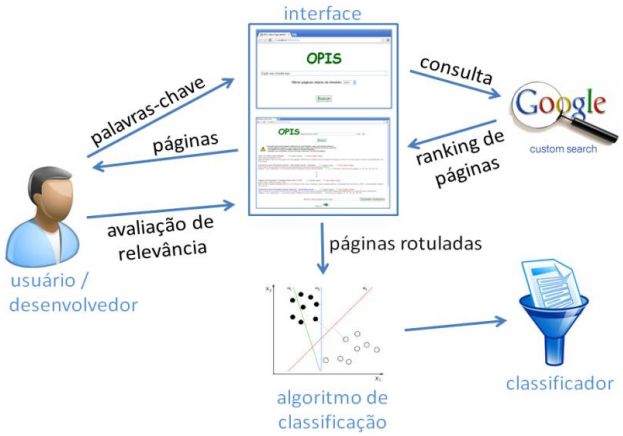


Figura 2. Construção do classificador, apoiada pela realimentação de relevância.

Inicialmente, o usuário deve estabelecer quais tipos de páginas-objeto podem existir no domínio a ser considerado (no exemplo do domínio de pesquisadores: páginas institucionais, pessoais, de currículo, etc. que descrevam um objeto pesquisador). Com base nisso, uma ou mais consultas por palavras-chave, visando recuperar essas páginas, devem ser criadas (como "*homepage professor doutor*"). O processo de realimentação de relevância é ilustrado na Figura 2. Após submeter uma dessas consultas, o usuário pode avaliar a relevância dos resultados recuperados, indicando páginas que tenham sido ou não consideradas páginas-objeto para o domínio em treinamento. Esse procedimento pode ser repetido com mais consultas, atualizando a coleção de treinamento e, consequentemente, o classificador, até que o usuário julgue necessário. Isso significa que o processo continua até que o usuário considere ter rotulado páginas suficientes para exemplificar tanto páginas não objeto quanto os principais tipos de páginas-objeto estabelecidos inicialmente.

4. Implementação e Experimentos

Uma interface web foi desenvolvida para suportar a tarefa de realimentação de relevância, permitindo que o usuário submeta suas consultas e possa avaliar a relevância das páginas resultantes, por meio de caixas de seleção apresentadas ao lado de cada resultado. A API *Google Custom Search* [Google 2012], sem nenhuma customização, foi usada para recuperar e ordenar os resultados das consultas submetidas pelo usuário. Mesmo não tendo sido mostrado na Figura 2, antes de serem usadas no treinamento do algoritmo de classificação, as páginas rotuladas passam por atividades de pré-processamento (como remoção de *tags* HTML, tradução de termos estrangeiros e remoção de *stopwords*) e são representadas através do Modelo de Espaço Vetorial, considerando TF-IDF [Baeza-Yates e Ribeiro-Neto 2011] como forma de ponderação. Na classificação, foi utilizado o algoritmo *LibSVM*, através da biblioteca de mineração *Weka* [Universidade de Waikato 2013], com núcleo linear (por ser de rápida execução).

Para a realização de experimentos, foi considerado um classificador construído, de acordo com o processo explicado na Seção 3, para o domínio de pesquisadores. Foram rotulados 10 exemplos de páginas-objeto (incluindo páginas institucionais, pessoais ou de currículo, que representassem um objeto pesquisador) e 10 exemplos negativos (incluindo páginas relacionadas a concursos, corpo docente e notícias), que serviram como base para o aprendizado do algoritmo. O classificador foi integrado à API *Google Custom Search* (sem nenhuma customização), usada como motor de busca convencional. Os experimentos contaram com a participação de 10 usuários, que criaram cinco consultas, para o domínio de pesquisadores, cada. Os usuários também especificaram objetivos e critérios de relevância para cada consulta, que guiaram a avaliação de relevância dos resultados recuperados.

Tabela 1. Resultados com a média das 50 consultas criadas pelos usuários.

	p@5	p@10	p@15	p@20
Google	0.068	0.094	0.108	0.115
OPIS	0.144	0.166	0.155	0.157

Na Tabela 1 são apresentados os resultados obtidos a partir da submissão e avaliação das 50 consultas criadas pelos usuários, considerando o Google (sem filtragem) e o OPIS (com a filtragem dos resultados recuperados pelo Google). A métrica de precisão em *n* (p@*n*), que considera somente os primeiros *n* resultados recuperados pelo sistema, foi usada com *n* de 5, 10, 15, e 20 para medir a precisão dos resultados apresentados e ter um indicativo das variações de precisão em relação ao posicionamento no *ranking*. Pode-se notar que o OPIS obteve melhores resultados em todas as precisões, considerando a média das consultas realizadas, tendo essa melhoria variado de 112% (p@5) a 37% (p@20). Isso significa que a filtragem realizada pelo OPIS permitiu que páginas-objeto relevantes para a consulta do usuário substituíssem páginas classificadas como não objeto no *ranking* das páginas recuperadas, apresentando ao usuário resultados que melhor atendam a sua necessidade de informação para buscas-objeto.

5. Considerações Finais e Trabalhos Futuros

Neste artigo foi introduzido o problema da busca-objeto em motores de busca convencional e proposto um novo método, chamado OPIS, para a identificação e a busca

de páginas-objeto. O OPIS usa realimentação de relevância e técnicas de pré-processamento de texto e de aprendizagem de máquina na construção, baseada no conteúdo de páginas web, de um classificador. A integração desse classificador a um motor de busca convencional permite a filtragem dos resultados de busca-objeto, fazendo com que apenas páginas classificadas como páginas-objeto sejam apresentadas aos usuários. Experimentos preliminares mostraram que o OPIS proporcionou um ganho de 37% de precisão, considerando as 20 primeiras páginas recuperadas pelas buscas-objeto, em relação a um motor de busca convencional.

Como trabalhos futuros, pretende-se: (i) incorporar expansão de consultas ao OPIS, para melhorar a precisão dos resultados; (ii) testar o método em um domínio adicional, a fim de atribuir maior confiabilidade à validação; e (iii) usar como *baseline* a abordagem de Pham et al.(2010), apresentada nos trabalhos relacionados.

Referências

Assis, G. T. (2008) Uma Abordagem Baseada em Gênero para Coleta Temática de Páginas da Web. Tese (Doutorado em Ciência da Computação) - Instituto de Ciências Exatas, Universidade Federal de Minas Gerais.

Baeza-Yates, R. e Ribeiro-Neto, B. (2011), Modern Information Retrieval: The Concepts and Technology behind Search. Addison Wesley, 2ª edição.

Bennett, P. N., Syore, K. e Dumais, S. T. (2010) Classification-Enhanced Ranking. In: 19th International Conference on World Wide Web, p. 111-120.

Blanco, L., Crescenzi, V., Merialdo, P. e Papotti, P. (2008) Supporting the Automatic Construction of Entity Aware Search Engines. In: 10th ACM Workshop on Web Information and Data Management, p. 149-156.

Geng, B., Yang, L., Xu, C. e Hua, X. (2009) Ranking Model Adaptation for Domain-Specific Search. In: 18th Conference on Information and Knowledge Management, p. 197-206.

Google (2013) "Google Custom Search", https://developers.google.com/custom-search, Junho.

Ji, L., Yan, J., Liu, N., Zhang, W., Fan, W. e Chen, Z. (2009) ExSearch: A Novel Vertical Search Engine for Online Barter Business. In: 18th Conference on Information and Knowledge Management, p. 1357-1366.

Lee, H., Nazareno, F., Jung, S. e Cho, W. (2011) A Vertical Search Engine for School Information Based on Heritrix and Lucene. In: 5th International Conference on Convergence and Hybrid Information Technology, p. 344-351.

Luo, G. (2009) Design and Evaluation of the iMed Intelligent Medical Search Engine. In: IEEE International Conference on Data Engineering, p.1379-1390.

Miklós, Z. (2010) From Web Data to Entities and Back. In: 22nd International Conference on Advanced Information Systems Engineering, p. 302-316.

Nie, Z., Ma, Y., Shi, S., Wen, J. e Ma, W. (2007) Web Object Retrieval. In: 16th International Conference on World Wide Web, p. 81-90.

Pham, K. C., Rizzolo, N., Small, K., Chang, K. C. e Roth, D. (2010) Object Search: Supporting Structured Queries in Web Search Engines. In: NAACL HTL 2010 Workshop on Semantic Search, p. 44-52.

Universidade de Waikato (2013), "Weka Data Mining Software API", http://www.cs.waikato.ac.nz/ml/weka, Junho.

Banco de Dados em Nuvem: Um Modelo para Garantia de Consistência dos Dados

Elyda Laisa S. X. Freitas, Fernando da Fonseca de Souza

Centro de Informática (CIn) - Universidade Federal de Pernambuco (UFPE)

Recife – Pernambuco – Brasil

{elsx,fdfd}@cin.ufpe.br

Nível: Mestrado

Ano de Ingresso no programa: 2012

Época esperada de conclusão: Março de 2014

Abstract. *This paper presents a work-in-progress focusing on a data consistency model, which uses the user's knowledge about the application to define which data needs strong consistency guarantees and which data does not need. Data with strong consistency guarantees are handled by eager update. In this case, an adapted architecture to cloud computing, that uses group communication, was designed. Data that do not require strong consistency can be treated with eventual consistency techniques.*

Resumo. *Este artigo apresenta o trabalho em andamento com foco em um modelo de consistência de dados, o qual se utiliza do conhecimento do usuário sobre a aplicação para definir quais dados necessitam de garantia de consistência forte e quais não. Os dados com garantia de consistência são tratados por meio da atualização ansiosa. Neste caso, uma arquitetura adaptada à nuvem, a qual se utiliza de comunicação em grupo, foi projetada. Para dados que não necessitam de consistência forte, técnicas de consistência eventual poderão ser utilizadas.*

Palavras-Chave

Banco de Dados como Serviço, Consistência de Dados, Banco de Dados em Nuvem.

1. Introdução e Motivação

Nos últimos anos, uma área de estudo de Tecnologia da Informação (TI) tem se destacado: a Computação em Nuvem, a qual tem por objetivo prover serviços de TI sob demanda aos usuários. O Sistema de Gerenciamento de Banco de Dados (SGBD) é um dos muitos serviços que podem ser disponibilizados na nuvem. Conhecido como DBaaS, do inglês *Database as a Service* (Banco de Dados como Serviço), esse novo paradigma tem recebido a atenção tanto da academia quanto de grandes empresas, como é o caso da Amazon, Oracle e Google (com os produtos Amazon S3, Oracle 12C e Bigtable, respectivamente).

Um dos problemas encontrados nos SGBD tradicionais foi "*a falta de suporte para particionamento dinâmico eficiente de dados, o que limitou a escalabilidade e a utilização de recursos*" (ELMASRI e NAVATHE, 2011).

No DBaaS, a distribuição, uma das características inerentes à nuvem, permite ao banco de dados armazenar grandes volumes de dados e crescer quase indefinidamente – aos limites da capacidade de armazenamento do *datacenter* hospedeiro. Por outro lado, essa característica impõe ao sistema de banco de dados algumas restrições. Conforme provado por Gilbert e Lynch (2002), não é possível atingir em um sistema distribuído, e ao mesmo tempo, as desejáveis características de Consistência, Disponibilidade e Tolerância à Partição (em caso de falha). Tal proposição é conhecida como Teorema CAP.

De acordo com Elmasri e Navathe (2011), para a preservação da consistência uma transação deve levar o banco de um estado consistente a outro. No entanto, diversos sistemas de bancos de dados em nuvem têm optado por relaxar a garantia de consistência, dando prioridade à disponibilidade do serviço (ZHAO et. al, 2012).

Para alguns tipos de aplicações, tal opção é perfeitamente válida, uma vez que não haja dados críticos armazenados. Por exemplo, não há maiores dificuldades se uma publicação não for imediatamente visualizada por todos os amigos de um usuário de uma rede social. No entanto, para grande parte das aplicações, inconsistências nos dados podem levar a transtornos imensuráveis, como no caso de uma aplicação bancária com valores equivocados. Desse modo, verifica-se que, em função do relaxamento da consistência, diversos sistemas de banco de dados em nuvem não estão aptos a receber todos os tipos de aplicações.

Entretanto, o relaxamento da consistência não é a única possibilidade: Dez anos após a concepção do Teorema CAP, Brewer (2012) afirma, em seu trabalho intitulado "*CAP Twelve Years Later: How the 'Rules' Have Changed*" (CAP Doze Anos Depois: Como as "Regras" Mudaram), que a compreensão desse teorema foi excessivamente simplificada, levando os projetistas a escolherem indiscriminadamente duas das proposições (como, por exemplo, tolerância à partição e disponibilidade; ou consistência e disponibilidade, apenas). Brewer (2012) recomenda, ainda, que a consistência não seja "cegamente" sacrificada.

Diante do exposto, torna-se perceptível a necessidade de explorar as diferentes nuances da consistência de um Banco de Dados em Nuvem, ao invés de simplesmente relaxar a consistência. Sendo assim, o seguinte problema de pesquisa foi formulado:

Como garantir a consistência dos dados armazenados em bancos de dados em nuvem a fim de permitir a utilização do modelo de DBaaS tanto por aplicações que necessitam de garantia de consistência quanto por aquelas que não necessitam?

2. Fundamentação Teórica

De acordo com a Oracle (2011), DBaaS "*É uma abordagem arquitetural e operacional que permite aos provedores de TI entregar funcionalidades de banco de dados como um serviço para um ou mais consumidores*". Isso significa dizer que no modelo DBaaS, o serviço oferecido pelo provedor são funcionalidades cotidianas dos SGBD, a saber: *backup* e recuperação de dados, entre outros. Um dos aspectos importantes na utilização do DBaaS é o modo como a consistência de dados é tratada. De acordo com Özsu e Valduriez (2011), "*Um banco de dados replicado está em um estado mutuamente consistente se todas as réplicas de cada um dos seus elementos de dados têm valores idênticos*".

Um método frequentemente utilizado para a distribuição nos sistemas de banco de dados em nuvem é a replicação, isto é, a cópia dos dados e armazenamento em mais de um local. Outra técnica de distribuição presente no DBaaS é o particionamento de dados. De acordo com Rahimi e Haug (2010), particionamento "*quebra uma tabela em dois ou mais pedaços chamados fragmentos ou partições e permite o armazenamento desses pedaços em diferentes locais*". Em geral, o particionamento de dados é realizado manualmente, sob responsabilidade do administrador do banco de dados. Na nuvem, o particionamento é dinâmico, realizado em tempo real.

Diversas abordagens podem ser utilizadas para replicar um banco de dados. Para que a replicação de dados ocorra de modo a garantir a consistência, pode-se utilizar a atualização ansiosa. Nessa abordagem, todas as atualizações são realizadas no contexto da transação. Consequentemente, quando a transação é confirmada, todas as réplicas possuem o mesmo valor (ÖZSU e VALDURIEZ, 2011).

Esse modelo de atualização geralmente utiliza a estratégia de *commit* em duas fases (*two-phase commit* - 2PC), que divide a execução do *commit* em duas etapas: na primeira, o coordenador envia um aviso de 'preparar para *commit*' a todos os servidores envolvidos na transação; na segunda fase, o coordenador pode enviar a mensagem 'realizar o *commit*' (caso tenha recebido todas as respostas positivas) ou abortar a transação. A atualização ansiosa, entretanto, pode causar problemas de desempenho, uma vez que durante a execução da transação, o dado a ser atualizado fica bloqueado (RAHIMI e HAUG, 2010).

Por outro lado, se não há necessidade de garantir a consistência dos dados de modo imediato, pode-se utilizar a abordagem de atualização preguiçosa. Nesse modelo, a atualização é realizada em uma réplica e repassada às outras réplicas de modo assíncrono. Como Gao e Diao (2010) afirmam: "*algoritmos de propagação preguiçosa postam as atualizações para as réplicas por meio de transações independentes, após a confirmação da transação de atualização no local de origem*". Deste modo, a atualização dos dados é um processo separado da propagação.

3. Caracterização da Contribuição

O presente trabalho parte do pressuposto de que uma mesma aplicação pode abrigar dados que necessitam de garantia de consistência e outros que não necessitam. Por exemplo, em um sistema de compra de passagens aéreas, a informação referente à quantidade de vagas disponíveis em determinado voo deve ser tratada de modo a garantir que todos os usuários visualizem a mesma informação, a fim de que não haja a venda de passagens excedentes. Nessa mesma aplicação, no entanto, a foto do cliente, parte do seu cadastro, não precisa ser imediatamente atualizada em todas as réplicas.

Para determinar o modo de tratamento dos dados, utiliza-se o conhecimento do usuário sobre sua aplicação: ele deve definir quais das tabelas devem ser tratadas com garantia de consistência e quais podem ter a garantia relaxada. No tocante aos dados com garantia de consistência, foi concebido um método diferente dos utilizados atualmente, conforme será mostrado na seção seguinte. Para isso, descartou-se o uso do protocolo 2PC, em benefício do desempenho.

Em seu lugar, há um modelo de comunicação em grupo, do tipo *publish-subscribe*, no qual "*assinantes (subscribers) registram seu interesse em um evento, ou um padrão de eventos, e são subsequentemente notificados, de forma assíncrona, de eventos gerados por aqueles que publicam (publishers)*" (EUGSTER et. al, 2003). No modelo proposto, após a definição de qual tabela deverá ser tratada com garantia de consistência, estas devem ser dispostas em réplicas da Camada 1, as quais se inscrevem para receber as atualizações desses dados, conforme mostra a Figura 1.

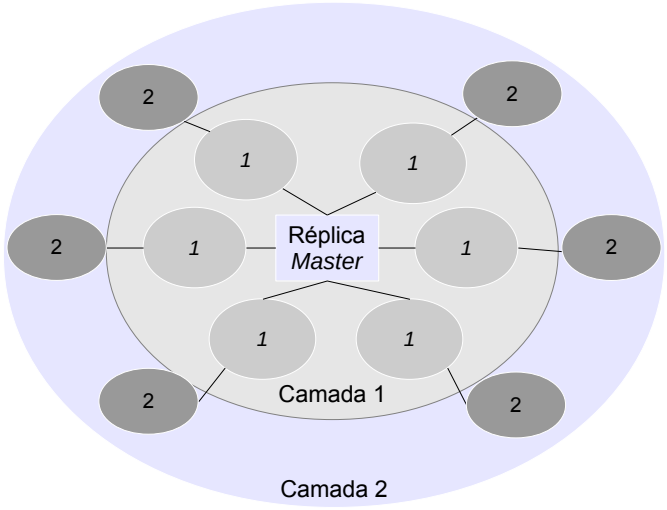


Figura 1. Modelo de replicação para garantia de consistência dos dados da Camada 1 e consistência eventual da Camada 2.

Conforme se pode verificar na imagem, há um único mestre, para o qual devem ser direcionadas as requisições. Quando a transação é confirmada no mestre, os *subscribers* são notificados e deverão aplicá-las em seus bancos de dados. Todos os *subscribers* deverão aplicar a atualização, a fim de que a consistência seja garantida. Uma vez que as atualizações podem ser realizadas em paralelo, espera-se uma melhoria de desempenho com relação aos modelos que se utilizam do *commit* em duas fases. Ademais, não há a utilização de bloqueios.

Para que não haja esperas indefinidas, devem ser estabelecidos prazos máximos de espera. Nestes casos, as réplicas não atualizadas deverão ficar indisponíveis para leitura até que sejam recuperadas e as atualizações aplicadas. Deste modo, mesmo que uma ou mais réplicas falhem e não sejam atualizadas, a consistência continua garantida. Além disso, consultas nas réplicas já atualizadas são imediatamente liberadas.

Para dar suporte à característica de particionamento dinâmico de dados, a seguinte regra deverá ser seguida: dados consistentes deverão ser particionados apenas em réplicas gerenciadas pela réplica *master,* de modo a garantir a consistência de dados. Ademais, todas as transações de leitura nas quais houver dados marcados com necessidade de consistência devem ser direcionadas às réplicas gerenciadas pelo *master* (Camada 1).

Em suma, a réplica *master* tem as seguintes funções: (1) receber as transações e aplicá-las normalmente; (2) capturar as transações aplicadas e preparar a mensagem; (3) armazenar as mensagens; e (4) gerenciar as réplicas e suas subscrições aos tópicos de interesse. Os tópicos são estruturas de dados criadas para armazenar as mensagens.

O *Provider* é o responsável por capturar as transações já confirmadas. Tal ação é realizada a partir do log da réplica *master*. Observe-se que a confirmação da transação na réplica *master* garante a verificação de restrições de consistência e eventuais dependências que possam existir no banco de dados, uma vez que não há mudanças nas funções-padrão do SGBD utilizado. Desse modo, essas transações podem ser executadas nas outras réplicas, mantendo tais benefícios, ao seguir a ordenação total.

Após capturar a transação, o *middleware* da réplica *master* deverá verificar o destino da mesma. Caso a mensagem seja destinada a uma réplica com dados que não necessitam de garantia de consistência, esta deverá apenas ser direcionada para as réplicas da Camada 2. Caso o destino seja as réplicas com dados que necessitam de garantia de consistência, a mensagem deverá ser armazenada no tópico.

No modelo *publish/subscribe* cada tópico possui um assunto e as réplicas subscrevem-se nos tópicos de acordo com o assunto de seu interesse (EUGSTER et. al, 2003). No modelo proposto, cada tópico deverá conter dados de uma tabela específica. Por exemplo, o tópico denominado “tópicoX” deverá conter todas as mensagens destinadas à tabela “passageiros”. As réplicas da Camada 1 deverão, portanto, subscrever-se nos tópicos com dados de suas respectivas tabelas. O responsável por coordenar os tópicos e suas particularidades é o TopicManager.

Em resumo, os elementos da arquitetura interagem do seguinte modo: 1) O cliente envia as mensagens (transações); 2) A réplica *master* recebe e aplica normalmente as transações; 3) O *Provider*, então, captura as mensagens e as organiza, adicionando dados como o destino, tempo de chegada e outras, caso sejam destinadas à Camada 1; ou as redireciona à Camada 2; 4) Através do *TopicManager*, o *Provider*, armazena a mensagem recebida no devido tópico; 5) O *TopicManager* devolve ao *Provider* a informação sobre qual o tópico onde a mensagem foi armazenada; e 6) Uma vez que a chegada das mensagens nos tópicos está em constante monitoramento por parte dos *subscribers*, estes deverão, neste momento, retirar a mensagem recebida.

Os dados que não necessitam de garantia de consistência (Camada 2), por sua vez, poderão ser tratados por meio da consistência eventual, que garante apenas que os

dados serão tornados consistentes eventualmente (ISLAM et. al, 2012). Nesse caso, técnicas consagradas poderão ser utilizadas sem alterações, uma vez que as mesmas já são largamente utilizadas no ambiente em nuvem. O trabalho de Vogels (2009) mostra diferentes variações da consistência eventual, as quais podem ser utilizadas no presente modelo. Para validar o modelo proposto, será desenvolvido um protótipo.

4. Trabalhos Relacionados

No que se refere à garantia de consistência em sistemas de banco de dados em nuvem, podem-se elencar diferentes trabalhos que tratam do referido tema. Um desses trabalhos é o de Wang et. al. (2010), o qual divide a estratégia de consistência em quatro categorias (que vai da consistência forte - isto é, o usuário sempre irá ler a versão mais atual do dado - à fraca, que não fornece essa garantia), de acordo com a frequência de leitura e escrita. Verifica-se, no entanto, que, ao contrário do presente trabalho, não há nenhuma participação do usuário no nível de consistência com o qual os dados devem ser tratados. Desse modo, este conhecimento sobre a aplicação é ignorado.

Outro trabalho relevante nesse contexto é o de Zhao et. al (2012), o qual apresenta um *framework* para replicação de banco de dados no ambiente em nuvem, onde o usuário deverá definir um SLA de atualidade dos dados para cada réplica. Deve-se observar, no entanto, que o modelo apresentado acima trata apenas da atualidade dos dados e não garante a consistência forte, uma vez que o SLA definido pelo usuário precisa ser violado para que o módulo de ação entre em atividade. Ademais, o processo de definição do SLA pode ser bastante dificultoso, já que não há nenhum suporte, por parte do sistema, a fim de auxiliar o usuário nesta tarefa.

Outro trabalho nesse contexto é o Harmony, apresentado no artigo de Chihoub et. al (2012). O Harmony tem por objetivo ajustar o nível de consistência adaptativamente de acordo com os requisitos da aplicação definidos pelo usuário (de 0 a 100% de taxa de leitura obsoleta aceitável) e do estado do sistema de armazenamento. Testes apresentados por Chihoub et. al (2012) mostram que o Harmony reduz a taxa de leitura de dados obsoletos, no entanto, o mesmo não garante a consistência forte dos dados.

Os trabalhos citados serviram de inspiração para este no modo como o usuário interfere diretamente no modelo, definindo os níveis de consistência desejados. No presente trabalho, o administrador de dados define as necessidades em um nível de granularidade mais baixo, que alcança a tabela na qual se deseja a garantia de consistência. Como o usuário conhece a aplicação, então a escolha das tabelas que deverão garantir a consistência torna-se uma tarefa mais simples.

5. Avaliação dos Resultados e Estado Atual do Trabalho

Espera-se, ao final deste trabalho, contribuir com a academia mediante a utilização de abordagens de replicação e propagação de dados reconhecidas e largamente estudadas (atualização ansiosa e preguiçosa) em um ambiente diferente daquele proposto inicialmente pelas mesmas, realizando as devidas adaptações para a nuvem.

Adicionalmente, o modelo de consistência desenvolvido explora os diferentes níveis da consistência de dados, conforme proposto por Brewer (2012). Atualmente, a revisão de literatura foi finalizada. O SGBD escolhido para o desenvolvimento do

protótipo foi o Oracle 12C, o qual permite o *download* do sistema, facilitando a montagem do ambiente controlado a ser utilizado para os testes. Por fim, será realizada uma pesquisa qualitativa, a fim de avaliar a adequação do protótipo desenvolvido.

Referências

Brewer, E. (2012) "CAP Twelve Years Later: How the 'Rules' Have Changed". Computer, vol.45, no.2, p.23-29, Feb. 2012.

Chihoub, H., Ibrahim, S., Antoniu, G. and Pérez, M. S. (2012) "Harmony: Towards Automated Self-Adaptive Consistency in Cloud Storage". In: 2012 IEEE International Conference on Cluster Computing (CLUSTER), p.293-301, 24-28 Sept. 2012.

Elmasri, Ramez and Navathe, Shamkant B. (2011) "Sistemas de Banco de Dados". 6ª Edição. Pearson Addison Wesley. São Paulo, 2011

Eugster, P. Th., Felber, P. A., Guerraoui, R. and Kermarrec, A. (2003) "The many faces of publish/subscribe". In: ACM Comput. Surv. 35, 2 (June 2003), p. 114-131.

Gao, A. and Diao, L. (2010) "Lazy update propagation for data replication in cloud computing". In: 5th International Conference on Pervasive Computing and Applications (ICPCA), p.250-254, 1-3 Dec. 2010.

Gilbert, S. and Lynch, N. (2002) "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services". SIGACT News, vol.33, no., p.51–59, 2002.

Islam, M.A., Vrbsky, S.V. and Hoque, M.A. (2012) "Performance analysis of a tree-based consistency approach for cloud databases". In: International Conference on Computing, Networking and Communications (ICNC), p.39-44, Jan. 30 2012-Feb. 2 2012.

Oracle. (2012) "Database as a Service: Reference Architecture – An Overview. An Oracle White Paper on Enterprise Architecture". September 2011. http://www.oracle.com/technetwork/topics/entarch/oes-refarch-dbaas-508111.pdf. June 2013.

Özsu, M. T. and Valduriez, P. (2011) "Principles of Distributed Database Systems". Springer; 3rd ed. 2011 edition (March 2, 2011).

Rahimi, S. K. and Haug, F. S. (2010) "Distributed Database Management Systems: A Practical Approach". Wiley-IEEE Computer Society Pr; 1 edition (August 2, 2010).

Vogels, W. (2009) "Eventually consistent". Commun. ACM, vol. 52, pp. 40–44, January 2009.

Wang, X., Yang, S., Wang, S., Niu, X. and Xu, J. (2010) "An Application-Based Adaptive Replica Consistency for Cloud Storage". 9th International Conference on Grid and Cooperative Computing (GCC), p.13-17, 1-5 Nov. 2010

Zhao, L., SAKR, S. and LIU, A. (2012) "Application-Managed Replication Controller for Cloud-Hosted Databases". In: IEEE Fifth International Conference on Cloud Computing (CLOUD), p. 922-929. Honolulu: 24-29 June 2012.

Correlation between the quality of focused crawlers and the linguistic resources obtained from them

Abstract. *Focused web crawlers have been used for the automatic acquisition of lexical resources for particular domains, gathering websites related to a set of topics of interest. For this purpose, a portion of the web graph is traversed, and the documents corresponding to pages considered relevant are stored and treated as a corpus. It is important to traverse this graph in a targeted way, organizing pages in a queue that prioritizes pages that are more likely to be relevant. Texts collected by these tools can be used to train domain-specific machine translation (MT) systems. In this work, we compare the performance of focused crawling algorithms, measured with standard metrics, and the quality of the linguistic resources obtained, in order to try to establish a correlation between both. Also, we propose a novel, extrinsic metric to evaluate the efficiency of a focused crawling algorithm.*

Palavras-chave: focused crawling, lexical resources, machine translation, correlation

Aluno: Bruno Rezende Laranjeira (bruno.rezendelaranjeira@inf.ufrgs.br)

Orientadora: Viviane Pereira Moreira (viviane@inf.ufrgs.br)

Co-orientadora: Aline Villavicencio (avillavicencio@inf.ufrgs.br)

Nível: Mestrado

Programa de Pós-Graduação em Computação

Instituto de Informática - Universidade Federal do Rio Grande do Sul

Ingresso em: Primeiro semestre de 2012

Época esperada de conclusão: Segundo semestre de 2013

Etapas concluídas: Foi construída e testada uma ferramenta extensível para coleta de documentos na web. Alguns algoritmos de coleta focada também já foram implementados. Procedimentos para a continuação dos experimentos já estão definidos

Etapas futuras: Finalizar a implementação de outros algoritmos de coleta focada. Treinamento do sistema de tradução automática. Avaliação das traduções geradas. Análise da correlação entre as duas medidas.

1. Introduction

Web crawlers [Liu 2009] are used by web search engines, like Google or Yahoo, to collect documents and use them to construct their indexes. To do this, they traverse the web graph, fetching pages and storing the information necessary for the intended task (e.g. text, hyperlinks, figures, etc.). Crawlers start collecting an initial set of seed pages and keep all the outlinks contained in them in a queue known as *URL frontier*. This process continues recursively, visiting pages contained in the URL frontier, whose links are extracted and added to the frontier, until the frontier is empty or a stopping criteria, which may be a given number of pages to be collected or a maximum time allowed for the crawl, has been reached. Besides their use on web search engines, crawlers can also be used in any other computational problems that need gathering web pages, like question answering systems and for using the web as a corpus.

Depending on their purpose, surface web crawlers can be classified in two categories [Liu 2009]: *universal crawlers*, which aim to collect all kinds of pages, and *focused crawlers* which intend to gather only pages belonging to a set of topics of interest. Focused web crawlers reveal themselves very useful for acquiring corpora about a given domain. Corpora gathered by a focused crawler do not have to be limited to a single language. A *multilingual comparable corpus* can be used to train topic specific machine translation (MT) systems. In this work, we treat *comparable corpora* as sets of texts about a given subject written in more than one language. They differ from parallel corpora in that they do not need to be exact translations of one another.

In this work, we analyze the correlation between the performance of a focused crawling algorithm and the quality of the linguistic resources that can be extracted from the gathered corpora. To estimate the quality of the resources, we use a focused crawling algorithm to collect comparable corpora and use them to train a domain-specific MT system, and then evaluate the generated translations, which must be about the same domain that guided the crawl. We also propose to use the quality of the lexical resources obtained as a novel measure to evaluate focused crawling algorithms. Since we believe that the quality of the translations generated by the trained MT system reflects the quality of the lexical resources obtained, this seems to be a reliable, although extrinsic, metric to assess the performance of focused crawling algorithms.

The remainder of this paper is organized as follows: In Section 2, we introduce the background knowledge necessary to understand our methodology and experiments. Section 3 presents related work, important to the development of this one. Sections 4 and 5 bring, respectively, the details of our methodology and of our preliminary experiments. Finally, Section 6 brings a discussion of the results and our intentions for future work.

2. Background

According to [Liu 2009], the performance of focused web crawlers is measured with a set of metrics based on the similarity of the collected pages and the topic of interest. Such metrics include the *Average Precision*, which is the average of the similarity of all collected pages with the topic of interest, and the *Harvest Rate*, that can be formalized as the ratio between the number of relevant pages collected (i.e., the ones whose similarity exceeds a certain threshold) and the total number of collected pages.

The aim of focused crawling algorithms is to visit relevant pages first. To better accomplish this task, most of the state-of-the-art algorithms take into account the radius one hypothesis [Chakrabarti 2002], which states that an outlink of a page related to a subject is more likely to belong to the same subject than a random web page. We estimate the relevance of a page by calculating its cosine similarity with a query, representing them both in a vector space model.

To calculate the similarity between two documents in the vector space model , we set each weight of every vector corresponding to a document as the IDF (*inverse document frequency*) of a term multiplied by its normalized frequency. The IDF of a term expresses how rare t is in a collection C and is formalized as $IDF(t, C) = \log \frac{\|C\|}{df(t,C)}$, where $df(t, C)$ is the number of documents of C containing t.

One of the simplest algorithms for focused crawling is the Best-First Search (BFS), as described in [Liu 2009]. It uses the radius one hypothesis to guide the crawl, organizing the URL frontier prioritizing the outlinks of the most relevant pages. At each iteration, the page pointed by the URL located at the head of the queue is collected. A slight variation of this method is the *Best-N-First*, that collects the N first URLs in the queue, instead of just the first.

3. Related Work

In [Granada et al. 2012], a methodology for the acquisition of comparable corpora exploiting commercial web search engines is proposed. Their approach relies on the availability of a multilingual ontology which is used to identify the important concepts in the domain of interest. The ontology labels are combined in groups. Each of these groups is submitted to a search engine and the documents corresponding to the returned URLs are retrieved and parsed. An interesting particularity of this work is that the ontology labels can be multiword expressions, which tend to be less polysemic than single terms and, as a result, have a better expressive power. A potential limitation, however, is the dependency on a web search engine, which may lead to problems if the search engine changes the way their services are accessed, by limiting the number of allowed queries, charging fees for the services or simply changing the way requests must be done.

[Talvensaari et al. 2008] propose the use of focused web crawlers to acquire comparable corpora in order to train domain-specific MT systems. The crawling method is divided into two stages. First, queries with keywords related to the target domain are manually submitted to a search engine. Then, fifty more queries are built, using the most frequent words in the result pages. The results of these later queries are scored according to their frequency and rank inside a query and are grouped by host. The second stage is the crawling itself. The seed pages are the URLs with the highest scores in the hosts whose pages had the largest sum of scores. The score given to each page in the frontier are measured by the ratio of relevant words (words present in a query, built with the same terms used to build the previous fifty queries) in the anchor text where the link was found in the pointer page and in the set of pages belonging to the same host as the pointer page. The corpus obtained at the end of the process was used to train automatic MT systems. In their experiments, some texts related to the domain of interest were translated using the proposed approach and their results overcame the ones from the baseline translator, trained with a larger corpus but on a generic domain.

[Uszkoreit et al. 2010] present a methodology for using highly heterogeneous corpora for training statistical MT systems. The main goal is to identify document-pairs in which one is a translation of the other. For this, they are initially translated into a common language, using a baseline MT system. Then, two sets of n-grams are extracted from every document. The first is called the set of matching n-grams and is used to build a list of candidate document-pairs. Two documents are put in this list only if the number of common matching n-grams are equals to or higher than a certain threshold. The second is a set of lower order n-grams and is used to measure how similar are two documents from a candidate document-pair. The similarity between two documents is measured by using only the IDF feature of the document vectors. Finally, sentences are aligned from every document-pair and they are used to train a MT system. In the experiments, they used around one billion documents as input for their method.

The work of [Achananuparp et al. 2008] brings a comparison of many different metrics for sentence similarity. Those metrics can be divided in three main groups. The simplest group contains metrics that consider only the number of words shared by two sentences. The second one is composed by metrics based on the classic TF-IDF model. The third, and more complex one, is the group that contains language based measures, depending on resources that indicates semantic relations between terms. All those metrics are used to calculate similarity between sentences and the results indicate that although it does not always well evaluated with recall, the classic TF-IDF model usually provides a good precision.

Our work has similarities with [Talvensaari et al. 2008], since we also use corpora collected with focused crawling to train MT systems. Although we are aware that using search engines may be a good alternative to collect comparable corpora, we intend not to depend on their availability. We make use of the strategy of [Uszkoreit et al. 2010] by making an initial translation in order to have all our sentences written in the same language, for us to be able to compare them with the classic TF-IDF model. Details of our proposal are described in the Section 4.

4. Crawling for Comparable Corpora

This section details our focused crawling algorithm. Also, we describe how we preprocess and use the collected corpora to train a MT system. The experimental details are on the Section 5.

We developed an extensible focused crawler which, from a given set of seed pages and a set of relevant pages, used as reference to assess relevance, seeks to fetch other relevant pages from the domain and languages of interest. The crawler has a URL manager, which maintains a list containing all the already visited URLs and selects which of them should be visited next, according to a score assigned to each URL. Hence, it is only necessary to implement new ways of assigning scores to URLs to be able to compare the differences between two or more distinct crawling strategies.

Figure 1 summarizes the crawler structure. First, the *main* method sends a configuration file to the *Config* class to properly load all user defined options, which include definitions of parameters like n-gram size, the crawling algorithm, the set of seed pages, and locations of files to load features such as the IDF vectors. Then, it starts processing URLs received from the *URL Frontier Handler*, that sorts its URL frontier following the

algorithm specified by the user, and sends the contents of the documents and the URLs to the crawler. These will be to managed according to the used method. The *Robots Analyzer* reads the $robots.txt$ file, present in the root of the host server, to prevent any disallowed URL from being added to the URL frontier. The score assigned to every document is computed by the *URL Frontier Handler*. Normally, it just calls the *Document Similarity* method, but it can be easily overwritten, in order to change the crawler's behavior. The *Document Similarity* method uses the standard cosine similarity measure between a document and a given query to estimate the relevance of every document. It is important to note that the relevance and the score are not necessarily equal, since the relevance is always calculated by the classic TF-IDF cosine similarity between the document and a query, while the score can be anything the user finds convenient. The *HTML Parser* is the module responsible for connecting with the servers, parsing the HTML documents, storing the texts in a repository, and sending them as a set of tokens to the *main* class.

All focused crawling algorithms that we have fully implemented are based on the *Best-N-First*. While a document is being parsed, every term found in it is stemmed and added to a vector that represents it. When the parsing finishes, the cosine similarity between this vector and a query is calculated and all the outlinks are added to the URL frontier, using this similarity as the score. We also implemented another algorithm that is slightly different from the *Best-N-First*, in what it uses n-grams to compose its vectors, instead of single terms.

After the crawling stage, the gathered corpora are pre-processed as follows. First, the corpora are split in sentences, which are, then, translated into a common language by a rule-based translation system. Then, we use the TF-IDF model to find similar sentence pairs that were originally written in different languages. The original, untranslated, text of similar sentences are used as input to train a statistical MT system.

Our main goal is to compare the quality of the focused crawling algorithms, measured with usual metrics, such as the harvest rate, with the quality of the lexical resources that can be extracted from the collected corpora, to establish a correlation between both. Besides, we propose to use the quality of these resources, measured by the accuracy of the generated translations, as an extrinsic measurement to evaluate a focused crawler.

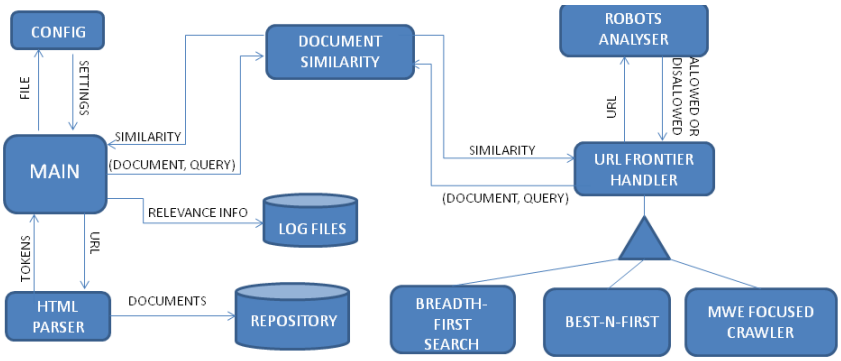


Figura 1. Basic crawling procedure

5. Preliminary Experiments

The goal of the experiment was to use the collected corpora to translate texts between Portuguese, English, and French on the *genetics* domain. For the crawling stage, the linguistic

resources used were generic corpora to compute IDFs for every term n-gram present in them. Newspaper collections from the years 1994 and 1995 for each of the languages were used. For English, the Los Angeles Times and Glasgow Herald were used; for French, we used texts from Le Monde and from the Swiss Document Agency. All these collections are available in the CLEF Test Suite, which can be purchased from ELRA[1]. For Portuguese, we used texts from Folha de São Paulo[2]. Due to memory limitations, we pruned all terms and n-grams that appeared in fewer than five documents.

We compared the *Best-50-First* (*Best-N-First* with $N = 50$) using single terms and using bigrams and trigrams. For every crawling algorithm, our stopping criteria was to reach 10000 successfully collected pages. The next stage was to split the output corpora into sentences. To perform this task, we used the Python *Natural Language Toolkit* [Loper and Bird 2002] sentence splitter for all three languages. Sentences containing less than 3 or more than 22 words were pruned. We did this to ignore stand alone terms or expressions, such as titles or menus, and cases where the sentence splitter grouped sentences and expressions together. We chose 22 to be the maximum number of words allowed in a sentence because of the analysis done in [Granada et al. 2012], who computed the average number of words in a comparable corpus. In Portuguese, they found 20.07 words per sentence, while for French and English, the numbers were 15.91 and 14.15, respectively.

To perform the sentence alignment, we first use Moses, which is an open-source statistical MT toolkit, that offers several tools for preprocessing, training a translator with parallel, data and translating sentences [Koehn et al. 2007]. We used a Moses translator trained with parallel sentences from the Europarl [Koehn 2005], to translate all sentences written in Portuguese and French to English. In order to avoid comparing all possible sentence-pairs, we index our sentences with Zettair[3], a search engine that allows the user to build indexes and make customized queries. To find what sentences are similar to a given sentence, we send her as a query to Zettair, that searches for similar sentences in an index that was built with all sentences that were originally written in one of the other two languages. We also set the search engine's ranking function to be the cosine similarity metric. Finally, we use the original forms of all pairs of similar sentences to train Moses.

To reach our main goal, which is to establish a correlation between the focused crawling performance and the quality of the lexical resources extracted from the collected corpora, we must know results of both measurements. We estimate the focused crawling performance using the standard metrics harvest rate and average precision. To evaluate the quality of the lexical resources, we generate translations about the genetics domain with the MT system trained with our comparable corpora. The translations are judged by humans that speak both source and target languages and the quality of the resources is measured from these judgments.

We have fully ran our experiments only until the crawling stage. The results indicate that the *Best-50-First* with unigrams outperforms the one with trigrams, which showed better results than bigrams. Although the sentence alignment and translation stages are still not fully implemented and evaluated, we expect the translator trained with corpora collected with the first algorithm to outperform the other ones.

[1]http://www.elra.info/
[2]Available from http://www.linguateca.pt
[3]http://www.seg.rmit.edu.au/zettair/

6. Conclusion

The aim of this work is to evaluate the quality of focused crawling algorithms and the quality of the lexical resources extracted from the respective collected corpora, in order to find a correlation between both and propose to use the quality of the resources as an extrinsic metric to evaluate focused crawlers. We have already implemented and experimented *Best-50-First* algorithm using single terms, bigrams and trigrams. Besides, we are still implementing a focused crawler that uses Hidden Markov Models to learn user browsing patterns [Liu et al. 2006] and developing another one that takes advantage of the expressive power of multiword expressions. Finally, to be able to establish a correlation between the measurements, we still need to manually judge a considerable quantity of generated translations between all language pairs.

Acknowledgements: The first author receives a scholarship from CNPq. This work was partially funded by CNPq (projects 478979/2012-6 and 480283/2010-9) and CAPES-COFECUB (project 707/11).

Referências

Achananuparp, P., Hu, X., and Shen, X. (2008). The evaluation of sentence similarity measures. In *Data Warehousing and Knowledge Discovery*, pages 305–316. Springer.

Chakrabarti, S. (2002). *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann.

Granada, R., Lopes, L., Ramisch, C., Trojahn, C., Vieira, R., and Villavicencio, A. (2012). A comparable corpus based on aligned multilingual ontologies. In *Proceedings of the First Workshop on Multilingual Modeling*, pages 25–31. ACL.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand. AAMT.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180. ACL.

Liu, B. (2009). *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications)*. Springer.

Liu, H., Janssen, J., and Milios, E. (2006). Using HMM to learn user browsing patterns for focused web crawling. *Data & Knowledge Engineering*, 59(2):270–291.

Loper, E. and Bird, S. (2002). Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1*, pages 63–70. ACL.

Talvensaari, T., Pirkola, A., Järvelin, K., Juhola, M., and Laurikkala, J. (2008). Focused web crawling in the acquisition of comparable corpora. *Information Retrieval*, 11(5):427–445.

Uszkoreit, J., Ponte, J. M., Popat, A. C., and Dubiner, M. (2010). Large scale parallel document mining for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1101–1109. ACL.

Análise da Evolução Temporal de Dados Complexos

Isis Caroline O. V. de Sousa, Renato Bueno

Programa de Pós Graduação em Ciência da Computação

Departamento de Computação – Universidade Federal de São Carlos (UFSCar)

São Carlos – SP – Brasil

{isis.sousa, renato} @dc.ufscar.br

Nível: Mestrado

Ingresso no programa: Primeiro semestre de 2012

Exame de qualificação: junho de 2013

Época esperada de conclusão: março de 2014

Abstract. *In complex data, it is common to make similarity queries, using the features extracted of these data. These data are in general represented in metric spaces, where only the elements and their features are known. Considering the necessity of associate time to metric data, the objective is analyze the temporal evolution of the metric-temporal data, proposed on a previous work, through the embedding of these data to multidimensional spaces. In the multidimensional space, we can analyze the trajectories, estimate the data's status in different moments of time and perform similarity search to this estimate in the multidimensional space. Initially, we intend to study algorithms of embedding that can preserve the distances between the elements as in the original space. Then, we will propose and evaluate different kinds of similarity search to perform estimates in the embedding space evaluating the outcomes in the moment of the search, starting with Range Query and Reverse k -NN. To finish, we intend to evaluate the use of multiple reference elements, when they are available, to calculate the estimates. With the results of the embedding and the queries on the estimates, we intend to validate the method proposed providing a support to real applications.*

Keywords: Complex Data. Metric Space. Temporal Evolution. Embedding. Similarity Search.

1. Problema de pesquisa e caracterização da contribuição

Além dos tipos convencionais de dados (números, datas e pequenas cadeias de caracteres), é cada vez mais comum a necessidade de suportar dados complexos, como imagens, áudio, vídeos, etc. Para realizar uma consulta aos dados complexos, um conjunto de características pode ser extraído dos mesmos e as comparações entre eles são baseados na relação de (dis)similaridade entre seus vetores de características [9]. Dessa forma, o problema essencial é encontrar, no conjunto de elementos, aqueles que são mais similares ao elemento de consulta, utilizando uma função de distância [1]. Realiza-se então consultas por similaridade, utilizando os sistemas baseados em Recuperação por Conteúdo (*Contend-based Retrieval* – CBR). Em se tratando de imagens (tipo de dado complexo a ser utilizado neste trabalho), utiliza-se a Recuperação de Imagem por Conteúdo (*Content-based Image Retrieval* – CBIR). Dados complexos são em geral representados em espaços métricos, domínio de dados onde as únicas informações disponíveis são os elementos e as distâncias entre eles.

A principal motivação para o estudo proposto é a necessidade de acrescentar a informação temporal aos dados métricos, pois em muitas aplicações, como acompanhamento de pacientes através de imagens de exames médicos, a associação do tempo é fundamental. Em [3] foi proposto o espaço métrico-temporal, que consiste no acréscimo da informação do tempo no espaço métrico, através de uma componente temporal. Outra proposta preliminar, idealizada por [2], refere-se ao mapeamento do espaço métrico-temporal para um espaço multidimensional, possibilitando, com os dados representados nesse espaço, analisar seu comportamento evolutivo no decorrer do tempo.

O objetivo deste trabalho é aperfeiçoar e estender a proposta preliminar de [2] para análise da evolução temporal de dados complexos. Pretende-se inicialmente escolher algoritmos de mapeamento dos dados para o espaço multidimensional que mantenham a distribuição dos dados no espaço original, possibilitando uma melhor avaliação da qualidade das estimativas. Serão propostas novas maneiras de se estimar o estado de um elemento em momentos de tempo diferentes daqueles em que estão disponíveis na base de dados. Por fim, será avaliada a utilização de múltiplos elementos de referência, em diferentes momentos, estendendo assim a proposta original.

O restante do artigo está estruturado da seguinte maneira: Na seção 2 é apresentada a fundamentação teórica do problema de pesquisa. Na seção 3 é apresentada a proposta preliminar de [2]. Na seção 4 é apresentado o desenvolvimento necessário para conclusão e na seção 5 as considerações finais e os resultados esperados.

2. Fundamentação Teórica

Dados complexos, diferentemente dos convencionais, não possuem uma Relação de Ordem Total, impossibilitando a utilização dos operadores relacionais ($<$, $\leq$, $>$, $\geq$). O procedimento comumente adotado é representar esses dados em espaços métricos e recuperá-los por similaridade.

Nas consultas por similaridade, o conteúdo do objeto complexo é comumente representado através do seu vetor de características. Os vetores de características são comparados para que se possa obter o grau de similaridade entre os objetos, onde quanto menor for o valor da distância resultante, maior é a similaridade entre os objetos.

Os tipos de consulta por similaridade mais comum são: (a) *Range Query*: a partir de um objeto de consulta e um raio de abrangência, são retornados os objetos que estão dentro desse raio; e (b) *k-Nearest Neighbor (k*-NN*)*: a partir de um objeto de consulta e um número k de elementos que devem ser recuperados, são retornados os k mais próximos. Dentre vários outros métodos e variações, neste trabalho pretende-se utilizar a *Reverse k*-NN [10] para avaliar a qualidade das estimativas retornadas por uma consulta. Em uma consulta *Reverse k*-NN são retornados os elementos que tem o objeto de consulta como um dos *k* vizinhos mais próximos.

2.1 Espaços Métricos

Em um espaço métrico não são consideradas informações geométricas ou dimensionais dos dados, isto é, só estão disponíveis os elementos de dados e as distâncias (dissimilaridades) entre eles. Para calcular as distâncias entre os elementos do espaço, a função de distância (métrica) deve satisfazer três propriedades [4]: Simetria ($d(x,y) = d(y,x)$), Não-negatividade ($0 < d(x,y) < \infty,\ x \neq y, d(x,x) = 0$), e Desigualdade Triangular ($d(x,y) \leq d(x,z) + d(z,y)$).

Para incorporar a informação temporal nas consultas por similaridade, foi desenvolvido o espaço métrico-temporal [3], composto por dois espaços métricos (uma componente métrica e uma temporal), sendo: $\langle S, d_s \rangle$ o espaço métrico para o cálculo da similaridade, sendo que S representa o conjunto de dados e d_s uma métrica para calcular a similaridade entre os elementos; e $\langle T, d_t \rangle$ o espaço métrico para as medidas de tempo, em que T representa as medidas de tempo e d_t a métrica para o cálculo da similaridade entre os valores de tempo, que podem ser instantes ou períodos temporais.

3. Proposta Preliminar

As consultas realizadas em espaços métricos ou mesmo métrico-temporais não permitem analisar a evolução temporal dos dados, pois mesmo com as informações temporais presentes, são disponíveis somente os elementos e as distâncias entre eles. Essa análise é necessária em muitos domínios de aplicação, para analisar o comportamento evolutivo dos dados complexos no decorrer do tempo, como na medicina, meteorologia, agricultura, entre outras. Um exemplo aplicado à medicina é o acompanhamento do diagnóstico de um paciente através de imagens de exames médicos.

Para estudar essa necessidade, pretende-se estender, nesse trabalho, a proposta apresentada por [2], onde foi proposto o mapeamento do espaço métrico-temporal para um espaço multidimensional, para possibilitar a análise da evolução temporal dos dados complexos. Baseando-se nas informações existentes de um determinado objeto no banco de dados (instâncias desse objeto em tempos diferentes), pode-se estimar e analisar o estado desse mesmo objeto em um outro instante no tempo.

Considere como exemplo o acompanhamento do diagnóstico de um paciente através de imagens de exames médicos, como ilustrado na Figura 1. No espaço mapeado, os pontos representam as imagens mapeadas de exames de pacientes. Existem duas imagens indexadas referentes ao paciente P_A, uma antes de iniciar o tratamento $(t = 0)$ e outra com 12 meses de tratamento $(t = 12)$. Deseja-se estimar o estado desse paciente quando ele estiver com 15 meses de tratamento. Por meio das posições das duas imagens de P_A existentes (em $t = 0$ e $t = 12$), estima-se qual será sua posição em

$t = 15$, utilizando interpolação/extrapolação. Porém, não é possível a construção/reconstrução de uma imagem (no espaço original) a partir dessa estimativa no espaço multidimensional. Realiza-se então uma consulta por similaridade (k-NN) no espaço multidimensional utilizando essa posição estimada como centro de consulta. Os objetos retornados são aqueles presentes na base que são os mais próximos da estimativa de P_A em $t = 15$.

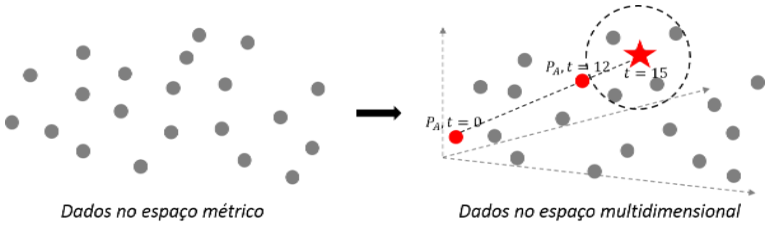


Figura 1: Exemplo de estimativa e consulta no espaço mapeado.

3.1 Experimentos

Em [2] foram realizados experimentos utilizando o algoritmo *FastMap* [6] para o mapeamento dos dados métricos. Foi utilizado o conjunto de imagens ALOI (*Amsterdam Library of Object Images*) [7], onde cada objeto (de um conjunto de 1000 objetos) foi fotografada em 72 ângulos de visão (com rotação de 5 graus entre uma e outra). Admite-se que diferentes posições dos objetos referem-se a diferentes tempos, como mostrado na Figura 2.

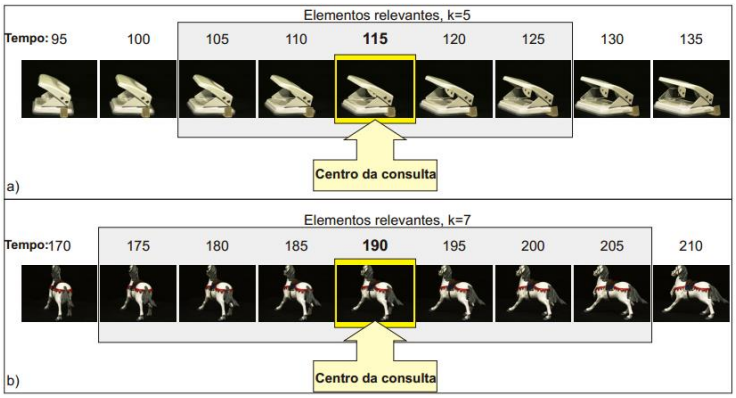


Figura 2: Exemplo do conjunto de imagens utilizado. [2].

Para cada objeto foram utilizadas duas imagens em tempos diferentes, a partir das quais foi estimado o estado do objeto de consulta em outra posição temporal. Foram extraídas características de cor (Histogramas) e formas (Zernike). Por exemplo, a partir de duas instâncias de um objeto x nos tempos 5 e 15, realizou-se uma estimativa da posição do objeto no tempo 20 e realizou-se uma consulta k-NN nessa posição. Para verificar a precisão da consulta, foi realizada, no espaço métrico-temporal original, a mesma consulta k-NN utilizando o objeto x no tempo 20 e os resultados das consultas foram comparados. Essa comparação foi utilizada para avaliar a qualidade das estimativas, realizadas em tempo passado, intermediário e futuro.

Para avaliar a qualidade do mapeamento, como mostrado no gráfico apresentado na Figura 3, a curva denominada "exato" compara os resultados de consultas k-NN sobre objetos no espaço métrico original com os resultados das mesmas consultas realizadas diretamente no espaço mapeado. As demais curvas indicam os resultados

referentes à avaliação das estimativas. Como pode ser visto, a qualidade dos resultados na análise das estimativas teve comportamento muito parecido com a qualidade do mapeamento, sendo um forte indicativo de que, por terem os mesmo níveis de precisão, a qualidade do mapeamento influenciou na qualidade das consultas às estimativas.

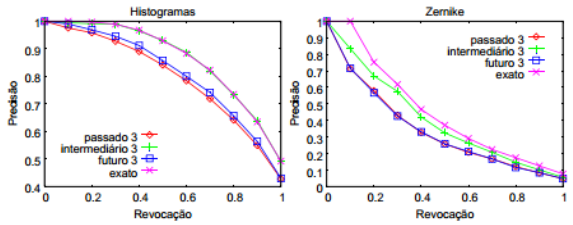


Figura 3: Avaliação da qualidade do mapeamento [2].

4. Desenvolvimento necessário para conclusão

Para dar continuidade ao estudo apresentado em [2], esse trabalho tem como parte da proposta a utilização de outros algoritmos para fazer o mapeamento dos dados do espaço métrico-temporal para o espaço multidimensional, a fim de manter o mais fielmente possível a distribuição dos dados. Considerando a possibilidade de maior precisão, dois algoritmos foram pré-selecionados a partir da literatura para mapear os dados e comparar os resultados, inclusive com o algoritmo inicialmente utilizado (*FastMap*). O algoritmo MDS clássico [11], técnica referente ao *Multidimensional Scaling* (MDS) faz o mapeamento a partir de uma matriz de distâncias entre os dados e a decomposição espectral dessa matriz. O *SparseMap* [8] faz o mapeamento dos dados a partir dos cálculos de distância entre os objetos do conjunto inicial e seus subconjuntos. Para testar os algoritmos, serão utilizadas bases de dados controladas considerando tamanho da base, custo do mapeamento, precisão e distribuição dos dados.

Numa segunda etapa da proposta, serão estudados outros tipos de consulta que possam proporcionar melhorias nos resultados e nas avaliações dos mesmos. As consultas propostas em [2], realizadas apenas com *k*-NN, impossibilitam a avaliação da qualidade dos resultados no momento da consulta. Além disso, os vizinhos mais próximos retornados podem não ser próximos o suficiente da estimativa para representar um bom resultado. Por exemplo, deseja-se estimar o estado de dois pacientes P_A e P_B nos tempos 15 e 8, respectivamente, como pode ser visto na Figura 4. Os objetos retornados para P_A em $t = 15$ estão muito mais distantes da estimativa de P_A em $t = 15$ do que aqueles retornados para P_B em $t = 8$ estão da estimativa de P_B em $t = 8$. Logo, a qualidade dos elementos retornados para P_B tende a ser superior à qualidade dos que foram retornados para P_A, pois provavelmente serão imagens mais próximas do que se estima ser o estado do paciente P_B com 8 meses de tratamento.

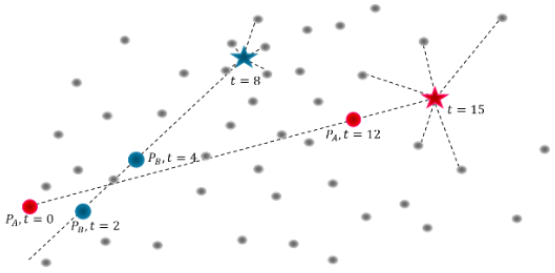


Figura 4: Exemplo de estimativa utilizando *k*-NN.

Serão estudadas e propostas duas possíveis maneiras de avaliar a qualidade das estimativas no momento da consulta, utilizando *Range Query* e *Reverse k*-NN.

Utilizando a *Range Query*, é possível delimitar a distância máxima desejada para os objetos retornados, sendo o raio dessa consulta um indicativo da qualidade dos resultados da estimativa. Isto é, dado um determinado raio de distância, se não houver elementos nesse raio, elementos mais distantes não serão retornados, diminuindo de certa forma a incidência de falsos positivos. Pretende-se estudar maneiras de utilizar a dimensão intrínseca do conjunto para a definição desse raio de abrangência, sendo que a dimensão intrínseca de um conjunto de dados é a dimensão do objeto no espaço representado pelo conjunto, independentemente do espaço onde eles estão imersos [5].

Utilizando a consulta *Reverse k*-NN após a *k*-NN, pode-se conferir se os objetos retornados possuem o objeto de consulta como um dos vizinhos mais próximos. Se os objetos retornados não possuírem o objeto de consulta como um de seus vizinhos mais próximos, pode ser um indicativo de que esse objeto retornado não representa um bom resultado.

Em uma terceira etapa do trabalho, pretende-se estudar maneiras de realizar as estimativas utilizando um número maior de elementos de referência, sendo que na proposta de [2] foram utilizados apenas dois elementos, com estimativas utilizando apenas interpolação/extrapolação. Quando presentes na base de dados, mais objetos podem ser utilizados com parâmetro para fazer a estimativa no tempo desejado. Além disso, pretende-se comparar os resultados das consultas realizadas a partir de dois objetos e a partir de um número maior deles, a fim de analisar se os resultados apresentam melhorias em precisão, considerando também o custo computacional adicional. No exemplo ilustrado na Figura 5, onde deseja-se também estimar o estado de P_A no tempo 15 e de P_B no tempo 8, se houver mais de duas imagens referentes a cada objeto, elas podem também ser utilizadas para fazer as estimativas.

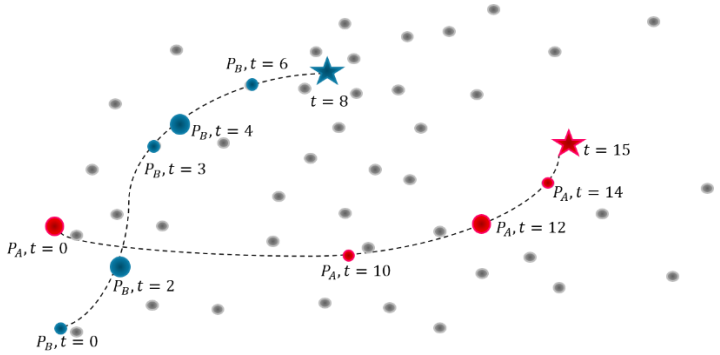


Figura 5: Exemplo de estimativa utilizando mais de dois objetos de referência.

5. Considerações Finais e Resultados Esperados

Considerando que não é possível fazer uma análise da evolução temporal em dados puramente métricos, o mapeamento desses dados para um espaço multidimensional torna-se necessário. Mesmo no espaço métrico-temporal, embora exista a informação do tempo, essa informação não representa necessariamente uma ordem cronológica. Verificada a efetividade da análise das trajetórias dos dados métrico-temporais, pode-se então proporcionar um modelo de aplicação que auxilie muitos domínios de aplicação que utilizam dados complexos.

Logo, espera-se, como resultados desse trabalho:

- Escolher novos métodos de mapeamento dos espaços métricos para espaços multidimensionais, visando manter a distribuição original dos dados, para que seja possível melhor avaliar a qualidade das estimativas;
- Estudar maneiras de avaliar a qualidade das respostas na consulta às estimativas no momento da consulta; e
- Explorar a utilização de mais elementos de referência para gerar as estimativas.

Com os resultados, espera-se possibilitar o uso desse modelo de análise dos dados complexos a aplicações reais, auxiliando nas aplicações necessárias.

6. Referências

[1] ALMEIDA, J. et al. **DAHC-tree: An Effective Index for Approximate Search in High-Dimensional Metric Spaces.** JIDM, v. 1(3), p. 375-390, 2010.

[2] BUENO, R. **Tratamento do tempo e dinamicidade em dados representados em espaços métricos.** Tese (Doutorado em Ciência da Computação). Instituto de Ciências Matemáticas e de Computação, USP. São Carlos, 2009.

[3] BUENO, R. et al. **Time-Aware Similarity Search: A Metric-Temporal Representation for Complex Data.** Proceedings of the 11th International Symposium on Advances in Spatial and Temporal Databases. Aalborg, Denmark: Springer-Verlag: 302-319 p. 2009.

[4] C. TRAINA, J. et al. **Fast Indexing and Visualization of Metric Data Sets using Slim-Trees.** IEEE Trans. on Knowl. and Data Eng., v. 14, n. 2, p. 244-260, 2002. ISSN 1041-4347.

[5] C. TRAINA, J. et al. **Fast feature selection using fractal dimension.** Brazilian Symposium on Databases (SBBD). MEDEIROS , C. M. B. E. B., K. EDITORS. João Pessoa, PB: p. 158-171, 2000.

[6] FALOUTSOS, C.; LIN, K.-I. **FastMap: a fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets.** SIGMOD Rec., v. 24, n. 2, p. 163-174, 1995. ISSN 0163-5808.

[7] GEUSEBROEK, J.-M.; BURGHOUTS, G. J.; SMEULDERS, A. W. M. **The Amsterdam Library of Object Images.** Int. J. Comput. Vision, v. 61, n. 1, p. 103-112, 2005. ISSN 0920-5691.

[8] HRISTESCU, G.; FARACH-COLTON, M. **Cluster-preserving Embedding of Proteins.** Center for Discrete Mathematics; Theoretical Computer Science. 1999

[9] KASTER, D. S. et al. **Nearest Neighbor Queries with Couting Aggregate-based Conditions.** JIDM, v. 2(3), p. 401-416, 2011.

[10] TAO, Y. et al. **Multidimensional reverse kNN search.** The VLDB Journal, v. 16, n. 3, p. 293-316, 2007. ISSN 1066-8888.

[11] YOUNG, G.; HOUSEHOLDER, A. S. **Discussion of a set of points in terms of their mutual distances**. Psychometrika**,** v. 3, n. 1, p. 19-22, 1938.

Efficient integrity checking for untrusted database systems

Anderson Luiz Silvério[1],
Supervised by Ronaldo dos Santos Mello[1] **and Ricardo Felipe Custódio**[1]

[1] Programa de Pós-Graduação em Ciência da Computação
Universidade Federal de Santa Catarina
Florianópolis – SC – Brazil

{anderson.luiz,ronaldo,custodio}@inf.ufsc.br

Level: MSc
Admission: March 2012
Qualifying exam: June 2013
Conclusion: March 2014 (expected)
Steps completed: literature review, preliminary solution and evaluation
Future steps: complete solution and evaluation

Abstract. *Unauthorized changes to database content can result in significant losses for organizations and individuals. As a result, there is a the need for mechanisms capable of assuring the integrity of stored data. Meanwhile, existing solutions have requirements that are difficult to meet in real world environments. These requirements can include modifications to the database engine or the usage of costly cryptographic functions. In this paper, we propose a technique that uses low cost cryptographic functions and is independent of the database engine. Our approach allows for the detection of malicious data update operations, as well as insertion and deletion operations. This is achieved by the insertion of a small amount of protection data into the database. The protection data is calculated by the data owner using Message Authentication Codes. In addition, our experiments have shown that the overhead of calculating and storing the protection data is minimal.*

Keywords: Data Integrity, Outsourced Data, Untrusted Database

1. Introduction

Database security has been studied extensively by both the database and cryptographic communities. In recent years, some schemes have been proposed to check the integrity of the data, that is, to check if the data has not been modified, inserted or deleted by an unauthorised user or process. These schemas often try to resolve one of the following aspects of the data:

- *Correctness*: From the viewpoint of data integrity, correctness means that the data has not been tampered with.
- *Completeness*: When a client poses a query to the database server it returns a set of tuples that satisfies the query. The completeness aspect of the integrity means that all tuples that satisfy the posed query are returned by the server.

In trying to assure database integrity, many techniques have been proposed, such as in [Kamel, 2009; Li et al., 2006]. However, most of them rely on techniques that require modification of the database kernel or the development of new database management systems. Such requirements make the application of the integrity assurance mechanisms in real-world scenarios difficult. This effort becomes more evident when we consider adding integrity protection to already deployed systems.

Most of the remaining work is based on authenticated structures [Di Battista and Palazzi, 2007; Heitzmann et al., 2008; Miklau and Suciu, 2005], such as Merkle Hash Trees [Merkle, 1989] and Skip-Lists [Pugh, 1990]. These works are simpler to put into practice, since they don't require modifications to the kernel of the DBMS. However, the use of authenticated structures limits their use to static databases. Authenticated structures are not efficient in dynamic databases because the structure must be recalculated for each update.

We assume the data owner outsources its data to an untrusted database server. This assumption is particularly interesting nowadays with cloud storages and a global market. Increasingly, businesses are outsourcing their data in order to reduce the maintenance costs of its infrastructure, to increase scalability and to simplify the growth of the infrastructure [Samarati and Capitani, 2010].

This scenario is preferable for the data owner in terms of operational costs, but by storing data in outsourced database systems, there are no mechanisms to guarantee the security of the outsourced data. That means the remote server can read, modify, insert and remove information. We consider the server to be vulnerable to external attacks, such as the case of an attacker gaining access to the server and performing malicious modifications to the stored data, or internal attacks, where someone from the personnel can be coerced to perform such modifications. Although we cannot prevent such attacks from happening in reality, our goal is to create the means to detect such attacks.

The primary contribution of our work is the design, implementation and performance evaluation of Message Authentication Codes (MACs) to provide the *correctness* aspect of data integrity for a client that stores his/her data in an untrusted relational database server. We first present a simple technique to allow the client to detect updates to the stored data and insertion of data by an attacker. Additionally, we introduce a new algorithm, called Chained-MAC (CMAC), to allow the client to detect the deletion of data.

The remainder of this paper is divided into five sections. In Section 2 we discuss related work. In Section 3 we describe in details our techniques for providing integrity assurance. In Section 4 we analyse the performance impact of our proposed method and section 5 presents our final considerations and future works.

2. Related work

The major part of integrity verification found in literature is based on authenticated structures. Namely, Merkle Hash Trees [Merkle, 1989] and Skip-Lists [Pugh, 1990].

Li et al. [Li et al., 2006] present the Merkle B-Tree (MB-Tree), where the B^+-tree of a relational table is extended with digest information as in an Merkle Hash Tree (MHT). The MB-Tree is then used to provide proofs of correctness and completeness for posed queries to the server. Despite presenting an interesting idea and showing good results in their experiments, their approach suffers from a major drawback. To deploy this approach, the database server needs to be adapted as the B^+-tree needs to be extended to support an MHT. Such modifications may not be feasible in real world environments, especially those that are already in use.

Di Battista and Palazzi [Di Battista and Palazzi, 2007] propose to implement an authenticated skip list into a relational table. They create a new table, called *security table*, which stores an authenticated skip list. The new table is then used to provide assurance of the authenticity and completeness of posed queries. This approach overcomes the requirement of a new DBMS, present in the previous approach. While only a new table is necessary within this approach, its implementation can be done as a plug-in to the DBMS. However, the experimental results are superficial. It is not clear what is the actual overhead in terms of each SQL operation. Moreover, their experiments show that the overhead increase as the database increases, while in our approach the overhead is constant in terms of the database size.

Miklau and Suciu [Miklau and Suciu, 2005] implement a hash tree into a relational table, providing integrity checks for the data owner. The data owner needs to securely store the root node of the tree. To verify the integrity, the clients need to rebuild the tree and compare the root node calculated and stored. If they match, the data was not tampered with. Despite using simple cryptographic functions, such as hash, the use of trees compromises the efficiency of their method. A tuple insert using their method is 10 times slower than a normal insert, while a query is executed 6 times slower. In our experiments, presented in section 4, we show that the naive implementation of our method is as good as their method.

E. Mykletun et al. [Mykletun et al., 2006] study the problem of providing *correctness* assurance of the data. Their work is most closely related to what we present in this paper. They present an approach for verifying data integrity, based on digital signatures. The client has a key pair and uses its private key to sign each tuple he/she sends to the server. When retrieving a tuple, the client uses the correspondent public key to verify the integrity of the retrieved tuple. This work was extended by Narasimha and Tsudik [Narasimha and Tsudik, 2005] to also provide proof of *completeness*.

The motivation of the authors to use digital signatures is to allow integrity checking in multi-querier and multi-owner models. Therefore, for multi-querier and multi-

owner models, their work is preferable. On the other hand, if the querier and the data owner are the same, our work can provide integrity assurance more efficiently.

3. Contributions

To achieve a low cost method to provide integrity and authenticity, we propose to perform the cryptographic operations on the client side (application), using Message Authentication Codes (MAC) [Bellare et al., 1996; Krawczyk et al., 1997]. The implementation consists of adding a new column to each table. This new column stores the output of the MAC function applied to the concatenation of the attributes (all columns, or a subset of them) of the table. The function also utilises a key, which is only known by the application. The value of the MAC column is later used to verify integrity.

The use of a MAC function ensures the integrity of the INSERT and UPDATE operations. However, the table is still vulnerable to the unauthorized deletion of rows. To overcome this issue, we propose a new algorithm for linking sequential rows, called "Chained-MAC (CMAC)". The result of the CMAC is then stored into a new column. The value of this new column, given a row n, a key k, and MAC_n as the MAC value of the row n, is calculated as follows:

$$CMAC_n = MAC(k, (MAC_{n-1} \oplus MAC_n)) \quad (1)$$

The use of CMAC provides an interesting property to the data stored in the table where it is used. When used, the CMAC links the rows in a way that an attacker cannot delete a row without being detected, since he does not have access to the secret key to produce a valid value to update the CMAC column of adjacent rows.

Despite linking adjacent rows, any subset of the first and last rows can be deleted without being detected. This is possible because the first row has no previous row and the last row does not have a subsequent row to be linked with. To overcome this issue, we propose changing the CMAC to a circular method. That is, for the first row, the $n-1$-th row to be considered will be the n-th row (i.e. the last row). With this change, if the last row is deleted, the integrity check will fail for the first row. Similarly, since the first row now has a predecessor, integrity checks can start at the first row (in the regular mode it would always start in the second row).

It is important to notice that the introduction of the CMAC brings a new requirement: the table must be ordered by some attribute. However, in real world scenarios, all tables have a primary key, and all the main DBMS orders the tables in terms of the primary key. Therefore, the requirement for a ordered table of the CMAC does not have a big impact to the deployment of our technique in real world scenarios.

3.1. Verifying the integrity of a table

To verify the integrity of a row with the MAC column, the application must calculate the MAC of that row and compare it with the value of the MAC column. The row can be considered as not modified if the calculated MAC is equal to the stored MAC. Applying this comparison to each row of a table will ensure the integrity of this table against insertion and modification attacks. As stated earlier, the use of the MAC does not provide a means to verify the integrity of a table against unauthorized deletions. In this case, the CMAC

column should be used. To verify the integrity of a table with the CMAC column, the application must check the integrity of each pair of sequential registries of the table. That is, a Table T has not been modified (unauthorized) if:

$$\forall t_{n-1}, t_n \in T : t_n.CMAC = CMAC(k, t_{n-1}, t_n) \quad (2)$$

4. Preliminary Experimental Evaluation

To assess the efficiency of our techniques we implemented a tool to evaluate the performance of using HMAC, as the MAC function, and CMAC. The prototype was implemented using the C programming language and the OpenSSL library. The DBMS used was MySQL database and the experiments were performed on a machine running both MySQL server and client application. The machine had Intel Core 2 Quad CPU Q8400 with 4Mb cache, at 2.66GHz, 4GB RAM 800Mz, and 320Gb disk, SATAII, 16Mb cache, 7200RPM, running an Ubuntu 11.04 32-bit operating system with OpenSSL 0.9.8d and MySQL 5.1. Additionally, we used the SHA-1 hash function to calculate the HMAC with a 256-bit long key.

We considered different scenarios to evaluate the performance of the proposed techniques. For each scenario, we executed the workload a thousand times over a table with 10 thousand tuples of random values. All the results shown below are the average of these executions. In all scenarios we focus on evaluating the amount of time spent on the operations of INSERT, UPDATE, DELETE and SELECT, performed under four distinct conditions:

1. Without security mechanisms;
2. Using HMAC only;
3. Using both HMAC and CMAC;
4. Using both HMAC and CMAC in the circular mode.

In the first scenario, we focused on measuring and comparing the execution times for the INSERT operation under each specified condition. The results show that the baseline took 42,3 μs, while the HMAC took 47 μs, 90% of which is spent on the server side and 10% on the client side. The scenario with the use of CMAC executed in 118,3 μs, with 91% of the time spent on the server and 9% on the client. The CMAC in the circular mode executed in 331,7 μs, where 72% is executed by the server and 28% by the client.

The CMAC in the circular mode can be optimized if the client stores a small amount of data. The major reason for the difference between the regular mode and the circular mode of the CMAC is that in the circular mode we need to retrieve and update additional rows. If the client stores the first row locally, we eliminate one query, reducing the execution time from 331,7 μs to 236,5 μs.

In the second scenario, we focused on measuring and comparing the execution times for the UPDATE operation under each specified condition. The results show that the baseline took 127,6 μs, while the HMAC took 134 μs, 95% of which is spent on the server side and 5% on the client side. The CMAC (both in regular and circular mode) executed in 381,9 μs, with 80% of the time spent on the server and 20% on the client. The reason that the execution time for the CMAC in the regular and circular mode are the same is because they execute the exact same operations.

We can also optimize the CMAC for the UPDATE operation if we consider that some values are available on the client side at the moment of the operation. In this case, when updating a row n, we need the MAC and CMAC of the $n + 1$-th and the $n - 1$-th rows. If these rows are available on the client side at the moment of the update, the execution time is 204,5 μs.

In the third scenario, we focused on measuring and comparing the execution times for the DELETE operation under each specified condition. The baseline executed in 51 μs and when using the HMAC to delete a row, there is no additional cost since there is no extra operations to be performed. On the other hand, the CMAC (both in regular and circular mode) executed in 186,5 μs, with 96% of the time spent on the server and 4% on the client. As we have shown for the UPDATE operation, the CMAC in the regular and circular mode have the exact same operations and therefore the overhead is the same.

We can use the same idea presented for the UPDATE operation to improve the efficiency of the CMAC. In the naive implementation, before deleting a row n, we execute a select query to retrieve the $n + 1$-th and the $n - 1$-th rows. Considering that these rows are available on the client side at the moment of the delete, the execution time is reduced from 186,5 μs to 105,2 μs.

Finally, in the last scenario, we focused on measuring and comparing the execution times to check the integrity during the SELECT operation under each specified condition. A SELECT query, without verifying the integrity (the baseline), took 18,4 μs. To verify the integrity of the HMAC the client needs to recalculate the HMAC and compare it to the one retrieved from the server. This operation executed in 22,5 μs, due to the calculation of the HMAC. When using the CMAC, the client needs to retrieve the HMAC of the previous row and recalculate both the HMAC and CMAC. These extra operations increase the execution time to 54 μs. However, if we consider that the previous row is available on the client side, the execution time is reduced to 27,6 μs.

5. Final remarks

This paper proposes secure and efficient methods for providing integrity for relational database systems. Our methods focus on strategies for detecting unauthorised actions (insertions, deletions and updates) from a vulnerable database server.

Prior work either requires modifications in the database implementation or uses inefficient cryptographic techniques (for example, public key cryptography). The requirement of modifying the core of a database system makes the deployment of these methods difficult in real world scenarios. Thus, one significant advantage of our method is that it is DBMS-independent and can be easily deployed in existing environments. Another advantage of our method is that we focused on using more simple and efficient cryptographic algorithms to provide integrity checks.

To complete this work we'll first improve the experimental evaluation, comparing our approach with prior work. Additionally, we'll investigate and propose methods regarding the management of the secret key, used to generate the MAC and CMAC values. That is, we need methods to allow the change of the secret key, in case of a simple key update and/or a key compromise.

References

Bellare, M., Canetti, R., and Krawczyk, H. (1996). Keying hash functions for message authentication. In *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '96, pages 1–15, London, UK, UK. Springer-Verlag.

Di Battista, G. and Palazzi, B. (2007). Authenticated relational tables and authenticated skip lists. In *Proceedings of the 21st annual IFIP WG 11.3 working conference on Data and applications security*, pages 31–46, Berlin, Heidelberg. Springer-Verlag.

Heitzmann, A., Palazzi, B., Papamanthou, C., and Tamassia, R. (2008). Efficient integrity checking of untrusted network storage. In *Proceedings of the 4th ACM international workshop on Storage security and survivability*, StorageSS '08, pages 43–54, New York, NY, USA. ACM.

Kamel, I. (2009). A schema for protecting the integrity of databases. *Computers & Security*, 28(7):698–709.

Krawczyk, H., Bellare, M., and Canetti, R. (1997). HMAC: Keyed-Hashing for Message Authentication. RFC 2104 (Informational). Updated by RFC 6151.

Li, F., Hadjieleftheriou, M., Kollios, G., and Reyzin, L. (2006). Dynamic authenticated index structures for outsourced databases. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, SIGMOD '06, pages 121–132, New York, NY, USA. ACM.

Merkle, R. C. (1989). A certified digital signature. In Brassard, G., editor, *CRYPTO*, volume 435 of *Lecture Notes in Computer Science*, pages 218–238. Springer.

Miklau, G. and Suciu, D. (2005). Implementing a tamper-evident database system. In *Proceedings of the 10th Asian Computing Science conference on Advances in computer science: data management on the web*, ASIAN'05, pages 28–48, Berlin, Heidelberg. Springer-Verlag.

Mykletun, E., Narasimha, M., and Tsudik, G. (2006). Authentication and integrity in outsourced databases. *ACM Transactions on Storage*, 2(2):107–138.

Narasimha, M. and Tsudik, G. (2005). Dsac: integrity for outsourced databases with signature aggregation and chaining. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, CIKM '05, pages 235–236, New York, NY, USA. ACM.

Pugh, W. (1990). Skip lists: a probabilistic alternative to balanced trees. *Commun. ACM*, 33(6):668–676.

Samarati, P. and Capitani, S. D. (2010). Data Protection in Outsourcing Scenarios : Issues and Directions. *ACM Symposium on Information, Computer and Communications Security*.

Avaliação da Qualidade em Linked Datasets: uma abordagem com foco nos requisitos da aplicação

Aluno: Walter Travassos Sarinho[1]

Orientadora: Bernadette Farias Lóscio[1]

Co-Orientadora: Damires Souza[2]

[1]Programa de Pós Graduação em Ciências da Computação – Centro de Informática (CIn) – Universidade Federal de Pernambuco (UFPE)
Recife – Pernambuco – Brasil

{wts, bfl}@cin.ufpe.br

[2]Instituto Federal de Educação, Ciência e Tecnologia (IFPB)
João Pessoa – Paraíba – Brasil

damires@ifpb.edu.br

Nível: Mestrado

Ano de Ingresso no Programa: 2012

Época Esperada de Conclusão: Fevereiro de 2014

Etapas Concluídas: Referencial Bibliográfico (Dez/2012), Definição do Problema (Mar/2013), Definição da Arquitetura (Abr/2012), Especificação dos componentes (Jun/2012).

Etapas Futuras: Implementação da Arquitetura (Ago/2013), Experimentação (Set/2013), Escrita da Dissertação (Dez/2013), Defesa da Dissertação (Fev/2014)

Abstract. *The increasing availability of datasets on the Web of Data faces some new challenges. One of them regards how to evaluate the quality of these datasets, whose solution may use Information Quality criteria. Most quality criteria are abstract concepts and need to be calibrated within a good situation before being used to evaluate any task. In this context, this work proposes an approach to evaluate the quality of linked datasets, i.e., datasets published according to the principles of Linked Data. Furthermore, we intend to evaluate the quality considering the application´s requirements. One distinguishing issue of our approach is the use of an extensible repository of quality criteria, which can be defined by a domain expert of the application. As a result, a quality score is generated and a ranking of the linked datasets is produced. This classification intends to serve as a comparative metric for the selection of the best datasets to be used in the process of rewriting queries and data integration.*

Keywords*: Information Quality, Linked Datasets, Semantic Web.*

1. Introdução e Motivação

A Web é um vasto repositório de dados estruturados, semi ou não estruturados, que cobrem os mais variados domínios do conhecimento. De maneira mais específica, destacam-se os conjuntos de dados disponíveis em RDF (*Resource Descripton Framework* [W3C, 2013]) e publicados de acordo com os princípios de *linked data* [Bizer *et al*., 2009], chamados de *linked datasets*. O interesse na publicação de *linked datasets* é crescente, uma vez que a natureza estruturada e o uso de vocabulários incrementam o nível semântico do dado facilitando sobremaneira o processamento deles por agentes de software. Por um lado, o imenso crescimento na disponibilidade desses dados tem motivado a ideia de desenvolver aplicações que façam uso de múltiplas *linked datasets* [Lóscio *et al*., 2012], por outro, este grande volume de dados e a falta de informações suficientes sobre as fontes trouxeram à tona um grande desafio: a avaliação da qualidade destes *datasets*.

Neste panorama, este trabalho tem como objetivo propor uma solução para a avaliação da qualidade de *linked datasets* de acordo com os requisitos de uma aplicação que consulta dados em múltiplas fontes de dados. Especificamente, *dada uma aplicação A, um conjunto de fontes linked data* $S=\{S_1, S_2,..., S_n\}$ *e um conjunto de consultas* $Q = \{Q_1,..., Q_m\}$*, que a aplicação deseja responder a partir dos dados disponíveis em S*, a abordagem proposta permite *calcular a qualidade das fontes de dados em S* levando em consideração as *consultas em Q*, bem como os demais *requisitos* considerados relevantes do ponto de vista do usuário ou da aplicação. Para mensurar a qualidade de uma fonte de dados, serão utilizados critérios de qualidade consagrados na literatura, como, por exemplo, disponibilidade, precisão e corretude [Zaveri *et al*., 2012; Wang *e* Strong, 1996]. Como resultado do processo de avaliação será obtida uma classificação ordenada das fontes, com suas respectivas medidas de qualidade, que serão obtidas a partir dos requisitos da aplicação e dos critérios de qualidade configurados.

Uma característica destacada nesta proposta é a utilização de um repositório de critérios de qualidade extensível e adaptável que contém diversos critérios para avaliação de *linked datasets*, passíveis de serem configurados por um especialista do domínio da aplicação. Para isso, considera-se que toda aplicação pertence a um domínio de dados, como, por exemplo, dados bibliográficos e dados governamentais. A proposta de ter um repositório adaptável é justificada pelo fato de existirem diversos domínios do conhecimento onde um determinado critério de qualidade pode ser considerado mais importante pelo especialista naquele domínio do que outro critério. O especialista de domínio deve conhecer a proposta da aplicação e seus requisitos para elencar corretamente quais critérios de qualidade devem ser usados, pois o uso indiscriminado de tais critérios de qualidade pode melhorar ou piorar a classificação da qualidade das fontes.

Como principais diferenciais da abordagem proposta, destacam-se: (i) Tem como foco a aplicação, ou seja, considera os requisitos da aplicação na avaliação da qualidade das fontes de dados; (ii) Faz uso de um repositório de critérios de qualidade extensível e configurável e (iii) É adaptável a diferentes domínios do conhecimento.

O restante deste trabalho está organizado como segue: a Seção 2 apresenta a fundamentação teórica; a Seção 3 descreve-se a caracterização da contribuição. Na Seção 4, alguns trabalhos relacionados são abordados e, por fim, na Seção 5, a avaliação dos resultados e estado atual do trabalho são apresentados.

2. Fundamentação Teórica

Qualidade da Informação (QI) é comumente definida como um conjunto de critérios ou dimensões utilizados para indicar o grau de qualidade geral de uma informação obtida por um sistema [Batista 2008; Wang e Strong 1996]. Na literatura, QI é definida como "adequação ao uso" [Wang e Strong, 1996], o que nos leva a considerar que, a informação é apropriada se atende a um conjunto de requisitos estabelecidos, seja por um usuário ou por um conjunto de normas. Dessa forma, o valor da informação depende da sua utilidade [Batista 2008]. Aspectos de QI incluem um conjunto de critérios, métodos de avaliação desses critérios e, normalmente, uma medição geral do grau da QI. Exemplos de critérios de qualidade são: disponibilidade (*availability*) – que verifica se a informação está disponível e alcançável para uso [Zaveri *et al.*, 2012] e precisão (*accuracy*) – mensura o quanto da informação representa corretamente um fato do mundo real [Zaveri *et al*., 2012]. Um critério de grande importância ao nosso trabalho é a completude do esquema (*schema completeness*). Este critério demanda informações sobre o que a aplicação quer consultar nas fontes de dados para se mensurar o quanto uma fonte de dados é completa para responder tais consultas [Zaveri *et al*. 2012].

Em 1996, Wang e Strong (1996) propuseram um *framework* conceitual onde foram agrupados quinze critérios de qualidade em quatro grupos iniciais que segmentam as características da QI em: Contextual, Intrínseca, Representacional e Acessibilidade. Em 2012, Zaveri *et al*. (2012) agrupou mais dois grupos (Confiança e Dinamicidade do *Dataset*) e novos critérios de qualidade ao que foi proposto inicialmente por Wang – totalizando seis grupos e vinte seis critérios de qualidade. O trabalho de Zaveri *et al*. foi realizado levando em consideração *linked datasets.*

Os dados disponíveis e padronizados nas *linked datasets* cobrem os mais diversos domínios, *e.g.* dados geográficos, publicações (dados bibliográficos) e dados governamentais. Contudo, apesar dessa padronização, existem problemas relacionados à qualidade dos dados na Web de Dados como, por exemplo: valores conflitantes entre conjuntos de dados diferentes [Mendes *et al*., 2012], diversidade dos dados [Flemming, 2010], ruído na informação e dificuldade para acessá-la [Hogan *et al*., 2012]. Tais fatores motivam a especificação de critérios de qualidade específicos para *linked datasets,* bem como o desenvolvimento de métricas adequadas para este contexto.

Outro fator importante a ser considerado na avaliação da qualidade dos dados é o domínio ao qual uma aplicação pode pertencer, o qual influencia diretamente na escolha dos critérios no processo de avaliação da qualidade. Por exemplo, o critério idade (*currency*) pode ser útil no domínio de aplicações financeiras, onde o dado deve ser tão atual quanto possível, no entanto, no domínio de dados bibliográficos ele torna-se irrelevante visto que os títulos das publicações são absolutos de acordo com sua data de inserção na fonte de dados. Também se deve considerar que aplicações distintas podem fazer parte de um mesmo domínio e possuir requisitos de aplicação heterogêneos. Isso também leva a uma possível diferença na classificação da qualidade das fontes de dados para cada uma dessas aplicações.

3. Caracterização da Contribuição

Esta seção descreve a abordagem proposta para avaliação da qualidade de *linked datasets*. A arquitetura da Figura 1 ilustra os principais componentes envolvidos no processo de avaliação proposto, os quais são descritos a seguir:

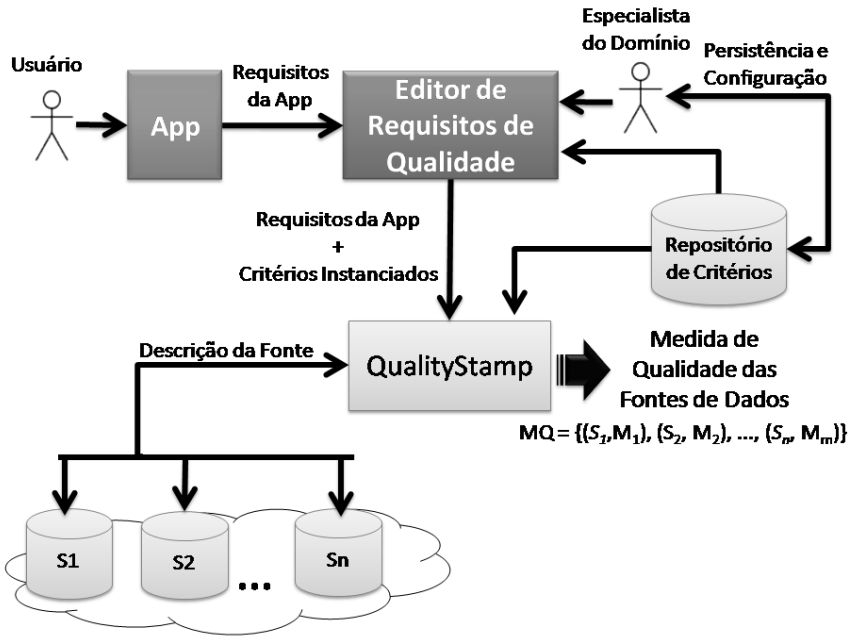


Figura 1 – Arquitetura proposta para avaliação da qualidade das fontes.

Aplicação (App): trata-se de uma aplicação que realiza consultas em um conjunto pré-definido de *linked datasets*. Nesta abordagem, considera-se que as aplicações são de domínio específico, ou seja, as fontes de dados consultadas pela aplicação ($\{S_1, S_2, ..., S_n\}$) pertencem a um mesmo domínio. Além disso, é considerada a presença de um especialista do domínio, o qual será responsável por identificar os critérios de qualidade mais adequados ao domínio em questão. Para cada aplicação, será definido um conjunto de requisitos, os quais podem ser identificados, principalmente, por meio das consultas que a aplicação executa. A partir das consultas, é possível identificar os conceitos do domínio que são relevantes para a aplicação.

Editor de Requisitos de Qualidade: o especialista do domínio, com base nos requisitos da aplicação em questão, usa o *Editor* para verificar quais critérios de qualidade estão disponíveis e configura quais serão instanciados para aquela aplicação. Além disso, é possível atribuir um peso (entre *zero* e *um*) a cada critério de qualidade. Ao escolher o peso *um*, será considerada a totalidade do critério e, optando por *zero*, o critério não será levado em consideração.

Repositório de Critérios de Qualidade: é um repositório central dos critérios de qualidade, o qual armazena os critérios e métodos para cálculo de cada um deles. Ele é extensível, ou seja, capaz de receber novos critérios e métricas. O repositório **disponibiliza** ao *Editor de Requisitos* quais critérios encontram-se disponíveis e, ao *QualityStamp,* como calcular cada critério solicitado pelo *Editor*.

QualityStamp: é o componente principal da arquitetura. Seu nome vem do conhecido selo de qualidade que pode ser atestado a um produto ou serviço. Este componente recebe como parâmetros de entrada as seguintes informações: (i) requisitos da aplicação e critérios instanciados provenientes da *App* e do *Editor de Critérios* respectivamente; (ii) métricas de cálculo obtidas a partir do *Repositório de Critérios* e (iii) Descrições das fontes *linked data* consultadas pela *App*. Como resultado final, o *QualityStamp* gera uma classificação da qualidade das fontes de dados num conjunto de pares (S_n, M_m) tal que: MQ = $\{(S_1,M_1), (S_2, M_2), ..., (S_n, M_m)\}$.

3.1. Visão geral da abordagem proposta

Como mencionado anteriormente, a abordagem proposta tem como objetivo obter uma classificação da qualidade de múltiplos *linked datasets* de acordo com os requisitos de uma aplicação específica. Para isso, inicialmente, o especialista de domínio utiliza o *Editor de Requisitos de Qualidade* para escolher quais critérios são considerados relevantes para o domínio da aplicação. Tais critérios são recuperados a partir do *Repositório de Critérios de Qualidade*. Uma vez que os critérios foram escolhidos, o próximo passo consiste em calcular os valores de cada critério para cada uma das fontes *linked data* $\{S_1,..., S_n\}$. Este cálculo é realizado pelo *QualityStamp* a partir das informações recebidas pelo do *Editor de Requisitos* e das métricas recuperadas a partir do *Repositório de Critérios de Qualidade*. No último passo, o *QualityStamp* calcula a *Medida de Qualidade da Fonte de Dados* para cada fonte *linked data* e, em seguida, apresenta uma classificação geral de qualidade dos *datasets* $\{(S_1,M_1), (S_2, M_2), ..., (S_n, M_m)\}$. A seguir, é apresentado um exemplo com o objetivo de tornar mais clara a proposta.

3.2. Um Exemplo

Como forma de ilustrar a abordagem proposta, será apresentado um exemplo onde se avalia a qualidade de um conjunto de múltiplas fontes *linked data* consultadas por uma aplicação, chamada *Frevo* (***For**m to **Ev**aluate Linked **O**pen Data Sets*). Esta aplicação pertence ao domínio de dados bibliográficos e consulta títulos de artigos a partir do nome de autores, ano de publicação e periódico de publicação do artigo. Especificamente, a aplicação recupera dados submetendo uma mesma consulta SPARQL em três fontes de dados distintas, as quais utilizam a ontologia de referência AKT[1]: IEEE[2], ACM[3] e DBLP[4]. A Tabela 1 apresenta alguns exemplos de consulta que podem ser submetidas pela aplicação.

Tabela 1. Exemplos de consultas da aplicação Frevo

Consulta Q1	**Consulta Q2**	**Consulta Q3**

SELECT DISTINCT ?titulo
WHERE {
?artigo akt:has-title ?titulo .
?artigo akt:has-author ?autor .
?autor akt:full-name **"Tim Berners-Lee"**.
}
	SELECT DISTINCT ?titulo
	WHERE {
	?artigo akt:has-title ?titulo .
	?artigo akt:has-author ?autor .
	?autor akt:full-name **"Alon Y. Halevy"**.
	?artigo akt:has-date akt-date:**1993** .
	?artigo akt:article-of-journal ?conferencia .
	?conferencia akt:has-title **"Workshop on Deductive Databases, JICSLP"**}
		SELECT DISTINCT ?titulo
		WHERE {
		?artigo akt:has-title ?titulo .
		?artigo akt:has-author ?autor .
		?autor akt:full-name **"Christian Bizer"**.
		?artigo akt:has-date akt-date:**2003** . }

Considere que para a avaliação da qualidade das fontes IEEE, ACM e DBLP os seguintes critérios, juntamente com seus respectivos pesos, foram escolhidos: (i) disponibilidade (100%); e (ii) completude do esquema (80%). A próxima etapa é o cálculo dos critérios escolhidos. O *QualityStamp* calcula os critérios de acordo com as métricas recuperadas a partir do *Repositório de Critérios de Qualidade*, como mostrado na Tabela 2.

Tabela 2. Repositório de Critérios de Qualidade

[1] http://www.aktors.org/publications/ontology/portal

[2] http://ieee.rkbexplorer.com/sparql/

[3] http://acm.rkbexplorer.com/sparql/

[4] http://dblp.rkbexplorer.com/sparql/

Critérios	Métricas	Comentários
Disponibilidade	SELECT * WHERE {?S ?P ?O} limit 1	Verifica se a fonte responde a uma consulta SPARQL. Caso positivo, a fonte está disponível (*um*). Caso ultrapasse um tempo limite de espera, a fonte não está disponível (*zero*).
Completude do Esquema	$CE = \sum C_f / \sum C_{app}$	Soma dos conceitos existentes na fonte dividido pela soma dos conceitos do conjunto de consultas da aplicação.

A Tabela 3, por sua vez, apresenta os valores dos critérios para as três fontes consideradas. No exemplo, considera-se que a fonte de dados DBLP não se encontra disponível no ato da consulta. Para o cálculo da completude, é necessário identificar os conceitos presentes nas consultas da aplicação. Conceitos podem ser identificados como os principais termos que descrevem uma consulta, os quais podem ser extraídos dos predicados presentes nos padrões de triplas da consulta. Neste caso, tem-se que: C_{Q1} = {*title, author, full-name*}, C_{Q2} = {*title, author, full-name, date, journal*} e C_{Q3} = {*title, author, full-name, date*}. Além disso, criamos um C_{app} com todos os conceitos do conjunto de consultas da aplicação tal que, consultando a descrição das fontes de dados encontram-se as seguintes intercessões de conceitos entre o conjunto de consultas da aplicação e as fontes: C_{ACM} = {*title, author, full-name, date*}, C_{IEEE} = {*title, author, date*}, C_{DBLP} = {*title, author, full-name, date, journal*}. Assim, na Tabela 3, tem-se que a fonte de dados ACM, por exemplo, tem valor de completude igual a 0,8 (*CE*= 4/5), uma vez que possui *quatro* dos *cinco* critérios relevantes para a aplicação.

Tabela 3. Valores de Critérios e Classificação das Fontes

Critério \ Fonte	ACM	IEEE	DBLP
Disponibilidade	1	1	0
Completude do Esquema	0,8	0,6	1
Classificação	**1** (0,82)	**2** (0,74)	**3** (0,40)

Por fim, a *Medida de Qualidade das Fontes de Dados* é calculada para cada fonte, usando a fórmula seguinte:

*MQ = ((critério 1 * peso 1) + (critério 2 * peso 2) + ... + (critério n * peso n)) / n*

Para a fonte de dados IEEE, por exemplo, tem-se a seguinte medida de qualidade: *MQ* = (1 * 100%) + (0,6 * 80%) / 2 = 0,74. O resultado da avaliação **(Medida de Qualidade das Fontes de Dados)** é mostrado em ordem crescente segundo a qualidade da fonte: **MQ = {(0,82 – ACM), (0,74 – IEEE), (0,40 – DBLP)}.** Como resultado conclui-se que o conjunto de dados ACM possui uma qualidade melhor de acordo com os requisitos da aplicação. Apesar do DBLP possuir uma completude do esquema de 100%, o fato de não estar disponível no momento da avaliação contribuiu negativamente para sua classificação final.

4. Trabalhos Relacionados

Em 2012, Mendes *et al*. (2012) simplificaram a tarefa de consumir dados de fontes *linked data* de alta qualidade por meio do *framework* SIEVE. A tarefa de avaliação de qualidade do SIEVE é realizada por um módulo flexível onde o usuário pode escolher quais características dos dados podem ser indicativos de uma boa qualidade, como essa qualidade pode ser quantificada e como ela deve ser armazenada no sistema. O SIEVE utiliza pelo menos *três* critérios iniciais para melhorar a qualidade dos dados: completude, concisão e consistência. O uso desses critérios permite o SIEVE remover valores conflitantes e redundantes.

Cordeiro *et al.* (2011) propuseram uma arquitetura para gerenciar e enriquecer a semântica ao publicar dados governamentais em formato *Linked Data*. O principal objetivo da proposta é propor uma plataforma que oferece suporte à exposição,

compartilhamento e associação dos recursos em formato *Linked Data* por meio de um ambiente amigável ao usuário. No processo de conversão dos dados para o formato *Linked Data* todas as etapas são monitoradas de forma a garantir a proveniência (critério de qualidade) e enriquecer a semântica da informação.

A proposta apresentada neste trabalho se diferencia das anteriores por ser uma arquitetura passível de ser expandida em fontes de dados na Web e não apenas em fontes que seguem os princípios *Linked Data*. Além disso, o repositório extensível de critérios de qualidade permite uma maior flexibilidade com relação aos domínios de aplicação e fontes de dados que podem ser considerados.

5. Avaliação dos Resultados e Estado Atual do Trabalho

Como principais atividades realizadas até o momento destacam-se o levantamento do referencial bibliográfico sobre critérios de qualidade para avaliação de qualidade de fontes *linked data* e a especificação da arquitetura proposta para avaliação da qualidade de fontes *linked data* com seus principais componentes. Para validar a proposta, um protótipo está sendo implementado com todos os componentes descritos na Figura 1. Atualmente, o foco do trabalho está na implementação do *Editor de Requisitos de Qualidade* que irá disponibilizar os critérios de qualidade para um especialista no domínio da aplicação. Como trabalho futuro tem-se a possibilidade de generalização da arquitetura para avaliação de fontes de dados na Web.

Referências

Batista, M. C. M. (2008)"Schema Quality Analysis in a Data Integration System". PhD Thesis, Universidade Federal de Pernambuco, 2008.

Bizer C., Heath T., Berners-Lee T. (2009) "Linked data – the story so far", In: Int. J. Semantic Web Inf. Syst.

Cordeiro, K.F., Faria, F.F., Pereira, B.O., Freitas, A., Ribeiro, C.E., Freitas, J.V.V.B., Bringuente, A.C., Arantes, L.O., Calhau, R., Zamborlini, V., Campos, M.L.M., and Guizzardi, G. (2011) "An approach for managing and semantically enriching the publication of Linked Open Governmental Data", In: Proceedings of the 3rd Workshop in Applied Computing for Electronic Government (WCGE), pages 82-95.

Flemming, A. (2010) "Quality characteristics of linked data publishing data sources", Master's thesis, Humboldt-Universitätzu Berlin.

Hogan, A., Harth, A., Passant, A., Decker, S., and Polleres, A. (2010) "Weaving the pedantic web", In: LDOW 2010.

Lóscio, B. F.; Batista, M. C. M.; Souza, D.; Salgado, A. C. (2012) "Using Information Quality for the Identification of Relevant Web Data Sources: A Proposal", In: iiWAS 2012.

Mendes, P., Mühleisen, H., and Bizer, C. (2012) "Sieve: Linked data quality assessment and fusion", In: Proceedings of LWDM (March 2012).

W3C (2013). Disponível em <http://www.w3.org/RDF/>. Acesso em: Agosto de 2013.

Wang, R. Y. and Strong, D. M. (1996) "Beyond accuracy: What data quality means to data consumers", In: Journal on Management of Information Systems, 12(4):5-34.

Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., Auear, A. (2012) "Quality Assessment Methodologies for Linked Open Data", In: IOS Press 2012.

Incorporando Dados Espaciais Vagos em Data Warehouses Geográficos: A Proposta do Tipo Abstrato de Dados VagueGeometry

Anderson Chaves Carniel[1],
Prof. Dr. Ricardo Rodrigues Ciferri[1]

[1]Pós-Graduação em Ciência da Computação
Departamento de Computação
Universidade Federal de São Carlos (UFSCar)
Rodovia Washington Luís, km 235 – SP-310 – 13565-905 – São Carlos – SP – Brasil

{anderson.carniel, ricardo}@dc.ufscar.br

Nível: Mestrado
Ingresso no programa: Março de 2012
Exame de qualificação: Abril de 2013
Época esperada de conclusão: Março de 2014

Abstract. *A data warehouse is a solution for organizing and storing multidimensional data related to decision-making processes in companies, generating a historical, highly voluminous, subject-oriented and nonvolatile database. A geographic data warehouse (GDW) stores spatial data (represented by crisp geometries) as attributes in dimension tables or as measures in fact tables. Thus, spatial data have exact location in the space and well-defined boundaries. However, modern geographic applications require the storage of vague spatial data, which have inaccurate location or uncertain boundaries. This master's project aims at incorporating vague spatial data to GDWs. More specifically, we address the implementation of a new abstract data type (ADT) called VagueGeometry to represent vague spatial data in the Spatial Database Management System PostgreSQL with the PostGIS extension. The proposal of the ADT VagueGeometry encompasses the issue of physical storage and the management of vague spatial data over GDW. Although there are few studies in the literature that implement vague spatial data, these studies have several limitations, for instance, the related work produces low performance in processing analytical queries with vague spatial predicates. This master's project, therefore, aims to investigate this gap in the literature of GDW related to the management of vague spatial data.*

Palavras-Chave. *Data Warehouse, Data Warehouse Geográfico, Dados Espaciais Vagos, Tipo Abstrato de Dados.*

Os autores agradecem o apoio financeiro das seguintes agências de fomento à pesquisa do Brasil: FAPESP, CNPq e CAPES.

1. Introdução

Data warehouse é um importante componente da inteligência de negócio (*business intelligence*) por ser uma base de dados voltada para a tomada de decisão visando a compreensão dos dados para elaboração de estratégias e melhorar a lucratividade nos negócios [Kimbal e Ross 2002]. Adicionalmente, dados geográficos podem ser armazenados, sendo formado um *data warehouse* geográfico (DWG) [Malinowski e Zimányi 2008]. Enquanto sobre o DW incidem consultas OLAP (*Online Analytical Processing*) para a análise das respostas [Kimbal e Ross 2002], sobre o DWG incidem consultas SOLAP (*Spatial OLAP*), que viabilizam a análise multidimensional e a análise espacial [Malinowski e Zimányi 2008].

Comumente no DWG, os dados espaciais têm natureza vetorial e usam tipos de dados geométricos para representar objetos espaciais que sejam simples, tais como ponto, linha e polígono, ou complexos, tais como multiponto, multilinha e multipolígono. Esses objetos frequentemente têm sua localização exata no espaço, ou seja, assume-se que suas coordenadas geográficas definem com clareza a posição geográfica do objeto. Além disso, uma região é representada no espaço com fronteiras bem definidas expressando com exatidão os limites da região. Tais objetos são denominados *crisp*. Contudo, fenômenos podem ter a localização inexata ou fronteiras incertas, sendo, portanto denominados objetos espaciais vagos. Na literatura, ainda se discute a respeito de padrões para a modelagem de dados espaciais vagos e para a definição de predicados topológicos envolvendo dados espaciais vagos. Consequentemente, não há suporte aos dados espaciais vagos em DWGs. De fato, também inexiste suporte nativo a dados espaciais vagos em Sistemas Gerenciadores de Banco de Dados (SGBD) Espaciais, tais como no PostgreSQL com a extensão PostGIS. Nesse sentido, a pesquisa sobre dados espaciais vagos é cada vez mais importante, desde que, com o avanço tecnológico, o uso de dados espaciais vagos em DWGs modernos é cada vez mais requerido para representar situações comumente encontradas no mundo real. Assim, este projeto de mestrado objetiva propor um novo tipo abstrato de dados (TAD), denominado *VagueGeometry*, que engloba uma forma de armazenamento interno para os dados espaciais vagos, os quais são complexos e podem possuir diversas partes disjuntas. Além de modificações na camada de processamento de consultas espaciais do SGBD PostgreSQL/PostGIS para permitir o tratamento de relacionamentos topológicos envolvendo dados espaciais vagos.

Este artigo está organizado como segue. Na seção 2 são descritos fundamentos sobre DWG e dados espaciais vagos. Na seção 3 são descritos os principais trabalhos correlatos. Na seção 4 é apresentada a proposta deste trabalho de mestrado, bem como a sua validação. Atividades realizadas e em andamento são descritas na seção 5. Por fim, na seção 6 são feitas as considerações finais e descritas as atividades futuras.

2. Data Warehouse Geográfico e Dados Espaciais Vagos

Um DWG [Malinowski e Zimányi 2008] assim como um DW convencional [Kimbal e Ross 2002] é uma base de dados histórica, integrada, orientada a assunto e não volátil que objetiva auxiliar na tomada de decisão estratégica. Um DWG pode ser implementado utilizando o modelo relacional por meio de tabelas de fatos e de dimensão. Enquanto as tabelas de fato armazenam as medidas numéricas, as tabelas de dimensão contêm os atributos descritivos que contextualizam essas medidas. Ademais,

um DWG armazena dados espaciais como atributos específicos em tabelas de dimensão ou como medidas em tabelas de fatos [Malinowski e Zimányi 2008].

Contudo, em DWGs, os atributos espaciais armazenados podem ter características de dados espaciais vagos. Existem diversos modelos de representação de dados espaciais vagos. Este trabalho se concentra na investigação de dois modelos: os modelos exatos e os modelos baseados na teoria de conjuntos *fuzzy*. Os modelos exatos tem o objetivo de reutilizar a implementação de dados espaciais *crisp* já existente, e os seus principais modelos existentes na literatura são: Egg-Yolk [Cohn e Gotss 1995], QMM [Bejaoui et al. 2009] e VASA [Pauly e Schneider 2010]. O modelo Egg-Yolk de Cohn e Gotss (1995) somente define regiões vagas o qual utiliza duas sub-regiões em sua representação, uma sub-região denominada clara (parte que engloba a vagueza) e outra denominada gema (parte que representa a exatidão), a qual está contida na clara. Já o modelo QMM (*Qualitative Min-Max model*) de Bejaoui et al. (2009) define dados espaciais vagos em dois limites, o limite mínimo (que se refere a parte que certamente pertence ao objeto espacial) e o limite máximo (que engloba o limite mínimo e o estende com a parte que possivelmente pertence ao objeto espacial). Além disso, utiliza classificações qualitativas para diferenciar "níveis de vagueza", tais como: completamente *crisp*, parcialmente vago e completamente vago. Por fim, o modelo VASA (*Vague Spatial Algebra*) de Pauly e Schneider (2010) propõe uma álgebra para definir tipos de dados espaciais vagos com base nos modelos exatos. Um dado espacial vago é definido por um par de objetos complexos *crisp* do mesmo tipo de dado. Seja $\alpha \in \{ponto, linha, região\}$, um tipo de dado espacial vago é definido formalmente como $v(\alpha) = \alpha \times \alpha$, o qual para $w = (w_n, w_c) \in v(\alpha)$ deve-se respeitar $disjunto(w_n, w_c) \vee toca(w_n, w_c)$, onde w_n é o núcleo e w_c a conjectura. Enquanto o núcleo se refere à porção conhecida e determinada, a parte da conjectura se refere à porção vaga. A álgebra VASA se torna mais ampla que os outros modelos exatos, por também definir vários operadores topológicos, numéricos e específicos para dados espaciais vagos.

Além dos modelos exatos, têm-se os modelos baseados na teoria de conjuntos *fuzzy* [Zadeh 1965]. Nesse sentido, um dado espacial vago é composto por uma função de pertinência associado a um objeto espacial que irá determinar o grau de pertinência de cada ponto do objeto espacial. Em Dilo, de By e Stein (2007) e Schneider (2008) são definidos tipos de dados espaciais *fuzzy*. Um ponto *fuzzy* contém um grau de pertinência associado a um par de coordenadas, sendo definido como $\mu P(x, y) = [0, 1]$, onde (x, y) é um par de coordenadas no espaço Euclidiano bidimensional. Uma linha *fuzzy* é definida por uma função contínua de pertinência que tem transições suaves entre seus pontos vizinhos ao longo da linha e não pode ter auto intersecção. A função contínua pode ser um homeomorfismo *h* de [0,1] para uma linha em $\mathbb{R}^2$, como definido em Dilo, de By, e Stein (2007). As regiões *fuzzy* podem representar fenômenos que contêm fronteiras indefinidas. Para uma região *fuzzy*, é definida uma função de pertinência que determina, para cada ponto, o quanto ele pertence a uma região, sendo ela contínua [Dilo, de By, e Stein 2007; Schneider 2008].

Uma representação de regiões *fuzzy* que reutiliza dados espaciais *crisp* é a região *plateau* [Kanjilal, Liu e Schneider 2010]. Cada região *plateau* é representada por uma sequencia finita de pares, onde cada par é formado por um objeto espacial *crisp* do tipo região e um grau de pertinência associado. Cada região *crisp* de uma região *plateau* é

chamada de sub-região. As sub-regiões estão topologicamente relacionadas com os predicados de "disjunto" ou "toca".

3. Trabalhos Correlatos

Na literatura existem poucos trabalhos que implementam dados espaciais vagos em sistemas gerenciadores de banco de dados (SGBD). Apesar da existência do iBLOB (*Intelligence Binary Large Objects*) de Chen et al. (2010) para definir TADs genéricos, esta estrutura não foi utilizada para definir dados espaciais vagos. Além disso, o iBLOB não foi testado exaustivamente para avaliar seu desempenho comparando com outras técnicas de implementações de TADs. Aspectos relacionados ao armazenamento de dados espaciais vagos em DWGs são investigados nos trabalhos de Siqueira et al. (2011; 2012). Testes de desempenho foram efetuados para investigar o impacto de manter dados espaciais vagos em uma única dimensão ou de separá-los em outra tabela. Já em Siqueira et al. (2012) foram propostos esquemas específicos no nível lógico de DW para permitir a representação de dados espaciais vagos a partir da implementação de dados espaciais *crisp* em SGBDs baseados em modelos relacionais. Porém, nestes trabalhos, não foi implementado um TAD específico para tratar dados espaciais vagos, o qual é o objeto deste trabalho.

Uma implementação de dados espaciais vagos baseado em modelos *fuzzy* é proposta em Dilo et al. (2004), o qual implementam ponto vago, linha vaga e região. Um ponto vago é armazenado como um tripla (x, y, λ) e uma linha vaga como um conjunto de triplas $((x_1, y_1, \lambda_1), ..., (x_N, y_N, \lambda_N))$, onde $(x, y) \in \mathbb{R}^2$ fornece a localização e $\lambda \in (0, 1]$ o grau de pertinência. Uma região vaga é composta por várias linhas vagas e pela triangulação de Delaunay. Esta implementação foi realizada no *software* GRASS e não em um SGBD. Além disso, é importante enfatizar que somente a operação de união foi implementada. Assim, os outros operadores geométricos de conjuntos intersecção e diferença, predicados topológicos e operadores numéricos não foram implementados. Limitações que não existirão neste projeto de mestrado, pois estes aspectos serão investigados e implementados no SGBD PostgreSQL.

Já em Pauly e Schneider (2007) foi implementada a álgebra VASA e implementados os predicados espaciais da seguinte forma. Um operador topológico *P* possui três funções definidas, recebendo dois objetos vagos A e B: (i) *true_P*(A, B), a qual retorna *true* se e somente se o predicado é verdadeiro e *false* caso contrário; (ii) *maybe_P*(A, B), a qual retorna *true* se e somente se o predicado talvez aconteça e *false* caso contrário; e (iii) *false_P*(A, B), a qual retorna *true* se e somente se o predicado é falso e *false* caso contrário. Esta adaptação foi necessária, pois os predicados espaciais da álgebra VASA podem retornar 3 valores lógicos: *true*, *false* ou *maybe*. Assim, o operador ~ foi proposto e que junto a um relacionamento topológico *P*, retorna *true* se o predicado com certeza ou talvez ocorra e *false* caso contrário. Porém, o operador ~ não foi implementado. Diferentemente deste projeto de mestrado, o trabalho de Pauly e Schneider (2007) não proporciona uma forma de representar os dados espaciais vagos internamente no SGBD, apenas oferecendo uma camada que adapta os operadores da álgebra para permitir o tratamento de relacionamentos topológicos e o uso de operações lógicas com três valores.

4. Proposta

4.1. Descrição

Este projeto de mestrado visa a implementação e definição de um TAD para dados espaciais vagos denominado *VagueGeometry*. Assim, pretende-se estender o SGBD PostgreSQL/PostGIS. O PostgreSQL/PostGIS foi escolhido por ser amplamente utilizado pela academia e indústria, ter um bom desempenho e ser de código fonte aberto. Mais especificamente, pretende-se investigar características de dados espaciais vagos, propor algoritmos de manipulação, e a proposta de operadores para a linguagem SQL no intuito de manipular dados espaciais vagos. Prioritariamente almeja-se implementar o TAD *VagueGeometry* usando o modelo exato da álgebra VASA e posteriormente, para o modelo baseado na teoria de conjuntos *fuzzy*.

4.2. Validação

A validação dos resultados obtidos será realizada por meio de testes de desempenho visando comparar o TAD proposto *VagueGeometry* com trabalhos correlatos existentes na literatura. Nos testes de desempenho serão considerados como fatores os tipos de dados espaciais vagos, a origem dos dados (sintético ou real), o volume dos dados, além do tipo e da seletividade das consultas. Os tipos de dados a serem usados serão dados espaciais vagos que podem ser ponto vago, linha vaga ou região vaga. Os demais fatores serão determinados ao longo do projeto.

Os testes enfocarão no uso do TAD *VagueGeometry*, o qual será implementado diretamente no SGBD PostgreSQL/PostGIS. As análises serão realizadas em termos do tempo gasto em segundos no processamento de consultas SOLAP sobre o DWG, utilizando dados espaciais vagos. Serão considerados esquemas diferentes de DWG (por exemplo, esquemas híbrido e convencional), a fim de investigar um esquema adequado para o processamento de consultas SOLAP utilizando o TAD proposto. Já com relação aos trabalhos correlatos a serem usados nos testes de desempenho, serão considerados os trabalhos descritos na seção 3, além de qualquer outro trabalho que por ventura seja proposto na literatura. Especialmente com relação ao SGBD PostgreSQL/PostGIS, pretende-se analisar formas de armazenamento e de processamento de consultas SOLAP com dados espaciais vagos reutilizando os tipos de dados espaciais definidos nesse SGBD, comparando-as com o TAD proposto.

5. Atividades em Andamento

As atividades de mestrado em andamento estão concentradas na etapa de definição e implementação do TAD *VagueGeometry* baseado no modelo *fuzzy* e na etapa de validação e refinamento do TAD *VagueGeometry* baseado no modelo exato da álgebra VASA. Assim, já existe uma versão preliminar do TAD *VagueGeometry* baseado no modelo exato da álgebra VASA que reutiliza estruturas de dados do PostGIS. Foram implementadas várias funções categorizadas como se segue: (i) **input/output**: recebe dados espaciais vagos em sua forma textual e os armazenam internamente, bem como o inverso; (ii) **métodos assessores**: edita, remove ou acessa partes dos dados espaciais vagos; (iii) **operações específicas de tipos**: manipula dados espaciais vagos de tipos pré-determinados (por exemplo, capturar a borda de regiões vagas); (iv) **operações geométricas de conjuntos**: operações de união, intersecção e diferença entre dados

espaciais vagos do mesmo tipo; (v) **operadores topológicos**: verifica os relacionamentos topológicos existentes entre dados espaciais vagos e retorna um objeto do tipo *VagueBool*, por exemplo, o predicado espacial "está contido" pode retornar *maybe*, *true* ou *false*; e, (vi) **operadores numéricos**: calculam medidas numéricas de dados espaciais vagos (por exemplo, a área de uma região vaga), bem como entre dados espaciais vagos (por exemplo, a distância entre duas regiões vagas), e retornam um objeto do tipo *VagueNumeric*, o qual contém um valor máximo e um valor mínimo. Operadores também foram propostos para manipular predicados espaciais entre dados espaciais vagos: (i) o operador unário **~**: retorna *true* se o predicado retorna *maybe* ou *true*, e *false* caso contrário; (ii) o operador unário **~~**: retorna *true* se o predicado retorna *maybe*, e *false* caso contrário; e, (iii) o operador unário **!**: retorna *true* se o predicado retorna *false*, e *false* caso contrário. Além desses operadores também foi implementado o operador binário **~** para comparar um *VagueNumeric* e um *Numeric*, o qual retorna *true* se o *Numeric* está entre o valor mínimo e o valor máximo do *VagueNumeric*. Por fim, também foram implementados os operadores da tabela verdade dos três valores lógicos da álgebra VASA: **&&** (**and**), **||** (**or**) e **!** (**not**). Assim, a validação destas operações está em fase de andamento, bem como a investigação de novos operadores e de refinamento das já existentes. A validação está na fase de configuração do ambiente para os testes, conforme descritos na seção 4.2.

Com relação a proposta do TAD *VagueGeometry* baseado no modelo *fuzzy*, a etapa em andamento é a implementação dos tipos de dados espaciais *fuzzy* e suas operações. Linhas *fuzzy* e pontos *fuzzy* foram implementados como segue: um ponto *fuzzy* é uma tripla *(x, y, u)* onde *(x, y)* é o par de coordenadas e *u* é o seu grau de pertinência; e uma linha *fuzzy* é um conjunto finito de pontos *fuzzy* e a interpolação linear é usada para calcular o grau de pertinência de um ponto na linha. Além disso, a implementação de regiões *fuzzy* está em andamento. A representação de região *fuzzy* considerada é a região *plateau*. As operações geométricas de conjuntos estão em desenvolvimento também. Os últimos aspectos a serem considerados são os operadores topológicos e numéricos, e a etapa da validação, conforme descrito na seção 4.2.

6. Considerações Finais e Próximas Atividades

Dados espaciais vagos são relevantes para a representação de vários fenômenos que contém fronteiras incertas ou localização inexata. Entretanto, no melhor do nosso conhecimento, inexiste um TAD responsável por manipular dados espaciais vagos bem como seus relacionamentos em um SGBD. Trabalhos correlatos que implementam este tipo de dado são limitados e incapacitam seu uso em contextos como a execução de consultas em DWGs com dados espaciais vagos. Dessa forma, este projeto de mestrado concentra-se na proposta de um TAD, denominado *VagueGeometry*, visando sua incorporação em DWGs.

As próximas atividades a serem realizadas envolvem a finalização da proposta do TAD *VagueGeometry* baseado no modelo *fuzzy*, bem como o refinamento da proposta do TAD *VagueGeometry* baseado no modelo exato da álgebra VASA. Em seguida, será focada a etapa de validação dessas propostas por meio de comparação com trabalhos correlatos ao se executar consultas SOLAP sobre DWGs com dados espaciais vagos. Neste sentido, está sendo submetido um pedido de bolsa BEPE para a FAPESP visando o estágio no exterior na University of Florida com o Prof. Markus Schneider. O objetivo deste estágio é usar o iBLOB para incorporar dados espaciais vagos e comparar

esta nova implementação com as implementações do TAD VagueGeometry propostas neste mestrado.

Referências

Bejaoui. L., Pinet, F., Bédard, Y. and Schneider, M. (2009) "Qualified topological relations between spatial objects with possible vague shape," International Journal of Geographical Information Science 23(7), p. 877-921.

Chen, T., Khan, A., Schneider M. and Viswanathan, G. (2010) "iBLOB: Complex Object Management in Databases Through Intelligent Binary Large Objects," In 3rd Int. Conf. on Objects and Databases, p. 85-99.

Cohn, A. G. and Gotts, N. M. (1995) "The Egg-yolk Representation of Regions with Indeterminate Boundaries," In P. A. Burrough, & A. U. Frank, Geographic Objects with Indeterminate Boundaries - GISDATA 2, p. 171-187.

Dilo, A., de By, R. A. and Stein, A. A. (2007) "A System of Types and Operators for Handling Vague Spatial Objects," International Journal of Geographical Information Science. v. 21, n. 4, p. 397-426.

Dilo, A., Kraipeerapun, P., Bakker, W. and de By, R. A. (2004) "Storing and handling vague spatial objects," In 15th Int. workshop on database and expert systems applications. p. 945-950.

Kanjilal, V., Liu, H. and Schneider, M. (2010) "Plateau Regions: An Implementation Concept for Fuzzy Regions in Spatial Databases and GIS," In 13th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems, p. 624-633.

Kimball, R. and Ross, M. (2002) "The Data Warehouse Toolkit". Wiley, 2nd ed.

Malinowski, E. and Zimányi, E. (2008). "Advanced Data Warehouse Design: From Conventional to Spatial and Temporal," Springer Publishing Company, Inc. 444 p.

Mateus, R., Siqueira, T., Times, V., Ciferri, R. and Ciferri, C. (2010) "How Does the Spatial Data Redundancy Affect Query Performance in Geographic Data Warehouses?," Journal of Information and Data Management, v.1, n.3, p. 519-534.

Pauly, A. and Schneider M. (2010). "VASA: An algebra for vague spatial data in databases," Inf. Syst. 35(1), p. 111-138.

Pauly, A. and Scheineder M. (2007) "Querying vague spatial objects in databases with VASA". In Int. Symposium on Spatial Data Quality.

Schneider, M. (2008) "Fuzzy Spatial Data Types for Spatial Uncertainty Management in Databases," Handbook of Research on Fuzzy Information Processing in Databases. p. 490-515.

Siqueira, T., Ciferri, C., Times, V. and Ciferri, R. (2012) "Towards Vague Geographic Data Warehouses," In 7th Int. conference GIScience. p. 173-186.

Siqueira, T., Mateus, R., Ciferri, R., Times, V. and Ciferri, C. (2011) "Querying Vague Spatial Information in Geographic Data Warehouses", In Advanced Geoinformation Science for a Changing World. p. 379-397.

Zadeh, L. A. (1965) "Fuzzy Sets," Information and Control, v.8, p. 338-353.

ImageDW-index: Uma estratégia de indexação voltada ao processamento de imagens em data warehouses

Jefferson William Teixeira[1],
Profa. Dra. Cristina Dutra de Aguiar Ciferri[1]

[1] Pós-Graduação em Ciências de Computação e Matemática Computacional
Instituto de Ciências Matemáticas e de Computação (ICMC)
Universidade de São Paulo (USP)
São Carlos, SP, Brasil

{william,cdac}@icmc.usp.br

Nível: Mestrado
Ano de ingresso no programa: 2012
Exame de qualificação: Abril de 2013
Época esperada de conclusão e defesa: Abril de 2014

Abstract. *A data warehousing environment offers support to the decision-making process. It consolidates data from distributed, autonomous and heterogeneous information sources into one of its main components, the data warehouse. Furthermore, it provides efficient processing of analytical queries (i.e. OLAP queries). A conventional data warehouse stores only alphanumeric data. On the other hand, an image data warehouse stores not only alphanumeric data but also intrinsic features of images, thus allowing non-conventional data warehousing environments to perform OLAP similarity queries over images. This requires the development of strategies to provide efficient processing of these complex and costly queries. In our master's research, we focus on this issue. We are developing the ImageDW-index, an index strategy aimed at the efficient processing of analytical queries extended with image similarity predicates. Although there are a number of approaches in the literature that propose indices for data warehouses and indices for image data separately, to the best of our knowledge, there is not an approach that investigate these two issues in the same setting. Therefore, our master's research aims to investigate this gap in the literature.*

Palavras-Chave*: data warehouse de imagens, consultas por similaridade, consultas OLAP, índices*

1. Introdução

Um ambiente de *data warehousing* (DWing) consolida dados de interesse de provedores de informação autônomos, distribuídos e heterogêneos em único banco de dados, o *data warehouse* (DW). Esse ambiente garante eficiência e flexibilidade na recuperação de informações estratégicas voltadas aos processos de gerência e de tomada de decisão [Chaudhuri and Dayal 1997]. Nesse ambiente, consultas analíticas, denominadas OLAP *(on-line analytical processing)*, são executadas diretamente no DW, sem acesso aos provedores originais.

Usualmente a modelagem de um DW segue o esquema estrela, o qual, em um ambiente relacional, consiste de uma tabela de fatos que armazena as medidas numéricas de interesse do DW, bem como referências às várias tabelas de dimensão, as quais contextualizam essas medidas. Por exemplo, em um ambiente de DWing para a área médica, dados relativos à incidência de câncer de mama (medida numérica) podem ser integrados segundo diferentes hospitais e faixas etárias ao longo dos anos (dimensões), oferecendo suporte para a execução de consultas tais como "Qual a incidência de câncer de mama nos últimos três anos em diferentes hospitais, considerando diferentes faixas etárias?".

DWs convencionais armazenam apenas dados alfanuméricos. Entretanto, tem-se estudado a incorporação de dados complexos (imagens) ao ambiente de DWing, de forma a permitir que usuários de sistemas de suporte à decisão (SSD) explorem uma nova gama de consultas analíticas que envolvam comparação de imagens. Por exemplo, certa equipe médica pode estar interessada na seguinte consulta "Qual a quantidade de imagens similares a uma determinada imagem de câncer de mama ocorreram nos últimos três anos no Hospital das Clínicas de Ribeirão Preto?".

Um DW de imagens armazena não somente dados alfanuméricos, mas também dados relacionados a imagens. Por ser uma área de pesquisa recente, ainda não existe um consenso sobre a definição de um DW de imagens com relação ao esquema, aos dados e ao processamento de consultas. Neste artigo, é usada a definição introduzida em [Annibal et al. 2010], a qual considera que um DW de imagens é projetado segundo um esquema estrela diferenciado pois, além de possuir tabelas de dimensão com dados convencionais, esse esquema também possui uma ou mais tabelas de dimensão especificamente voltadas à manipulação de imagens por meio de vetores de características.

Uma questão importante no desenvolvimento desses ambientes não convencionais de DWing refere-se à necessidade de processamento eficiente de consultas analíticas estendidas com predicados de similaridade de imagens. Embora na literatura existam abordagens voltadas à indexação em ambientes de DWing convencionais e à indexação de imagens em bancos de dados complexos, essas propostas apresentam como limitação o fato de não considerarem essas duas áreas de pesquisa conjuntamente.

O projeto de mestrado visa suprir essa limitação existente na literatura, por meio da proposta do ImageDW index, uma estratégia de indexação voltada ao processamento eficiente de consultas analíticas estendidas com predicados de similaridade de imagens. A estratégia em desenvolvimento utiliza conceitos bem difundidos de DW (ex.: índice bitmap de junção) e de armazenamento e recuperação de imagens (ex.: técnica Omni).

Esse artigo está estruturado da seguinte forma. Na seção 2 é descrita a fundamentação teórica, na seção 3 são resumidos os trabalhos correlatos, e na seção 4

é detalhada a proposta do ImageDW-index e descrito o estágio atual de desenvolvimento do trabalho. O artigo é concluído na seção 5, com as considerações finais e próximas atividades a serem desenvolvidas.

2. Fundamentação Teórica

2.1. Índice Bitmap de Junção

Um índice bitmap consiste de vários vetores de bits, cada um construído para um valor do domínio de um atributo. Cada entrada desses vetores faz referência a uma tupla da base de dados e contém o bit "1" se a tupla original possui o valor representado, ou o bit "0", caso contrário [O'Neil and Quass 1997]. Como resultado, operações lógicas bit a bit são realizadas rapidamente pelos processadores. Técnicas de codificação e compressão também são usadas para melhorar o desempenho de índices bitmaps. Ademais, a técnica de *binning* reduz o tamanho do índice criando-se grupos de identificadores pelos quais os valores de um atributo são organizados [Wu et al. 2008].

O índice bitmap é usado em ambientes de DWing convencionais para evitar a necessidade de se realizar operações de junção entre as tabelas de fatos e as tabelas de dimensão no processamento de consultas OLAP. Nesse sentido, para cada atributo de cada tabela de junção, um índice bitmap de junção [O'Neil and Graefe 1995] pode ser construído para indicar o conjunto de tuplas da tabela de fatos que faz junção com os valores daquele atributo. Índices bitmap são adequados a bases de dados do tipo *read-only*, como é o caso de DWs, devido ao alto custo de atualização desse tipo de índice.

2.2. Espaço Métrico e Técnica Omni

Dados complexos podem ser modelados em um espaço métrico, o qual é um par ordenado $(\mathcal{U}, d)$, onde $\mathcal{U}$ é um conjunto de objetos e $d : \mathcal{U} \times \mathcal{U} \rightarrow \mathbb{R}^+$ é uma métrica, isto é, uma função de distância que mede o grau de (dis)similaridade entre os objetos e obedece às propriedades de identidade, simetria, não-negatividade e desigualdade triangular [Ciaccia and Patella 2002]. Nesse espaço, podem ser realizadas consultas por similaridade, dentre as quais tem-se as consultas por abrangência, as quais retornam os elementos contidos em um raio de tolerância centrado no objeto de consulta. Para possibilitar a execução de consultas por similaridade, aplicações que manipulam dados complexos referentes ao domínio de imagens compõem e armazenam vetores de características. Ou seja, atributos como forma, textura e cor, entre outros, são extraídos das imagens para a composição desses vetores, os quais representam o conteúdo visual das imagens por meio de valores numéricos.

O processamento de consultas por similaridade pode ser otimizado por meio de métodos de acesso métricos (MAMs). A técnica Omni é um MAM baseado no uso de representantes globais (focos) da base de dados [Traina et al. 2007], os quais são obtidos por heurísticas, e no armazenamento da distância de cada outro elemento da base aos representantes. Em uma consulta por abrangência, os representantes, as distâncias armazenadas e o elemento de consulta são usados para determinar uma região do espaço métrico denominada *mbOr* (*minimum-bounding-Omni-region*), na qual os elementos mais similares ao elemento de consulta residem, formando um conjunto de candidatos. Esses candidatos são então refinados calculando-se suas distâncias ao elemento de consulta.

3. Trabalhos Correlatos

Na literatura existem diversos trabalhos que propõem índices para ambientes de DWing e índices para o processamento de imagens, porém, não foram encontrados trabalhos que consideram esses dois aspectos conjuntamente. Portanto, os trabalhos correlatos a esse projeto abrangem: (i) DWs de imagens; (ii) índices para DWs convencionais e para o espaço métrico; e (iii) adaptações de índices bitmap para o processamento de consultas por similaridade.

Em [Arigon et al. 2007, Chen et al. 2008, Jin et al. 2010, Annibal et al. 2010], dados multimídia são incluídos em ambientes de DWing não-convencionais. Dentre esses trabalhos, a proposta mais abrangente e flexível é a descrita em [Annibal et al. 2010], a qual possui as seguintes características: (i) esquema estrela estendido para armazenar também vetores de características de imagens em tabelas de dimensão; (ii) a possibilidade de se usar diferentes camadas perceptuais, ou seja, diferentes vetores de características para representar cada imagem (ex.: um vetor para cor, outro para textura); (iii) estratégia de extração, tradução e carregamento de dados de imagens no DW estendido; (iv) e uso de consulta por abrangência para a execução de consultas OLAP baseadas em similaridade de imagens. Entretanto, esse trabalho não inclui a proposta de uma estratégia de indexação especificamente projetada para um ambiente de DWing de imagens, o que é o objetivo do ImageDW-index.

Considerando MAMs e índices para DWs convencionais, existem muitas propostas, por exemplo [Chmiel et al. 2009, Carélo et al. 2011]. Embora os índices para DWs melhorem o desempenho do processamento de consultas OLAP, eles não oferecem funcionalidades voltadas ao processamento de consultas por similaridade. Em contrapartida, embora os índices métricos melhorem o desempenho do processamento de consultas por similaridade de imagens, eles não enfocam características intrínsecas de ambientes de DWing, como a multidimensionalidade dos dados. A proposta do ImageDW-index visa considerar características de ambientes de DWing e de consultas por similaridade de imagens conjuntamente.

Por fim, os trabalhos de [Jeong and Nang 2004, Nang et al. 2010, Cha 2004] são adaptações de índices bitmap para o processamento de consultas por similaridade. Em [Jeong and Nang 2004], os vetores de características são representados por meio de bits, os quais indicam quais dimensões são representativas, ou seja, com valores relativamente maiores do que os das outras dimensões. Em [Nang et al. 2010], é proposta uma hierarquia de intervalos visando a criação de representações binárias dos vetores de características, as quais identificam as dimensões representativas entre dois objetos considerando cada intervalo. Nesses dois trabalhos, o índice bitmap é usado para o cálculo de uma distância aproximada entre um objeto de consulta e os elementos da base, gerando um conjunto de elementos candidatos a resposta, o qual é posteriormente refinado. Em [Cha 2004], realiza-se o agrupamento dos dados da base para cada dimensão dos objetos, identificando vários intervalos por dimensão. Índices bitmap são criados para cada *cluster*, indicando quais objetos pertencem aos intervalos encontrados. Dois pontos classificados em um mesmo intervalo são considerados similares naquela dimensão. Entretanto, nenhum desses trabalhos considera as especificidades de ambientes de DWing, como a grande quantidade de dados e sua organização multidimensional. Em especial, o volume de dados impacta de forma negativa nesses trabalhos, principalmente devido ao

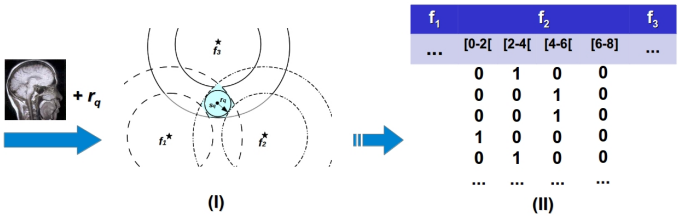
alto custo de construção desses índices. A proposta do ImageDW-index visa enfocar esses ambientes, e suas características intrínsecas.

4. Proposta e Estágio Atual de Desenvolvimento

4.1. Descrição do ImageDW-index

Como ponto de partida, está sendo investigado o armazenamento de intervalos fixos de distâncias dos vetores de características das imagens aos representantes da base de dados, aproveitando as vantagens oferecidas pela técnica Omni. Nessa abordagem, os representantes globais possuem um conjunto fixo de intervalos de distância, cada qual com seu respectivo vetor de bits indicando a pertinência dos objetos da base aos intervalos representados. Dessa forma, a intersecção dos intervalos para formação da *mbOr* é realizada de maneira muito mais rápida.

Uma ilustração do ImageDW-index é feita na Figura 1. Em uma busca por abrangência, dada uma imagem de consulta s_q e um raio de tolerância r_q, primeiramente calcula-se a distância de s_q aos representantes (f_1, f_2, f_3) para definir os intervalos em torno de cada representante (anéis). Para f_2, por exemplo, tem-se o intervalo $[d(f_2, s_q) - r_q, d(f_2, s_q) + r_q]$. Utilizando o ImageDW-index, os objetos pertencentes à *mbOr* (interseção dos anéis de cada representante) são encontrados rapidamente por meio de operações lógicas bit a bit. A etapa de refinamento é realizada posteriormente sobre o conjunto de elementos candidatos retornados.



f_1	f_2			f_3
...	[0-2[	[2-4[	[4-6[	[6-8]	...
	0	1	0	0	
	0	0	1	0	
	0	0	1	0	
	1	0	0	0	
	0	1	0	0	
	...	...	...	...	

Figura 1. ImageDW-index para geração otimizada da *mbOr* (I), dada uma imagem de consulta e um raio de tolerância r_q. Cada elemento representativo (f_1, f_2, f_3) possui intervalos de distância, para os quais são gerados os índices bitmap, conforme ilustrado em (II).

4.2. Validação do ImageDW-index

Foram realizados testes de desempenho preliminares usando um conjunto de 131.656 imagens médicas, previamente processadas por cinco descritores diferentes, ou seja, foram geradas cinco diferentes camadas perceptuais para cada imagem. As imagens foram disponibilizadas pelo Grupo de Banco de Dados e Imagens (GBdi) da USP-São Carlos e os dados convencionais foram obtidos do site `www2.datasus.gov.br/datasus`. Dados sintéticos também foram gerados, utilizando o *benchmark* TPC-H [Poess and Floyd 2000]. Os testes foram realizados em uma máquina com processador AMD Phenon II Quad-Core, 4GB de memória RAM, rodando o sistema operacional Linux (Ubuntu 12.04). Ademais, foi utilizada a implementação de índices bitmap da biblioteca *open-source* FastBit [Wu 2005].

Foram definidas duas configurações para o ImageDW-index: **Conf1**, a qual considerou apenas a indexação dos intervalos de distâncias; e **Conf2**, a qual considerou a indexação dos intervalos de distâncias, bem como a de atributos convencionais usando também índice bitmap. A indexação das distâncias aos representantes globais foi feita utilizando a técnica de *binning*. Para a etapa de refinamento, foi necessário recuperar os vetores de características dos objetos na *mbOr*, dessa forma, em cada configuração foi testado se a indexação dos vetores de características foi mais vantajosa do que acessar o DW para recuperá-los.

Os testes realizados consistem da execução de várias consultas analíticas, derivadas da consulta "Quantas imagens são similares a uma dada imagem de consulta segundo um raio de abrangência de 30% nas cinco camadas perceptuais e concomitantemente são imagens geradas no hospital da macrorregião da Grande São Paulo, nos anos de 1992 e 1993, referentes a pacientes com suspeita de tumor, do estado de São Paulo e com idade entre 0 a 30 anos?" Em cada consulta, os predicados convencionais foram progressivamente eliminados de acordo com sua seletividade (do mais seletivo até a ausência total do predicado). Cada consulta foi executada 10 vezes em cada configuração e o tempo médio em segundos foi coletado.

Comparando as duas configurações, Conf2 apresentou melhores resultados em termos de tempo em segundos do que Conf1, visto que em Conf2 todos os atributos (convencionais e de imagens) são indexados, de modo que o DW é acessado apenas para recuperar os vetores de características. Ademais, em Conf1, consultas com predicados convencionais não são plenamente beneficiadas pelo índice bitmap, dessa forma, o DW deve ser acessado para filtrar os dados pelos atributos convencionais e ainda para recuperar os vetores de características para o refinamento, o que torna mais custoso o processamento dessas consultas nessa configuração.

Os mesmos testes foram realizados para comparar as configurações propostas com a estratégia de otimização de consultas definida no trabalho correlato [Annibal et al. 2010]. Conf2 apresentou melhores resultados em termos de tempo em segundos, obtendo ganhos que variaram de 20% até 62% nos testes realizados. Isso demonstrou que o ImageDW-index é capaz de prover bons resultados de desempenho no processamento de consultas OLAP que possuam predicados de similaridade de imagens. Esse bom desempenho é oriundo do uso conjunto de conceitos bem difundidos em DW, como o índice bitmap, e de conceitos bem difundidos para o armazenamento e recuperação de imagens, como a técnica Omni.

5. Considerações Finais e Próximas Atividades

Avançando no estado da arte da pesquisa em DW de imagens, é apresentado o ImageDW-index, uma estratégia de indexação voltada ao processamento eficiente de consultas analíticas envolvendo predicados de similaridade entre imagens. De acordo com o estágio atual de desenvolvimento, a abordagem une as vantagens de índices bitmap e da técnica Omni. As próximas atividades referem-se à continuidade do processo de validação da proposta atual, por meio da realização de testes de escalabilidade, submetendo as configurações propostas a grandes volumes de dados, e comparação com outros trabalhos correlatos. Pretende-se também adaptar o índice proposto em [Jeong and Nang 2004] ao ImageDW-index, de modo a criar um segundo mecanismo de filtragem.

Referências

Annibal, L., Felipe, J., Ciferri, C., and Ciferri, R. (2010). icube: A similarity-based data cube for medical images. In *CBMS*, pages 321 –326.

Arigon, A.-M., Miquel, M., and Tchounikine, A. (2007). Multimedia data warehouses: a multiversion model and a medical application. *Multimedia Tools Appl.*, pages 91–108.

Carélo, C. C. M., Pola, I. R. V., Ciferri, R. R., Traina, A. J. M., Jr, C. T., and de Aguiar Ciferri, C. D. (2011). Slicing the metric space to provide quick indexing of complex data in the main memory. *Information Systems*, pages 79 – 98.

Cha, G.-H. (2004). Efficient and flexible bitmap indexing for complex similarity queries. In Lee, Y., Li, J., Whang, K.-Y., and Lee, D., editors, *DASFAA*, pages 708–720. Springer Berlin Heidelberg.

Chaudhuri, S. and Dayal, U. (1997). An overview of data warehousing and olap technology. *SIGMOD Rec.*, pages 65–74.

Chen, M., Song, Y., Sun, Z., Chen, H., and Sang, A. (2008). Multimedia database retrieval based on data cube. In *ICALIP*, pages 1265 –1269.

Chmiel, J., Morzy, T., and Wrembel, R. (2009). Hobi: Hierarchically organized bitmap index for indexing dimensional data. In *DaWaK*, pages 87–98. Springer-Verlag.

Ciaccia, P. and Patella, M. (2002). Searching in metric spaces with user-defined and approximate distances. *TODS*, pages 398–437.

Jeong, J. and Nang, J. (2004). An efficient bitmap indexing method for similarity search in high dimensional multimedia databases. In *ICME*, pages 815 –818.

Jin, X., Han, J., Cao, L., Luo, J., Ding, B., and Lin, C. X. (2010). Visual cube and on-line analytical processing of images. In *CIKM*, pages 849–858. ACM.

Nang, J., Park, J., Yang, J., and Kim, S. (2010). A hierarchical bitmap indexing method for similarity search in high-dimensional multimedia databases. *JISE*, pages 393–407.

O'Neil, P. and Graefe, G. (1995). Multi-table joins through bitmapped join indices. *SIGMOD Rec.*, pages 8–11.

O'Neil, P. and Quass, D. (1997). Improved query performance with variant indexes. In *SIGMOD Rec.*, pages 38–49. ACM.

Poess, M. and Floyd, C. (2000). New tpc benchmarks for decision support and web commerce. *SIGMOD Rec.*, pages 64–71.

Traina, Jr., C., Filho, R. F., Traina, A. J., 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. *VLDB*, pages 483–505.

Wu, K. (2005). Fastbit: an efficient indexing technology for accelerating data-intensive science. *JPCS*, page 556.

Wu, K., Stockinger, K., and Shoshani, A. (2008). Breaking the curse of cardinality on bitmap indexes. In *SSDBM*, pages 348–365. Springer-Verlag.

Utilizando Regras baseadas no Contexto para Reescrever Consultas

Aluno: Antônio Ezequiel de Mendonça
E-mail: aem@cin.ufpe.br

Orientadores
Ana Carolina Salgado
E-mail: acs@cin.ufpe.br
Damires Yluska de Souza Fernandes
E-mail: damires@ifpb.edu.br

Nível – Mestrado
Dissertação de Mestrado em andamento
Ingresso: março/2012
Conclusão prevista: fevereiro/2014
Etapas Concluídas: Créditos, Seminário de Acompanhamento, Definição do Problema, Especificação, Implementação de Protótipo.
Etapas Futuras: Escrita da Dissertação, Finalização da Implementação, Realização de Experimentos.
Previsão de Defesa: janeiro/2014.

Universidade Federal de Pernambuco – UFPE
Programa de Pós Graduação em Ciência da Computação
Área de Concentração: Ciência da Computação.
Linha de Pesquisa: Banco de dados, Contexto Computacional.

Abstract
When users access applications, they aim to obtain useful information. Sometimes, however, the user needs to reformulate submitted queries several times and go through many answers until a satisfactory set of answers is achieved. In this scenario, the user may be in different contexts, and these contexts may change frequently. A context surrounding his tasks (e.g., queries), for instance, may be built from his specific interests, location or expertise. In this work, we address the issue of rewriting queries considering the context acquired at query submission time. To this end, we propose a query rewriting approach, which makes use of context-based rules to produce new related expanded or relaxed queries. In this paper, we describe our approach and present some results we have obtained.

Keywords
Query Rewriting; Context; Rules.

1. Introdução e Motivação

O desenvolvimento de aplicações computacionais cada vez mais complexas e, ao mesmo tempo, adaptáveis e flexíveis, fez surgir a necessidade de mecanismos que pudessem ajudar no desenvolvimento destas. Um desses mecanismos diz respeito à utilização do *contexto* no qual uma interação entre um usuário e a aplicação acontece. Aplicações em geral fazem uso intensivo de dados e lidam com diversos tipos de usuários que interagem em momentos diferentes, por vezes, utilizando dispositivos ou ambientes também diferentes. Nessa perspectiva, o contexto pode ser usado como recurso computacional que possibilita às aplicações adaptarem-se às necessidades específicas de cada usuário ou ambiente em questão. Para Bolchini *et al.* (2009), o contexto é definido como um conjunto de variáveis que podem ser de interesse de um agente, podendo então influenciar as ações em uma dada tarefa.

Em aplicações que fazem uso intensivo de dados, às vezes uma consulta submetida pelo usuário pode conter uma descrição incompleta das informações de que ele necessita. Até mesmo quando a descrição é bem definida o mecanismo de consulta pode não ser capaz de retornar respostas que o usuário deseja. Nestes casos, argumentamos que o contexto pode ser utilizado para proporcionar a reescrita das consultas, de modo que respostas mais adequadas à necessidade do usuário possam ser retornadas. Com isto em mente, propomos uma abordagem de reescrita de consultas chamada CORE – ***Co****ntext-based* ***R****ules for r****E****writing*, que oferece expansão e/ou relaxamento da consulta de acordo com o contexto adquirido.

Para se trabalhar com o contexto, este precisa ser identificado e gerenciado. Particularmente em aplicações que usam SGBDs, algumas estratégias de gerenciamento de contexto vêm sendo utilizadas: (i) a nativa que consiste do desenvolvimento de um código dentro do próprio SGBD, usando operadores físicos e algoritmos específicos [Levandoski *et al.* 2010] e (ii) a externa que, por meio de um *plug-in* ou de um *middleware*, permite a tradução de uma consulta, sem modificar a implementação original do SGBD [Koutrika *et al.* 2010]. A abordagem CORE é baseada na segunda solução e é parte de uma arquitetura proposta por Maciel *et al.* (2013), que visa fornecer recursos de sensibilidade ao contexto em um SGBD. Este artigo tem como foco apresentar a abordagem CORE que vem sendo desenvolvida por meio de um componente de reescrita de consultas desta arquitetura de referência. Na CORE, utiliza-se consultas relacionais no padrão SQL92[1] e regras de produção [Newell *et al.* 1973] baseadas no contexto capturado. No processo de reescrita de consultas, o usuário acessa uma aplicação e submete consultas que serão alteradas de acordo com o contexto identificado. Dessa forma, o usuário irá receber respostas mais relevantes associadas a seu contexto.

Particularmente, definimos o nosso problema da seguinte forma: *Dada uma consulta do usuário Q, expressa em SQL, nosso objetivo é gerar uma consulta reescrita Q', a qual é semanticamente relacionada à consulta original Q. A relação semântica entre as duas consultas é determinada pelo contexto adquirido, que é especificado por meio de regras.*

O processo de criação de regras é desempenhado por um *especialista no domínio* (ED) da aplicação que expressa seu conhecimento por meio de regras de produção. Os elementos que formam a condição da regra são denominados *elementos contextuais*

[1] http://www.sqlteam.com/.

(EC) e representam os fatos a serem usados pelo motor de inferência. Os ECs referem-se a qualquer dado, informação ou conhecimento que permite caracterizar uma entidade em um domínio [Vieira *et al.* 2011]. Por exemplo, *localização* e *preferências* são elementos contextuais associados a uma entidade *Usuário*.

Há algumas maneiras em que a nova consulta Q' pode ser semanticamente relacionada com a consulta original Q. Neste trabalho, duas técnicas são utilizadas: (i) *expansão de consultas*, onde é realizada a adição de novos termos adquiridos a partir do processamento das regras baseadas em contexto; (ii) *relaxamento de consultas*, onde se faz uma redução do escopo de consultas através da remoção de alguns de seus termos, a fim de otimizar a consulta e reduzir o número de respostas incorretas ou redundantes. Para viabilizar o processo de reescrita, um conjunto de operadores (e.g., *trunk, order_value, etc*) vem sendo especificado. Estes operadores são utilizados na definição das regras e visam facilitar o processo de confecção das mesmas. Com base nas definições da abordagem, foi implementado uma versão inicial da *CORE*, juntamente com uma aplicação front-end. Estes tópicos serão abordados ao longo deste artigo.

Este artigo está organizado como segue: a Seção 2 introduz alguns conceitos básicos; a Seção 3 caracteriza as contribuições do trabalho; a Seção 4 apresenta a avalia os resultados obtidos até o momento. A Seção 5 descreve alguns trabalhos relacionados. A Seção 6 tece algumas considerações e indica o desenvolvimento necessário para a conclusão.

2. Fundamentação Teórica

Para Godfrey e Gryz (1996), *reescrita de consultas* é uma técnica que utiliza algum tipo de conhecimento semântico, como caches semânticas, visões materializadas ou o conhecimento do domínio, a fim de realizar uma tradução da consulta. Ao reescrever uma consulta Q em uma consulta Q', técnicas de enriquecimento ou relaxamento podem ser empregadas. Assim, a *expansão de consultas* é definida por Andreou *et al.* (2005) como um processo de inclusão de novos termos em uma consulta submetida pelo usuário, com a finalidade de melhorar as respostas obtidas. Por outro lado, o *relaxamento de consultas* refere-se ao processo ao qual a consulta é simplificada por restrições de enfraquecimento das expressões que são responsáveis por falhas [Stuckenschmidt *et al.* 2005]. O objetivo é generalizar uma consulta que contém falhas, criando uma consulta mais eficiente, eliminando ou relaxando algumas restrições da consulta original.

Segundo Dey e Abowd (2000), contexto se refere a qualquer informação que caracteriza a situação de uma entidade, onde uma entidade é uma pessoa, lugar ou objeto considerado relevante para a interação entre um usuário e uma aplicação. O contexto pode ser utilizado para ampliar o conhecimento que se tem sobre uma determinada situação, desempenhando um papel importante em qualquer domínio que envolva requisitos como compreensão, raciocínio, resolução de problemas ou aprendizado [Vieira *et al*. 2011]. Neste sentido, este trabalho busca usar o contexto identificado no momento da submissão de uma consulta, de forma associada às técnicas de expansão e relaxamento para reescrever consultas. Para viabilizar a inferência do contexto, utiliza-se regras de produção, aqui denominadas regras baseadas em contexto.

3. Caracterização da Contribuição

A abordagem CORE é parte integrante da arquitetura *Texere* [Maciel *et al.* 2013]. O objetivo da *Texere* é proporcionar características de sensibilidade contextual a um

SGBD tradicional, por meio de alguns componentes, como, por exemplo, editor de regras, definição de modelo de persistência do contexto e o componente de reescrita de consultas – foco deste trabalho.

Nesse panorama, a abordagem CORE usa as diretivas de reescrita retornadas pelo motor de inferência, a partir do processamento das regras de produção para realizar a reescrita da consulta Q. A diretiva de reescrita é um comando destinado a produzir um pedaço de código, o qual é traduzido para o padrão SQL 92. As regras criadas pelo Especialista de Domínio (ED), serão utilizadas para realizar a inferência do contexto e produzir novos fatos. Estes podem ser uma caracterização do contexto ou uma diretiva para reescrita da consulta.

O processo de reescrita faz uso de duas estratégias: expansão ou relaxamento. A decisão de qual estratégia utilizar na reescrita depende do contexto adquirido. Por exemplo, se o usuário, no momento de submissão da consulta, estiver utilizando um *smartphone*, o resultado de uma regra disparada para esse contexto é uma diretiva do tipo *revisão trunc 200*. Isso implica em um comando de expansão, pois a consulta original Q não previa essa restrição, em que o campo *revisão* deve apenas retornar os 200 primeiros caracteres.

O processo de reescrita se inicia quando o usuário conectado a uma aplicação envia uma consulta denominada Q, e esta é capturada pelo *CORE*. Ao final do processamento das regras, um conjunto de diretivas é devolvida ao *CORE*. Cada diretiva é traduzida para um comando em SQL 92, que é integrado à consulta original, gerando uma nova consulta Q'. A diretiva pode ser de dois tipos: relaxamento e expansão. Nesta, novos elementos serão incorporados à consulta original; naquela, a diretiva leva à subtração de algum elemento ou, até mesmo, o aumento da abrangência de alguma restrição. Finalmente, a consulta reescrita Q' é executada no banco de dados da aplicação e os resultados obtidos são devolvidos ao usuário. A Figura 1 ilustra uma visão geral da *CORE*. Os principais componentes são apresentados a seguir:

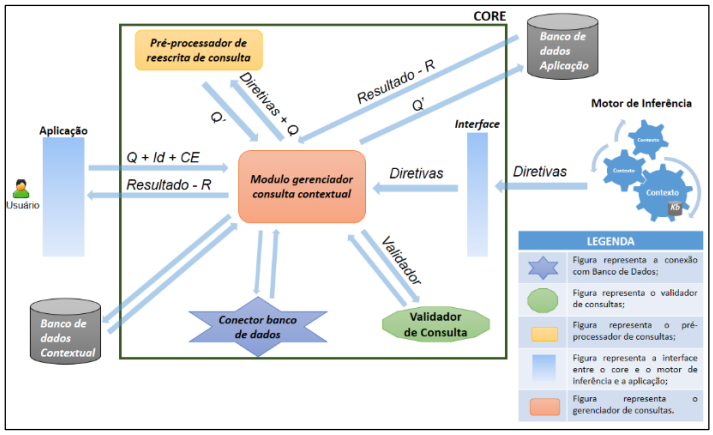


Figure 1. Principais componentes do CORE.

(I) **Conector do Banco de Dados** – gerência as conexões com os bancos de dados necessários ao processo de reescrita, como banco de dados *metadata* e o da aplicação; (II) **Validador de Consulta** – valida sintaticamente as consultas enviadas pela aplicação e a consulta final Q', resultante do processo de reescrita; (III) **Pré-processador de reescrita de consulta** – recebe a consulta Q e as diretivas, posteriormente traduz cada uma para um comando SQL padrão. Em seguida, verifica se a diretiva é uma operação de expansão ou de relaxamento. Então é executando a

reescrita da consulta Q, com a operação apropriada a diretiva traduzida; (IV) **Interface do motor de inferência** – Recebe as diretivas de reescrita de consulta enviadas pelo motor de inferência; (V) **Gerenciador de consulta contextual** – instancia os ECs, realiza a integração dos módulos do *CORE* e coordena o processo de reescrita.

Ao criar regras, o ED usa uma linguagem de alto nível. Esta linguagem é semelhante à linguagem natural e tem seus próprios operadores. Por exemplo, o operador **Trunk,** que não existe no padrão SQL, foi criado para possibilitar restringir a quantidade de caracteres retornados em uma coluna específica (ver operadores na Tabela 1). Por exemplo, em uma diretiva que contenha o operador **Trunk**, este será transformado em uma função *substring (coluna, caraterInicial, quantidadeCaracteresRetornados).*

Tabela 1 – Alguns Operadores usados no CORE.

Escrita	Ação	Exemplo
Trunk	Limita o retorno de uma quantidade x de caracteres de um campo ao submeter a consulta.	**IF** dispositivo igual celular **Then** revisão trunc 200
Order Value	Ordena a consulta por um valor específico de uma coluna.	**IF** idioma igual frances **Then** lingua order_value 'francesa'

4. Avaliação dos Resultados e Estado Atual do Trabalho

Para validar a proposta desse trabalho, foi implementado um protótipo que contempla todos os componentes da *CORE* descritos na Seção 3 foi implementado. Como estudo de caso inicial, desenvolvemos uma aplicação *front-end* de consulta a livros denominada TexereLibrary (cuja interface é mostrada na Figura 2), que usa o *CORE* para obter a reescrita das consultas com base no contexto identificado. O usuário pode realizar consultas, escrevendo comandos SQL padrão e habilitando a opção de consultas *com* ou *sem* contexto.

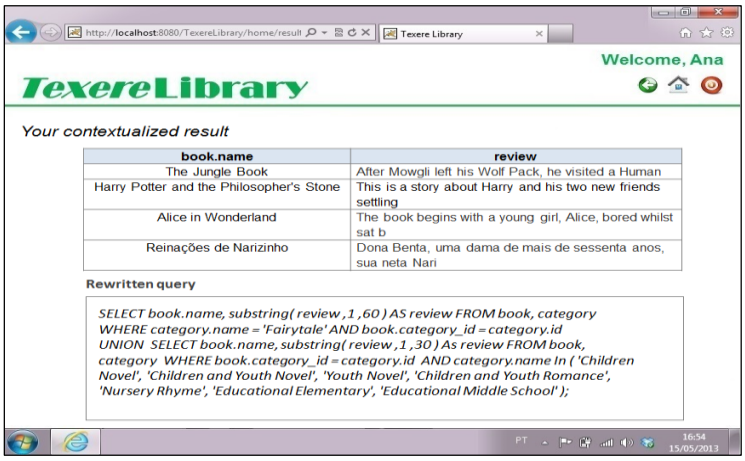


Figura 2 – Página de Resultados do TexereLibrary.

Como ilustração, suponha que o usuário Ana é uma menina de nove anos que mora no Brasil e está usando o TexereLibrary. Ana se registra na aplicação e recebe um id (user_id = 10) e uma sessão (session_id = 15). Ana está usando um *smartphone* (device = smartphone) e o mês atual é julho. Esses ECs são capturados pelo *CORE* e

armazenados. Ana então submete a seguinte consulta (SQL) Q: *Select name, review From book*. O *CORE*, após a submissão da consulta e a execução das regras definidas na Tabela 2 pelo motor de inferência, recebe as diretivas de reescrita.

Tabela 2 – Regras disparadas.

IF *mobile_device* **Then** truncate_information (revisão);
IF *user_age* < 12 **Then** show (*book_category*) in (Children Novel, Children Youth);
IF estação in (verão) **Then** *school_vacation* is true;
IF *school_vacation* is true and user_age <= 12 **Then** category order_value = ‘Fairytale’;

Após o processamento das regras e considerando as diretivas de reescrita apresentadas em cada regra na cláusula *Then*, a consulta Q’ é reescrita como segue:

Select book.name, substring(review, 1, 200) As review From book, category Where category.name = 'Fairytale' AND book.category_id = category.id Union
Select book.name, substring(review, 1, 200) As review From book, category Where book.category_id = category.id AND category.name In (‘Children Novel’, ‘Children Youth’);

Esta consulta é executada e seus resultados são retornados a Ana.

Experimentos estão sendo realizados com dois objetivos básicos: (i) obter um feedback dos usuários quanto à utilização do contexto para reescrever consultas (em termos de relevância das respostas), e (ii) verificar o desempenho da aplicação acoplada ao serviço de reescrita, em termos de tempo de resposta. O primeiro está sendo montado por meio de um formulário que será preenchido pelos usuários. O segundo já foi realizado e, diante de seus resultados, foi implementado um mecanismo de cache que melhorou sensivelmente o tempo de resposta das consultas.

5. Trabalhos Relacionados

Quanto à utilização de contexto em consultas, Amo e Pereira (2012) apresentam uma extensão da linguagem SQL denominada CPrefSQL. Para realizar a personalização, o usuário precisa informar previamente suas preferências (regras) que são do tipo "IF <contexto> THEN <preferencia>", onde <preferencia> indica o desejo do usuário em uma situação que satisfaz o <contexto>. Levandoski *et al*. (2010) trabalham com um sistema de banco de dados sensível ao contexto, implementado dentro do SGBD PostgreSQL, chamado CareDB. Este tem por objetivo redefinir a resposta de uma consulta tradicional, através da integração de vários tipos de contextos. Hachani *et al.* (2006) apresenta uma abordagem para estender os critérios de restrição de uma consulta, a fim de fornecer respostas aproximadas. A abordagem utiliza o relaxamento das restrições, bem como o contexto do usuário no processo de reescrita.

Comparando estes trabalhos, Amo e Pereira (2012) concentram-se em uma extensão de SQL. O usuário precisa registrar suas preferências (regras) em um momento imediatamente anterior ao de realizar consultas, tornando o processo de customização mais caro e estático. Em Levandoski *et al.* (2010), a abordagem exige um grande esforço para desenvolver um código nativo no SGBD, a fim de proporcionar a criação de consultas contextuais. Outra desvantagem é a impossibilidade de usar este mesmo código em outro SGBD. No trabalho de Hachani *et al.* (2006), a reescrita é feita apenas

com o relaxamento das restrições, limitando assim as possibilidades de customização da consulta. Em nossa abordagem, a geração de regras é feita pelo ED, que identifica e registra as regras sem a necessidade de desativar o serviço do *CORE*. Trabalhamos com o padrão SQL, por isso não há necessidade de alterar o algoritmo interno do SGBD relacional subjacente. Assim, aumentamos a portabilidade do sistema, reduzindo os custos com a implementação nativa e permitindo que qualquer SGBD relacional compatível com o padrão SQL 92 use o *CORE*.

6. Considerações e Desenvolvimento Necessário para Conclusão

Este trabalho apresentou a abordagem *CORE*, a qual foi especificada e uma primeira versão foi implementada. Atualmente, estamos estendendo os operadores definidos que suportam a identificação de diretivas e o processamento das regras. Além disso, após o processo de reescrita serão incorporadas técnicas de *tuning* sobre Q', a fim de melhorar o desempenho de execução da consulta reescrita.

Referências

Amo, S. and Pereira, F. (2010) "Evaluation of conditional preference queries". Journal of Information and Data Management (JIDM). v. 1, p. 521–536.

Andreou A. (2005) "Ontologies and query expansion". M.S. thesis. School of Informatics, University of Edinburgh, Edinburgh, UK.

Bolchini, C. Curino, A. Quintarelli, E. Schreiber, A. e Tanca, L. (2009) "Context information for knowledge reshaping". Int. J. Web Eng. and Tech., Milano, Italia.

Dey, A. K., Abowd, G. D. (2000) "Towards a Better Understanding of Context and Context-Awareness", In: Proceedings of the CHI 2000 Workshop on The What, Who, Where, When, and How of Context-Awareness, The Hague, Netherlands.

Godfrey, P. Gryz, J. (1996) "A framework for intensional query optimization". In DDLP, p. 57–68.

Hachani, N. Ali, M. Hassine, B. and Ounelli, H. (2009) "Cooperative Answering of Fuzzy Queries". Journal of Computer Science and Technology, p. 675-686

Levandoski J. J., Mokbel M. F., and Khalefa M. E. (2010) "CareDB: A Context and Preference -Aware Location-Based Database System". In Proceedings of the VLDB Endowment, p. 1529-1532.

Koutrika, G. (2010) "Query Personalization based on User Preferences". V. 35, Abril, New York, USA.

Newell A. (1973) "In Visual Information Processing. Academic Press". Chase E. (editor), New York, USA, p.283-308.

Maciel, Paulo. (2013) "Texere, a Context-aware System for Improving Database Queries". Technical Report, Federal University of Pernambuco, Brazil.

Stuckenschmidt H., Giunchiglia F., and van Harmelen F. (2005) "Query processing in ontology-based peer-to-peer systems". In V. Tamma, S. Craneeld, T. Finin, and S. Willmott, editors, Ontologies for Agents: Theory and Experiences. Birkhuser.

Vieira, V., Tedesco, P., and Salgado A. C. (2011) "Designing Context-Sensitive Systems: Na integrated Approach". Expert Systems with Applications 38. p. 1119-1138.

Mineração de Preferências em *Data Streams*

Programa de Pós-Graduação em Ciência da Computação
da Universidade Federal de Uberlândia

Aluna:
Jaqueline Aparecida Jorge Papini
jaque@comp.ufu.br

Orientadora:
Profª Drª Sandra Aparecida de Amo
deamo@ufu.br

Nível: Mestrado

Ingresso no Programa: 2012/01 **Previsão da Defesa:** fevereiro de 2014

Etapas Concluídas:

- Aprovação do Plano de Trabalho 03/2013
- Proposta de Estratégias para Minerar Preferências em *Data Streams* 04/2013
- Proposta do Algoritmo FPSMining para Minerar Preferências em *Data Streams* 05/2013
- Submissão de Artigo para o KDMiLe 2013 – Aceito 06/2013
- Proposta do Algoritmo IncFPSMining para Minerar Preferências em *Data Streams* 06/2013
- Submissão de Artigo para o SBBD 2013 07/2013
- Submissão de Trabalho para o WTDBD do SBBD 2013 – Aceito 07/2013

Etapas Futuras:

- Implementação de um Gerador de *Stream* de Dados Sintéticos para Preferências 08/2013
- Submissão de Artigo para o ACM SAC 2014 09/2013
- Proposta de um Algoritmo Heurístico para Minerar Preferências em *Data Streams* 10/2013
- Submissão de Artigo para o FLAIRS 2014 11/2013
- Defesa da Dissertação 02/2014

Abstract. *The traditional preference mining setting has been widely studied in the literature in recent years. However, the problem of mining preferences increasingly requires solutions that quickly adapt to change. The main reason for this is that frequently user's preferences are not static and can evolve over time. In this work, we address the problem of mining contextual preferences in a data stream setting. Contextual Preferences have been recently treated in the literature and some methods for mining this special kind of preferences have been proposed in the traditional setting. As main contribution of this work, we formalize the contextual preference mining problem in the stream setting and propose appropriate algorithms for solving this problem.*

Keywords: bayesian networks, concept drift, context-awareness, data mining, data streams, preference mining

1. Introdução

O gigantesco aumento do volume de dados digitais presenciado nos últimos anos foi parcialmente ocasionado por uma nova classe de aplicações emergentes, em que os dados são gerados a taxas muito elevadas, na forma de *data streams*. Um *data stream* pode ser visto como uma sequência de tuplas relacionais que chegam continuamente em tempo variável. Alguns dos típicos domínios de aplicação de *streams* são: mercado financeiro, aplicações web, dados de sensores. Abordagens tradicionais não conseguem processar com êxito os *streams*, principalmente devido ao seu volume de dados potencialmente infinito e a sua evolução sobre o tempo. Com isso, várias técnicas de mineração de *streams* surgiram para lidar apropriadamente com este novo formato dos dados [Domingos and Hulten 2000, Bifet and Kirkby 2009, Gama 2010].

Apesar disso, a maioria das pesquisas de mineração de preferências têm se concentrado em cenários em que o algoritmo de mineração tem a sua disposição um conjunto de informações estáticas sobre as preferências do usuário [Jiang et al. 2008, de Amo et al. 2013]. As questões mais importantes que tornam o processo de mineração de *streams* muito mais desafiador do que no cenário tradicional, conhecido como *batch*, são: (1) Os dados não são armazenados e não estão disponíveis sempre; cada tupla deve ser aceita conforme ela chega e uma vez inspecionada ou ignorada, deve ser descartada; (2) o processo de mineração precisa lidar com limitações de memória e tempo; (3) o algoritmo de mineração deve ser capaz de produzir o melhor modelo em qualquer instante que é solicitado, usando apenas os dados de treinamento que foram observados até o momento. Mais detalhes são encontrados em [Rajaraman and Ullman 2011].

O aprendizado de preferências pode ser dividido em dois problemas distintos [Fürnkranz and Hüllermeier 2011]: *label ranking* e *object ranking*. No problema de *object ranking* as preferências de um usuário são obtidas a partir das características dos objetos, enquanto que no problema de *label ranking* as preferências de um conjunto de usuários são obtidas a partir das características dos usuários. Este trabalho atua sobre o problema de *object ranking*, ou seja, dado um usuário, estamos interessados em descobrir suas preferências a partir de suas escolhas.

Este trabalho foca em um tipo particular de preferências, as *preferências contextuais*. Modelos de preferências podem ser especificados sobre um framework *quantitativo* ou *qualitativo*. Na formulação quantitativa, preferências sobre filmes (por exemplo) podem ser elicitadas por pedir ao usuário para avaliar cada filme em uma base de dados. Na formulação qualitativa, o modelo de preferências consiste em um conjunto de regras especificadas em um dado formalismo matemático, capazes de expressar as preferências do usuário. Neste trabalho são consideradas as *regras de preferência contextuais* (*cp-rules*) introduzidas em [Wilson 2004]. Uma *cp-rules* permite informar preferências sobre os valores de um atributo dependendo dos valores de outros atributos. Por exemplo, um usuário pode preferir comédias a dramas *se o diretor for Woody Allen*.

1.1. Exemplo de Motivação

Considere que um site de notícias *online* deseja aprender as preferências dos usuários sobre notícias e, com base nisso, efetuar recomendações. Um cenário típico é aquele em que o usuário se loga no site e então se depara com várias manchetes de notícias. Para obter as preferências do usuário sobre notícias sem ser inoportuno, o sistema captura de

forma automática e *online* informações implícitas de preferências geradas através de ações do usuário. Uma das formas do sistema fazer isso é através da obtenção do *stream* de cliques e consultas do usuário. Normalmente, o usuário indica as notícias que tem maior interesse por efetuar cliques sobre a manchete das mesmas ou por buscar explicitamente por uma notícia. O sistema então converte este *stream* de ações do usuário em um *stream de preferências*, onde cada elemento do *stream* indica que uma notícia é preferível a outra em um dado instante de tempo. Através do acesso ao *stream de preferências* do usuário, o sistema consegue realizar a mineração de suas preferências ao longo do tempo.

1.2. Objetivos

O objetivo principal deste trabalho é formalizar o problema de mineração de preferências contextuais no cenário de *streams* e propor algoritmos eficientes para solucionar este problema. Para alcançar esse objetivo, os objetivos específicos propostos são: (1) Introduzir o conceito de *stream de preferências* como uma forma de representação das preferências do usuário ao longo do tempo; (2) Propor um método qualitativo para a elicitação *online* das preferências do usuário sob a forma de um *stream de preferências*; (3) Introduzir o conceito de *concept drift* para o ambiente de preferências; (4) Propor 4 algoritmos para solucionar o problema abordado, cada um com um enfoque diferente; (5) Avaliar os algoritmos propostos segundo critérios de qualidade a serem adaptados para *streams*; (6) Implementar um gerador de *stream* sintético de preferências para a realização de testes nos algoritmos propostos; (7) Avaliar os algoritmos propostos sobre dados reais.

1.3. Principais Contribuições

As principais contribuições desta pesquisa para a área de mineração de preferências são: (1) Identificação e formalização sistemática de um novo problema de mineração de preferências: "A partir do *stream de preferências* do usuário, é possível realizar o aprendizado sobre as suas preferências contextuais de forma eficiente em termos de tempo e memória?"; (2) Proposta de 4 algoritmos de mineração de preferências do usuário em *data streams*; (3) Introdução do conceito de *concept drift* para o cenário de preferências; (4) Desenvolvimento de um gerador de *stream* de dados sintéticos para preferências.

2. Problema de Pesquisa

O objetivo de um método de mineração de preferências é prover uma relação de preferência sobre certo conjunto de dados. Uma relação de preferência sobre um conjunto finito de objetos $A = \{a_1, a_2, ..., a_n\}$ é uma ordem parcial estrita sobre A, que é uma relação binária $R \subseteq A \times A$ satisfazendo as propriedades irreflexiva e transitiva.

Os dados de entrada do método de mineração de preferências são representados por um *stream de preferências*, que é potencialmente infinito. Um *stream* de preferências é formado por elementos do tipo (u, v, t), que chamaremos de bitupla temporal, representando o fato de que o usuário prefere a tupla u a tupla v no instante de tempo t. Fig. 1(b) ilustra parte de um *stream* de preferências sobre o esquema relacional $R(A, B, C, D)$, representando uma amostra de suas preferências em relação às tuplas de I (Fig. 1(a)) até o instante de tempo t_7, coletada do *stream* de ações do usuário em um site.

Esse problema consiste em extrair um modelo de preferência de um *stream* de preferências em um dado instante de tempo. O modelo de preferência será especificado por uma *Rede Bayesiana de Preferência* (RBP).

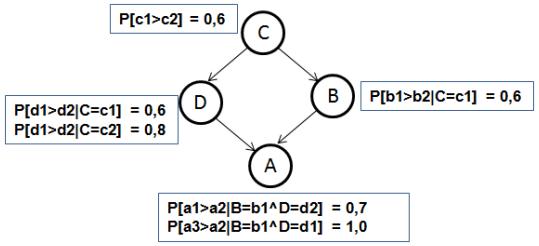
Uma RBP sobre um esquema relacional $R(A_1, ..., A_n)$ em um dado instante de tempo é um par (G, θ), onde: (1) G é um grafo direcionado acíclico, cujos nós são atributos em $\{A_1, ..., A_n\}$ e as arestas representam a dependência entre os atributos; (2) θ é um mapeamento que associa para cada nó de G um conjunto finito de regras de probabilidades condicionais. Fig. 1(c) ilustra uma $RBP\ PNet$ sobre R. Note que as preferências sobre os valores do atributo B dependem dos valores do contexto C: se $C = c1$, a probabilidade do valor $b1$ ser preferido ao valor $b2$ para o atributo B é 60%.

Id	A	B	C	D
u_1	a_1	b_1	c_1	d_1
u_2	a_1	b_1	c_1	d_2
u_3	a_2	b_1	c_1	d_2
u_4	a_1	b_2	c_1	d_2
u_5	a_2	b_1	c_2	d_1
u_6	a_3	b_1	c_1	d_1

(a)

(u_1, u_2, t_1)
(u_1, u_3, t_2)
(u_4, u_5, t_3)
(u_4, u_2, t_4)
(u_5, u_6, t_5)
(u_3, u_5, t_6)
(u_4, u_1, t_7)

(b)



(c)

Figura 1. (a) Uma instância I, (b) Um *stream* de preferências S, (c) Rede de Preferências PNet no instante t

Em nosso cenário, uma RBP é usada para comparar pares de tuplas. A qualidade de uma RBP como uma ferramenta de ordenação é mensurada por meio de sua precisão e *recall*.

Dessa forma, o problema de Mineração de Preferências Contextuais do Usuário em *Data Streams* é dado por:
Entrada: um esquema relacional R e um *stream* de preferências sobre R.
Saída: sempre que requisitado, retorne uma RBP sobre R no instante de tempo da solicitação, tendo boa precisão e *recall*.

3. Andamento do Trabalho

3.1. Estado Atual

No início deste trabalho foi identificado e formalizado o problema de mineração de preferências contextuais em *data streams*. Em seguida, alguns métodos para resolver esse problema foram propostos. Todos os métodos de mineração de preferências propostos trabalham apenas com um conjunto reduzido de estatísticas suficientes coletadas a partir do *stream* de preferências do usuário recebido como entrada. Estas estatísticas contêm o mínimo de informações necessárias para representar o *stream* de preferências, e proporcionam tanto uma maior velocidade na mineração quanto uma economia de memória.

Primeiramente, foram propostas duas estratégias para resolver este problema: a estratégia Gulosa e a Heurística. A principal ideia da estratégia Gulosa é criar uma relação de preferências a partir das mais promissoras dependências entre os atributos do *stream* de preferências. Já a estratégia Heurística é baseada em um Algoritmo Genético Incremental sobre um *stream* de preferências para executar a fase de aprendizado da estrutura de grafo da rede de preferências. Apenas a estratégia gulosa foi implementada, e testes preliminares sobre a mesma foram realizados, mostrando que ela é factível.

A estratégia Gulosa foi melhor trabalhada e deu origem ao algoritmo FPSMining. Mais detalhes sobre este algoritmo são apresentados nas próximas subseções.

Após isso, foi proposto o algoritmo IncFPSMining. Esse algoritmo constrói o modelo de preferências incrementalmente. A cada k bituplas temporais (parâmetro do algoritmo) que chegam no *stream* de preferências, o modelo corrente é atualizado. Essa atualização consiste em incrementar o modelo atual agregando novas arestas ao seu grafo. Utiliza-se a teoria do limite de Hoeffding para garantir que arestas inseridas anteriormente não percam sua validade ao longo do tempo. Foram realizados testes experimentais neste algoritmo, comprovando sua eficiência sobre dados reais.

3.1.1. O Algoritmo FPSMining

A principal ideia do FPSMining é criar uma relação de preferências a partir das mais promissoras dependências entre os atributos de um *stream* de preferências. Para medir a dependência de um par de atributos é usado o conceito de *grau de dependência*. O grau de dependência de um par de atributos (X, Y) com relação a um *snapshot* Q das estatísticas do *stream* de preferências S no instante t é um número real que estima como as preferências dos valores do atributo Y são influenciadas por valores do atributo X.

Dado isso, este algoritmo constrói uma RBP a partir das estatísticas como se segue: (1) Tire um snapshot Q das estatísticas de S no instante t; (2) Calcule o grau de dependência dd entre cada possível par de atributos de R de acordo com Q. Seja Ω o conjunto resultante desses cálculos, com elementos do tipo (A_i, A_j, dd), onde A_i e A_j são atributos de R; (3) Elimine de Ω todos os elementos cujo $dd < 0.5$ (indica fraca dependência); (4) Ordene os elementos (A_i, A_j, dd) em Ω de forma decrescente de acordo com o seu dd; (5) Considere que o grafo G da RBP começa com um nó para cada atributo de R. Dentro de um laço, pegue cada $(A_i, A_j, dd) \in \Omega$ e insira a aresta (A_i, A_j) no grafo G apenas se a inserção não formar ciclo em G. Assim, o grafo G da RBP é criado; (6) Uma vez que se tem G, estime as tabelas de probabilidade condicional θ da RBP usando o Princípio da Probabilidade Máxima (veja [de Amo et al. 2013] para detalhes) sobre Q.

3.1.2. Resultados Experimentais do FPSMining

Os dados reais usados nos testes do FPSMining contêm preferências relacionadas a filmes coletadas pelo *GroupLens Research* [1] a partir do *MovieLens web site* [2]. Foi simulado um *stream* de preferências a partir destes dados, como se segue: foi estipulado um intervalo de tempo λ (medido em dias e horas), e cada tupla no conjunto de dados foi comparada a todas as outras em um raio λ referente ao seu *timestamp*, gerando assim as bituplas temporais. O *stream* de preferências resultante possui cinco atributos (diretor, gênero, idioma, ator e ano), e seus elementos correspondem a preferências sobre filmes determinadas por um dado usuário. Como técnica de amostragem foi utilizado o *holdout* para *data stream*.

Fig. 2(a),(b) apresentam os resultados para a variação dos parâmetros *usuário* e λ, respectivamente. Os valores de *recall* e precisão apresentados são os valores médios de todas as medições efetuadas no *holdout*. Fig. 2(a) mostra que quanto mais filmes o usuário avaliou (tuplas), melhor é o *recall* e a precisão do FPSMining com relação a

[1] Disponível em `http://www.grouplens.org/taxonomy/term/14`
[2] Disponível em `http://movielens.umn.edu`

suas preferências. Fig. 2(b) mostra que até λ=1 dia os valores de qualidade aumentam conforme o número de elementos no *stream* de preferências aumenta. Na Fig. 2(a) foi utilizado λ=1 dia e na Fig. 2(b) foi utilizado *usuário*=$U1$.

Nos testes, os valores de *recall* e precisão foram satisfatórios, mostrando que o FPSMining é eficiente para minerar as preferências do usuário em *streams*. O tempo gasto na geração do modelo foi de 16ms e o tempo para completar um ciclo do *holdout* foi de 31ms, mostrando que o algoritmo atende as restrições de tempo exigidas em *streams*.

Usuário	Tuplas	Bituplas	*Recall*	Precisão
$U1$	7359	1580710	0.871	0.874
$U2$	6047	1658450	0.794	0.801
$U3$	4483	563419	0.778	0.784
$U4$	4449	198618	0.769	0.785

(a)

Stream	λ	Bituplas	*Recall*	Precisão
$S1$	1h	227100	0.800	0.814
$S2$	12h	645319	0.846	0.853
$S3$	1d	1048228	0.879	0.882
$S4$	7d	1580710	0.871	0.874
$S5$	15d	1759753	0.875	0.876

(b)

Figura 2. Conjunto de Testes Experimentais

3.2. Desenvolvimento Necessário para a Conclusão

De imediato, planeja-se elaborar uma comparação experimental entre os algoritmos FPSMining e IncFPSMining. Sobre as propostas de algoritmos, pretende-se ainda: 1) Implementar a estratégia Heurística, pois é esperado que ela apresente resultados bastante promissores, devido a sua capacidade potencial de poder atingir qualquer ponto do espaço de busca; 2) Implementar um algoritmo de *ensemble*, usando como algoritmo de aprendizado base os algoritmos propostos. Pretende-se também implementar um gerador de *stream* sintético de preferências (com introdução de *concept drift*), para possibilitar testes com uma quantidade de dados tão grande quanto se desejar. Finalmente, planeja-se avaliar o comportamento dos algoritmos propostos frente a *concept drifts*.

4. Trabalhos Relacionados

Com o intuito de prover uma análise comparativa, a Tabela 1 apresenta uma síntese sobre artigos importantes para a área de mineração de preferências. Segue o significado das siglas utilizadas na Tabela 1: LR/OR (*Label Ranking*/*Object Ranking*) – informa o problema de mineração de preferências; QT/QL (Quantitativo/Qualitativo) – informa se a elicitação de preferências é formalizada sobre um *framework* quantitativo ou qualitativo; U/A (Usuário/Automática) – informa se durante a elicitação de preferências foi solicitado ao usuário informações sobre suas preferências, ou se isso foi extraído de maneira automática; B/I (*Batch*/Incremental) – informa se o artigo aborda a mineração *batch* ou incremental.

Propostas de algoritmos adequados para resolver o problema de mineração de preferências do usuário em *streams* têm sido pouco explorados na literatura. Por exemplo, [Jembere et al. 2007] apresenta uma abordagem para minerar as preferências do usuário em um ambiente com vários serviços cientes de contexto, mas usa aprendizagem incremental somente para o contexto, e não para as preferências do usuário. Enquanto isso, [Somefun and La Poutré 2007] apresenta um método *online* que visa utilizar o conhecimento agregado sobre as preferências de muitos clientes para fazer recomendações a clientes individuais. Nenhum deles aborda especificamente o problema que o nosso trabalho aborda.

Tabela 1. Síntese de Artigos de Mineração de Preferências

Artigo	LR/ OR	QT/ QL	Modelo de Preferência	U/ A	Entrada	Saída	B/ I
[de Amo et al. 2013]	OR	QL	Contextual	U	BD de Preferências	RBP	B
[Jiang et al. 2008]	OR	QL	Pareto	U	$\{Superiores\}$, $\{Inferiores\}$	SPS	B
[Beretta et al. 2011]	OR	QL	Contextual	A	Log de Consultas	$\{\alpha - preferences\}$	I
[Jembere et al. 2007]	LR	QT	Contextual	A	Log de Dados	Sistema de Recomendação	I
[Somefun and La Poutré 2007]	LR	QL	Pareto	A	Ofertas sobre Pacotes	Mecanismo Aconselha.	I
Nossa Proposta	OR	QL	Contextual	A	*Stream* de Preferências	RBP_t	I

Referências

Beretta, D., Quintarelli, E., and Rabosio, E. (2011). Mining context-aware preferences on relational and sensor data. *International Workshop on Database and Expert Systems Applications*, pages 116–120.

Bifet, A. and Kirkby, R. (2009). Data stream mining: a practical approach. Technical report, The University of Waikato.

de Amo, S., Bueno, M. L. P., Alves, G., and da Silva, N. F. F. (2013). Mining user contextual preferences. *Journal of Information and Data Management - JIDM*, 4(1):37–46.

Domingos, P. and Hulten, G. (2000). Mining high-speed data streams. In *International Conference on Knowledge Discovery and Data Mining*, pages 71–80.

Fürnkranz, J. and Hüllermeier, E. (2011). Preference learning. Springer.

Gama, J. (2010). *Knowledge Discovery from Data Streams*. Chapman & Hall/CRC.

Jembere, E., Adigun, M. O., and Xulu, S. S. (2007). Mining context-based user preferences for m-services applications. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, pages 757–763.

Jiang, B., Pei, J., Lin, X., Cheung, D. W., and Han, J. (2008). Mining preferences from superior and inferior examples. In *KDD*, pages 390–398. ACM.

Rajaraman, A. and Ullman, J. D. (2011). *Mining of Massive Datasets*. Cambridge University Press.

Somefun, D. J. A. and La Poutré, J. A. (2007). A fast method for learning non-linear preferences online using anonymous negotiation data. In *Proceedings of the AAMAS workshop and TADA/AMEC conference*, pages 118–131.

Wilson, N. (2004). Extending cp-nets with stronger conditional preference statements. In *Proceedings of the 19th national conference on Artifical intelligence*, pages 735–741.

Recomendação de Consultas de Banco de Dados utilizando Agrupamentos de Usuários

Márcio de Carvalho Saraiva, Carlos Eduardo Santos Pires (Orientador), Leandro Balby Marinho (Orientador)

Pós-Graduação em Ciência da Computação
Departamento de Sistemas e Computação
Universidade Federal de Campina Grande (UFCG)
Campina Grande, PB, Brasil

marcio@copin.ufcg.edu.br, {cesp,lbmarinho}@dsc.ufcg.edu.br

Nível: Mestrado
Ano de ingresso no programa: 2012
Exame de qualificação: Dezembro de 2012
Época esperada de conclusão: Fevereiro de 2014

Abstract. *Database systems are becoming increasingly popular in the scientific community to support data exploration. In this scenario, users may not have the necessary knowledge about the domain of the database or how to formulate SQL queries to analyze the data. To solve this problem, we developed a study on the most recent techniques for query recommendation to improve them in such a manner that the user can receive better recommendations. In this proposal we discuss the main challenges of query recommendation systems and how clustering users can improve recommendations. Preliminary experimental results using real user query logs show that our study can generate effective query recommendations.*

Keywords: Query Recommendation, SQL, Databases, Query logs, Clustering.

1. Introdução

A humanidade produz milhares de exabytes (bilhões de gigabytes de dados) por ano (The Economist, 2010), o que faz com que sistemas de bancos de dados se tornem cada vez mais presentes na vida de usuários e desenvolvedores. Neste sentido, diversas empresas passaram a disponibilizar seus dados para estudos e pesquisas. Por exemplo, o site do projeto *UCSC Genome Browser* (Genome, 2013) fornece acesso a um banco de dados sobre genética, e o site do projeto *SkyServer* (*Skyserver*, 2013) disponibiliza um banco de dados sobre astronomia. O acesso a essas bases de dados ocorre por meio de aplicações que possuem uma interface web e permitem que vários usuários espalhados pelo mundo enviem consultas, normalmente escritas na linguagem SQL. Em geral, os bancos de dados apresentam esquemas complexos contendo uma grande quantidade de elementos (e.g. tabelas e colunas). Por serem ambientes abertos, os sites que disponibilizam seus bancos de dados na web possuem vários tipos de usuário. Entre esses tipos de usuário, encontramos aqueles que ainda não estão familiarizados com os elementos do esquema do banco de dados e podem não saber formular consultas que poderiam recuperar dados relevantes para suas necessidades.

Neste contexto, diversos trabalhos têm procurado identificar características em consultas de bancos de dados realizadas por usuários e a consequência da exploração dessas características para a recomendação de consultas. Julien Aligon, et al. (2013) e Marcel e Negre (2011) revisaram a literatura sobre técnicas de recomendação em banco de dados e observaram que faltam concepções mais refinadas dos usuários e de seus papéis com relação às consultas realizadas. Essa concepção auxiliaria na produção de técnicas e ferramentas que atendessem melhor aos anseios de perfis específicos de usuários, o que faz com que as atividades do usuário que possuam interação com o banco de dados sejam mais proveitosas, retornando dados que são importantes para suas consultas. Diferentemente, verificamos que não há conhecimento sobre técnicas para agrupamento de usuários do banco de dados em perfis que utilizam padrões de comportamento de usuários considerando características extraídas do histórico de consultas realizadas pelos mesmos. O agrupamento de usuários pode auxiliar na etapa de seleção das consultas que serão utilizadas nas técnicas de recomendação.

Neste trabalho, propomos uma técnica de mineração de padrões de acesso a banco de dados para recomendações de consultas. Essa técnica é baseada em agrupamentos de usuários construídos segundo características extraídas do histórico de consultas dos mesmos. A proposta visa auxiliar os usuários não familiarizados com o esquema de um banco de dados e que, por isso, têm dificuldade em formular consultas que poderiam recuperar dados relevantes.

2. Trabalhos relacionados

Foi realizado um levantamento do estado da arte, referente à área de recomendação de consultas de banco de dados. Com isso, observou-se que existem trabalhos que utilizam consultas realizadas por usuários em sistemas de banco de dados para auxiliar técnicas de recomendação de consultas SQL. Recentemente, Marcel e Negre (2011) afirmaram que há apenas as tentativas de Chatzopoulou et al. (2011) e Stefanidis et al., (2009) de formalizar recomendações de consultas de banco de dados para a exploração de esquema de banco de dados. Stefanidis et al. (2009) propõem uma matriz relacionando

usuários x consultas, com o intuito de medir a utilidade de uma consulta para um usuário. Esta utilidade é igual ao número de vezes que o usuário realizou a consulta. A técnica proposta por Chatzopoulou et al. (2011) de recomendação de consultas para exploração interativa de bancos de dados se enquadra nesta abordagem, embora as entradas da técnica sejam diferentes (sessões de acesso x tuplas recuperadas).

Khoussainova et al. (2010) e Akbarnejad et al. (2010) focam em fragmentos (atributos, tabelas, junções e predicados) de consultas realizadas para recomendações de novas consultas e consideram, portanto, as sessões de acesso dos usuários x matriz de fragmentos de consulta. A partir de um fragmento de consulta, o sistema procura outros fragmentos do histórico de consultas dos usuários que são similares entre si que compõem um conjunto Q'. O trabalho de Akbarnejad et al. (2010) realiza recomendações de consultas baseadas no histórico de consultas contendo fragmentos de Q'. A técnica proposta por Khoussainova et al. recomenda os fragmentos mais prováveis de Q' sabendo-se que o fragmento inicial é Q.

Buscando identificar características dos usuários de banco de dados, o trabalho de Lyes Limam et al. (2010) extrai o interesse (partes mais visualizadas do esquema e dos dados) dos usuários para expandir consultas, observando o histórico de consultas dos mesmos, sem a necessidade do preenchimento de formulários de preferências e interesses. O trabalho de Zhang e Nasraoui (2006) propõe uma técnica de mineração de dados composta por duas etapas: a primeira extrai o comportamento sequencial dos usuários (ou seja, a ordem de tabelas ou resultados que o usuário acessa) e a segunda analisa a similaridade entre consultas realizadas pelos usuários.

Há também trabalhos que expandem consultas SQL realizadas pelos usuários utilizando registros de preferências de cada usuário, como as pesquisas de Yang, Procopiuc e Srivastava (2009), e Koutrika e Ioannidis (2004), buscando adequá-las ao perfil dos usuários em sistemas de banco de dados; e trabalhos como o de Stefanidis et al. (2009) que acrescenta mais dados aos resultados retornados pelas consultas por meio da realização implícita de consultas semelhantes.

Todos esses trabalhos trouxeram contribuições significativas para a área, porém não foram encontrados em nossa revisão bibliográfica trabalhos que estudam padrões de comportamento de usuários considerando características extraídas do histórico de consultas realizadas pelos mesmos. Esse conhecimento pode auxiliar no agrupamento de usuários do banco de dados, o que facilita a escolha mais precisa de históricos de consultas de usuários para realizar recomendações e, assim, aprimorar métricas de avaliação da qualidade de retorno de técnicas de recomendação de consultas de banco de dados.

3. Estado atual do trabalho

Até o presente momento, foram selecionadas a base de dados que está sendo utilizada em nossos estudos e as técnicas de recomendação que servirão de *baseline* para a pesquisa. Além disso, foram feitos testes para avaliar a qualidade de retorno dessas técnicas, para isso foram empregadas métricas comumente utilizadas na literatura, *precision* (redefinida neste trabalho como a fração entre a quantidade de consultas retornadas como recomendação que são relevantes e a quantidade total de consultas que foram retornadas como recomendação) e *recall* (fração entre a quantidade de consultas

retornadas como recomendação que são relevantes e a quantidade total de consultas relevantes) de três técnicas selecionadas.

3.1. Base de dados selecionada

Utilizamos em nosso estudo uma base de dados disponibilizada pelo site *Skyserver*, constituída pelo histórico de consultas dos usuários do sistema durante o período de 03/2010 à 07/2011. Na base podemos observar o histórico de consultas de 447 usuários, com média aproximada de 387 consultas por usuário, constituindo 79.538 consultas executadas.

3.2. Técnicas selecionadas

Primeiramente agrupamos as técnicas estudadas de acordo com o tipo de entrada utilizado, apresentados na Tabela 1.

Tabela 1. Agrupamento das técnicas estudadas de acordo com o tipo de entrada utilizado

Usuários x Consultas	Usuários x Tuplas Recuperadas	Usuários x Fragmentos de consultas	Usuários x Esquema e dados	Usuários e registros de preferências do usuário
Stefanidis et al. (2009)	Chatzopoulou et al. (2011)	Akbarnejad et al. (2010)	Lyes Limam et al. (2010)	Yang, Procopiuc e Srivastava (2009)
		Khoussainova et al. (2010)	Zhang e Nasraoui (2006)	Koutrika e Ioannidis (2004)

Os trabalhos que utilizam usuários e registros de preferências do usuário necessitam de uma fase de obtenção de preferências do usuário, seja de forma direta com o preenchimento de formulários ou indireta com a mineração de padrões. No entanto, quando trabalhamos com um esquema de banco de dados grande[1], comum em sistemas nos quais o banco de dados está disponível na internet para qualquer usuário realizar consultas, essa fase de obtenção de preferências se torna inadequada. Por esse motivo, as técnicas pertencentes a este grupo não foram selecionadas para estudo.

As técnicas que utilizam usuários x esquemas e dados são aconselhadas para uso em ambientes em que os dados e o esquema do banco de dados pouco são alterados com o passar do tempo. Essas técnicas são sensíveis a alterações, pois levam em consideração os elementos do esquema e podem proporcionar recomendações diferentes para a mesma entrada se o esquema utilizado for alterado. Como a base de dados utilizada em nossos estudos é sujeita a alteração de dados recorrentes e o esquema de dados é grande, esse tipo de técnica de recomendação não é aconselhado.

Assim, foram selecionadas três tipos de técnicas de recomendação de consultas, usuários x consultas, usuários x tuplas recuperadas, usuários x matriz de fragmentos de consultas, que utilizam como parâmetros de entrada apenas o histórico de consultas dos usuários. Desses tipos de técnica de recomendação, foram selecionadas as técnicas

[1] Por exemplo, o esquema do banco de dados do projeto *SkyServer* contém 91 tabelas.

citadas nos trabalhos mais recentes para serem utilizadas em nossos estudos. São eles: Akbarnejad et. al. (2010), Stefanidis et. al. (2009) e Chatzopoulou et. al. (2011).

A base de dados utilizada em nossa pesquisa possui diversos tipos de retorno, como imagens e dados numéricos que podem sofrer alteração com o tempo, assim, desenvolvemos a técnica proposta por Chatzopoulou et. al. (2011) substituindo as tuplas que seriam utilizadas no algoritmo por consultas dos usuários. Assim, adaptamos em particular este trabalho para entrada de usuários x consulta, tornando-a mais adequada para nossos estudos utilizando a base de dados do *Skyserver* e mais semelhante ao tipo de entrada utilizada nas outras técnicas selecionadas para serem estudadas.

3.3 Análise das técnicas

Foi realizada uma análise das recomendações de consultas após a aplicação das técnicas selecionadas para 100 usuários selecionados aleatoriamente dentre os 477 usuários do histórico de consultas do banco de dados do site *SkyServer* em duas etapas. Na primeira etapa da análise, calculamos duas métricas de avaliação da qualidade dos retornos das técnicas de recomendação estudadas. Na segunda etapa, comparamos as técnicas entre si por meio de testes estatísticos para definirmos qual será o *baseline* para nossa pesquisa.

Na primeira etapa, o histórico de consultas de cada usuário foi dividido pela metade. A primeira parte foi utilizada como entrada para o algoritmo e a segunda parte utilizada como teste. Se as consultas retornadas como recomendações pelas técnicas estivessem presentes na parte de teste, essas seriam consideradas consultas relevantes para o cálculo das duas métricas estudadas. Ao fim desta etapa pudemos verificar que a média de *precision* e *recall* das três técnicas é baixa (~20%). A Figura 1 ilustra uma amostra dos experimentos executados para medir as métricas estudadas utilizando três históricos de usuários selecionados aleatoriamente.

	Técnicas/ Precision		
Usuarios	Chatzopoulou	Akbarnejad	Stefanidis
s0,s1,s2	1,00	0,07	0,18
s3,s4,s5	0,01	0,00	0,00
s6,s7,s8	0,00	0,00	0,00
s1,s2,s3	0,43	0,02	0,18
s2,s3,s4	0,60	0,90	0,87
s4,s5,s6	0,78	0,00	0,39
s5,s6,s7	1,00	0,71	0,36
s7,s8,s0	0,50	0,60	0,02
s8,s0,s1	0,25	0,15	0,34
s0,s1,s3	0,67	0,08	0,17
s1,s3,s5	0,43	0,02	0,17
s2,s4,s6	0,60	0,90	0,87
s3,s5,s7	0,01	0,00	0,00
s4,s6,s8	0,48	0,00	0,00
s5,s7,s9	0,17	0,71	0,00
s7,s8,s9	1,00	0,40	0,02
s9,s8,s6	0,40	0,10	0,38
s8,s7,s5	0,25	0,15	0,33
s7,s6,s4	0,50	0,60	0,08
s6,s5,s3	0,00	0,00	0,00
média	0,45	0,27	0,22

	Técnicas/ Recall		
Usuarios	Chatzopoulou	Akbarnejad	Stefanidis
s0,s1,s2	0,03	0,10	0,25
s3,s4,s5	0,00	0,00	0,00
s6,s7,s8	0,00	0,00	0,00
s1,s2,s3	0,10	0,06	0,58
s2,s3,s4	0,02	0,61	0,59
s4,s5,s6	0,46	0,00	0,46
s5,s6,s7	0,18	0,13	0,07
s7,s8,s0	0,01	0,02	0,02
s8,s0,s1	0,02	0,16	0,02
s0,s1,s3	0,06	0,11	0,24
s1,s3,s5	0,10	0,06	0,55
s2,s4,s6	0,02	0,61	0,59
s3,s5,s7	0,00	0,00	0,00
s4,s6,s8	0,12	0,00	0,00
s5,s7,s9	0,03	0,13	0,00
s7,s8,s9	0,02	0,02	0,02
s9,s8,s6	0,06	0,15	0,57
s8,s7,s5	0,02	0,16	0,35
s7,s6,s4	0,01	0,02	0,02
s6,s5,s3	0,00	0,00	0,00
média	0,06	0,12	0,22

Figura 1. Amostra de *precision* e *recall*, observando os retornos das técnicas selecionadas utilizando como entrada três usuários.

Estudamos a distribuição dos valores obtidos com aplicação das métricas em cada uma das técnicas e visualizamos que as distribuições não são do tipo normal.

Assim, por meio do teste de Kruskal-Wallis, verificamos que todas as técnicas são equivalentes analisando *precision* e *recall*. Por esse motivo, todas elas serão consideradas em nossos estudos como *baseline* da pesquisa. Verificamos também que, quanto maior o número de históricos de consultas de usuários que são utilizados como entrada para as técnicas de recomendação estudadas, menores serão as médias encontradas para as métricas observadas.

Assim, pretendemos estudar um modo de selecionar a quantidade ideal de históricos de usuários a ser utilizada em recomendações, pois assim teremos maiores valores para *precision* e *recall*. Assumimos heuristicamente que os históricos de usuários podem ser divididos em grupos, nossa hipótese neste trabalho é que estes grupos indicam quais os melhores históricos de consultas que devem ser considerados para recomendação de consultas para um usuário. Poderemos assim, também colaborar com a solução de problemas vistos nas demais técnicas apresentadas na Seção 3.2, como a necessidade de uma fase de obtenção de preferências do usuário, conhecimento do esquema do banco de dados e problemas com constantes mudanças nos dados.

4. Desenvolvimento necessário para conclusão

Estamos estudando a relevância dos grupos de usuários para a escolha das consultas que devem ser recomendadas para um usuário que está utilizando o sistema. Na Figura 2, apresentamos como será o funcionamento da técnica proposta em nossa pesquisa. Na etapa 1, diversos usuários acessam o banco de dados que armazena o histórico de consultas realizadas por cada usuário. Na etapa 2, após a extração dos atributos presentes no histórico de consultas dos usuários, é realizado o agrupamento dos usuários utilizando os atributos extraídos e uma técnica de agrupamento. Por fim, são utilizados os atributos extraídos do histórico de consultas e dos grupos de usuários para realizar recomendações de novas consultas na etapa 3.

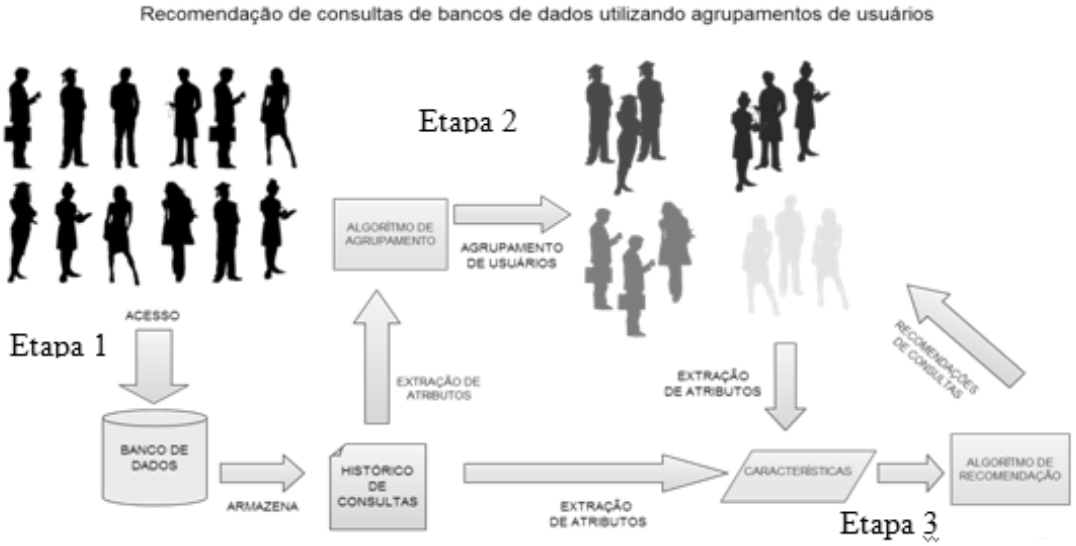


Figura 2. Visão geral da técnica proposta.

Iremos comparar o impacto da criação de grupos nos valores das métricas *precision* e *recall* de técnicas já existentes. Além disso, verificaremos se uma nova técnica de recomendação de consultas que será desenvolvida em nossa pesquisa

apresenta melhoras na qualidade das recomendações de consultas realizadas em banco de dados.

5. Avaliação dos resultados

Assim como descrito na Seção 3.3, repetiremos os experimentos buscando avaliar as métricas observadas após adicionar a etapa de agrupamento de usuários nas técnicas selecionadas e na proposta em nossa pesquisa. Em seguida, serão realizados testes estatísticos buscando demonstrar uma nova técnica para recomendar consultas em banco de dados que retorne recomendações com melhor *recall* e *precision* do que as demais técnicas encontradas no estado da arte.

6. Referências

Koutrika, G. and Ioannidis, Y. (2004). Personalization of Queries in Database systems. In Proceedings of 20th Intl. Conf. On Data Engineering (ICDE). Boston, MA, USA. p. 597-608.

Akbarnejad, J., Chatzopoulou, G., Eirinaki, M., Koshy, S., Mittal, S., On, D., Polyzotis, N. and Varman, J. S. V. (2010). SQL QueRIE recommendations. In Proceedings of the VLDB Endowment. v.3 n.1-2.

Limam, L., Coquil, D., Kosch, H. and Brunie, L. (2010). Extracting user interests from search query logs: A clustering approach. In Proceedings of the 2010 Workshops on Database and Expert Systems Applications (DEXA '10). p. 5-9.

Aligon, J., Golfarelli, M., Marcel, P., Rizzi, S. and Turricchia, E. (2013) Similarity measures for olap sessions. To appear in International Journal of Knowledge and Information Systems (KAIS).

Marcel, P. and Negre, E. (2011). A survey of query recommendation techniques for dataware-house exploration. In Proceedings of the 7th Conference on Data Warehousing and On-Line Analysis (Entrepts de Donnes et Analyse) (EDA'11), p. 119-134.

Khoussainova, N., Kwon, Y., Balazinska, M., Suciu, D. (2010). SnipSuggest: context-aware autocompletion for SQL. In Proceedings of the VLDB Endowment. v.4 n.1, p. 22-33.

Revista The Economist, The Data Deluge, edição do dia 25 de Fevereiro de 2010.

Site do Projeto Genome disponível em http://genome.ucsc.edu/ - acesso em 31/05/2013

Site do Projeto Skyserver disponível em http://cas.sdss.org/ - acesso em 31/05/2013

Stefanidis, K., Drosou, M., Pitoura, E. (2009). You May Also Like results in relational databases. In PersDB, p. 37-42

Yang, X., Procopiuc, C. M. and Srivastava, D. (2009). Recommending join queries via query log analysis. In 25th International Conference on Data Engineering (ICDE 2009), p. 964–975.

Zhang, Z., Nasraoui, O. (2006). Mining search engine query logs for query recommendation, Proceedings of the 15th international conference on World Wide Web, Edinburgh, Scotland, p. 1039-1040.

APPWM - Agrupamento Personalizado de Pontos em Web Maps usando um modelo multi-dimensional

Marcio Bigolin[1], Helena Grazziotin Ribeiro[2], Renata Galante[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil
[2]Centro de Computação e Tecnologia da Informação Universidade de Caxias do Sul (UCS) – Caxias do Sul – RS – Brazil

{mbigolin, galante}@inf.ufrgs.br, hgrib@ucs.br

Nível: mestrado

Programa: Programa de Pós-graduação em Computação - PPGC/UFRGS

Ano de ingresso: março/2012

Época esperada para conclusão: março/2014

Etapas já concluídas: proposta de mestrado (OUT/2012), créditos (DEZ/2012), seminário de andamento (MAI/2013), submissão de artigo (JUN2013).

Etapas futuras: submissão de artigos (NOV/2013), defesa da dissertação (MAR/2014).

Resumo. *Com a evolução dos Sistemas de Informação Geográficos (SIG) e de mapas na web, a quantidade de informação e a sua complexidade cresceram de forma descontrolada. Essa grande quantia de dados em um mapa pode tornar as interpretações confusas. O objetivo deste trabalho é, através de um modelo multi-dimensional, organizar e estruturar os dados para a criação de agrupamentos , visando uma melhor interpretação. O modelo proposto utiliza dimensões clássicas que podem ser replicadas a diversos problemas de espacialização de dados. Com essa estrutura, pretende-se que o usuário, sem muito conhecimento em processamento de imagem, possa gerar mapas e consultas interativas, com um maior poder de decisão, sem a necessidade de um grande processamento computacional.*

Palavras-chave: Modelagem Multi-dimensional, Agrupamento de dados, Sistemas de Informação Geográfico

Abstract: *The evolution of Geographic Information Systems (GIS) and maps on the web, the amount of information and its complexity increased uncontrollably. This large amount of data on a map can make confusing interpretations. The objective of this work is, through a multi-dimensional model, organize and structure the data for clustering, seeking a better interpretation. The proposed model uses classical dimensions that can be replicated to several problems of spatial data. With this structure, it is intended that the user without much knowledge in image processing, can generate maps and interactive queries with greater decision-making power, without requiring a large computational processing.*

Keywords: *Modeling Multi-Dimensional, Clustering Data, Geographic Information Systems*

1. Introdução

Uma das características dos Sistemas de Informação Geográficos (SIG) é a visualização de planos de informação, ou seja, a sobreposição de camadas de informações distintas, podendo assim, em tempo real, produzir mapas diferentes. Um exemplo de plano de informação muito comum é o de pontos. Com ele, é possível identificar a localização de eventos ou outros objetos geográficos (pontos telefônicos, casas, pontos comerciais, hidrantes, entre outros). Normalmente, o objetivo deste plano de informação é mostrar a distribuição dos dados no mapa [Tyner 2010]. Em muitos casos o número/quantidade de pontos é relevante para a compreensão da informação, bem como a análise da frequência dos objetos e/ou eventos.

Um dos grandes desafios que a comunidade de Informação Geográfica enfrenta é fornecer aos tomadores de decisão ferramentas avançadas, que sejam capazes (semântica e visualmente) de integrar aspectos quantitativos, qualitativos e cognitivos de um domínio de interesse. Quando uma grande quantidade de dados está disponível, a síntese de informação, bem como um resultado derivado pode resultar em uma atividade demorada e de custo elevado [De Chiara et al. 2011]. A análise Geovisual trabalha com problemas que envolvem o espaço geográfico e vários objetos, eventos, fenômenos e processos. As técnicas visuais de análise são essenciais para lidar com os conjuntos de dados atuais (que ampliam-se rapidamente em tamanho e complexidade). As abordagens que trabalham em um ambiente puramente analítico ou em um nível puramente visual não obtém sucesso, devido à dinâmica e complexidade dos processos subjacentes [Andrienko et al. 2011].

1.1. Problema de pesquisa

O grande volume de dados que ao ser plotado em um mapa pode gerar visualizações confusas e de difícil interpretação. Esse problema somado as diversas possibilidades de navegação que um mapa em ambiente *Web* proporcionam amplificam o problema. A Figura 1 ilustra o problema no qual pode-se ver o mapa com pontos de produtores de suínos dos municípios de abrangência do COREDE-SERRA no Rio Grande do Sul [FEE 2011].

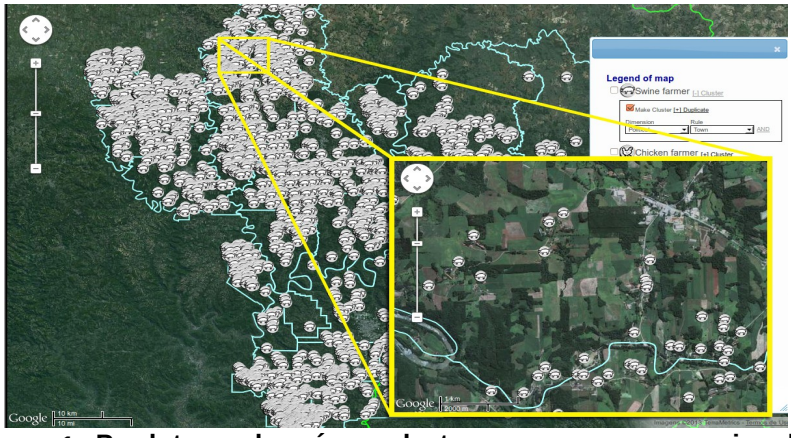


Figura 1. Produtores de suínose destaque com zoom aproximado

A parte em destaque na Figura 1 mostra um *zoom* de determinada área, onde apresenta-se uma grande quantidade de pontos. Isto demonstra que, dependendo do nível de detalhamento ou da região de abrangência analisada, torna-se quase impossível

visualizar e acessar a dispersão dos pontos, sendo, neste caso, necessário utilizar técnicas de agrupamento ou de frequência para melhorar o entendimento. Para obter uma navegação mais eficiente, com possibilidades de tomada de decisão em diferentes níveis de *zoom* e/ou atomicidade da informação, de forma a reduzir a carga cognitiva e proporcionar maior agilidade na interpretação de informações, está sendo proposto o desenvolvimento de um método suportado por uma ferramenta para atender essas necessidades.

1.2. Principais contribuições do trabalho

O trabalho visa propor um método de agrupamento que utiliza um modelo multi-dimensional, semântico, genérico e bem definido para a escolha dos critérios dos grupos de dados para melhorar a visualização de grande densidade de informação em um mapa na *web*. O modelo multi-dimensional facilitará ao usuário a escolha dos grupos e refinamento dos mesmos. Cabe ressaltar que o modelo estará implícito para o usuário, isto é, ele não precisará ter conhecimento em modelagem multi-dimensional. Paralelamente, o modelo também permitirá o estudo para melhorar o desempenho da aplicação. Experimentos preliminares mostram que a transferência de dados e o desenho de informação geográfica em mapas *web* foi mais rápido.

O modelo proposto poderá ser considerado genérico se mostrar-se capaz de ter aplicabilidade em outros domínios e aplicações. O modelo deve ser bem definido de modo que, com a aplicação em áreas distintas, não sejam feitas alterações substanciais. Experimentos envolverão aplicações reais da área de zoologia, engenharia civil e agricultura. Por fim, o modelo deve ser semântico por dar sentido aos agrupamentos gerados.

As agregações e a desagregação de múltiplos planos de informação em um mesmo *web* mapa, necessita de outras variáveis visuais possibilitando-a apresentar diversos tipos de informação ao mesmo tempo no mesmo mapa. A ferramenta visa suportar isso com o uso do modelo multi-dimensional, assim como com a geração de artifícios visuais para os ícones facilitando a interpretação.

2. Principais trabalhos relacionados

Existem diversas técnicas para o agrupamento de informações em mapas: símbolos ou círculos graduados, na qual os valores quantitativos são agrupados em classes, dentro de uma classe, todos os recursos são desenhados com o mesmo símbolo [Auer et al. 2011; MacEachren et al. 2011]; mapas coropléticos, os quais consistem em uma forma de representar grande quantidade de dados usando a abstração de cores [Goovaerts 2012]; *Dot Density Map* é uma solução simples, que utiliza o seguinte conceito: um (1) ponto representa um conjunto de informações [Young and Jensen 2012]. Em *web* mapas é comum a utilização de bibliotecas de grupo (*cluster)*. Estas bibliotecas a partir de um considerado número de pontos em um quadrante no mapa geram um grupo. Esses grupos são gerados novamente conforme o usuário navega pelo mapa de forma interativa.

Para análise de informações geográficas usando técnicas OLAP pode se citar o *Spatial* OLAP (SOLAP) Bedard et al. (2007). SOLAP explora os dados através de uma navegação multi-dimensional em qualquer forma de visualização de dados, métricas calculadas e filtragem pelos atributos das dimensões possuindo estrutura dos dados que suporte as diferentes formas geométricas e múltiplas representações para diferentes

escalas. Silva, R. e Santos (2011) propõe o agrupamento de dados para densidade espacial utilizando o algoritmo *DBScan*. Gerando uma nova representação para cada agrupamento, diminuindo o número de objetos espaciais que precisam ser colocados no mapa. Se houver visualização de dados não espaciais, os mesmos também são agregados mantendo a sincronização entre o mapa e uma visualização tabular [Bedard et al. 2007; Silva and Santos 2011]. A ferramenta proposta utiliza um modelo multi-dimensional voltado para consulta e genérico portanto independente de algoritmos para criação de grupos.

Os requisitos elencados, tendo como base a necessidade verificada no protótipo (Figura 1), assim como de trabalhos como o de [Auer et al. 2011] para gerar um mapa na *web* com uma grande densidade de informação são: mostrar densidade aparente da informação (em uma primeira visão obter onde possui mais ou menos informação); densidade exata da informação (apresentar, quantos eventos estão agrupados); permitir visualizar granularidades diferentes (ver grupos conforme a escala do mapa); e manter a mesma abstração em escala diferentes (para diminuir a carga cognitiva, ou seja, manter a mesma forma de analisar o mapa independentemente do *zoom*, por exemplo, um círculo graduado que em um determinado nível de *zoom* represente uma quantidade e em outro nível um círculo de mesmo diâmetro represente quantidade diferente). Esses quatro requisitos elencados foram tabulados comparativamente na Tabela 1.

Tabela 1. Tabela resumo dos recursos disponíveis em cada técnica.

Requisitos	Círculos graduados	Mapas coropléticos	Dot Density Maps	Agrupamento
Mostrar densidade aparente de informação	Sim	Sim	Sim	Sim
Densidade exata da informação	Não	Não	Sim	Sim
Permitir visualizar atomicidades diferentes	Sim/Médio	Sim/Difícil	Sim/Fácil	Sim/Fácil
Manter mesma abstração em nível de zoom diferente	Não	Não	Pode	Sim

3. Andamento da proposta

Esta seção descreve a proposta de mestrado que é gerenciar o agrupamento de pontos em mapas web com a utilização de um modelo multi-dimensional. Uma visão geral da aplicação é ilustrada na Figura 2.

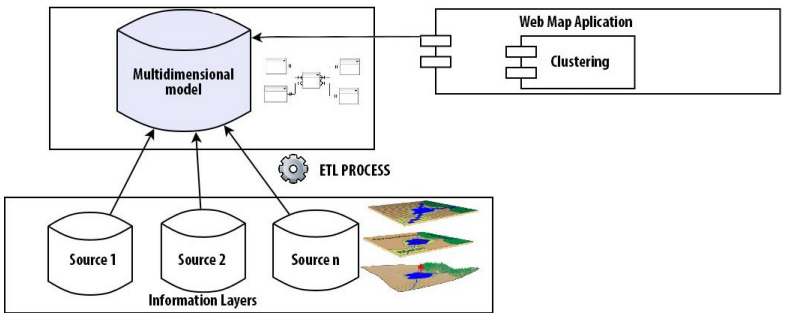


Figura 2. Visão geral do sistema

3.1. Modelo Multi-Dimensional

O modelo proposto trabalha com 3 dimensões bases a fim de otimizar a geração dos grupos: a dimensão “dim_politica” (auxilia na organização dos grupos por divisões politicas, como município, estado e países), a dimensão “dim_fisica” (auxilia na organização por meio de características físicas comuns, como bacia hidrográfica, região hidrográfica, formação vegetal entre outras) e por ultimo a dimensão “dim_tempo”

(essa dimensão armazena características temporais referente ao ponto, como estado ou validade da informação). A Figura 3 mostra em mais detalhes a relação destas dimensões com o fato Ponto. A dimensão física está subdividida em duas outras dimensões a fim de permitir a implementação das características de forma a manter o modelo em formato estrela [Kimball and Ross 2002] e simplificar as consultas. Outras dimensões físicas como tipo de solo e características climáticas podem ser adicionadas, conforme for verificado a necessidade.

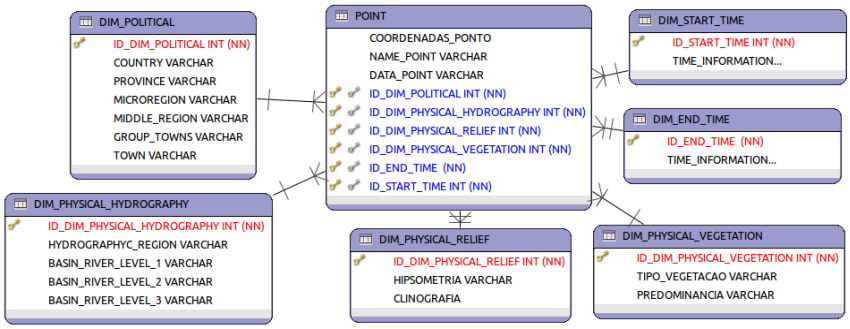


Figura 3 . Modelo Multi-dimensional

3.2. Ferramenta de mapa web

O desenvolvimento da ferramenta que servirá para validar o trabalho utiliza a API do GoogleMaps v 3.0. Os dados são armazenados em um SGBD PostgreSQL com o módulo Postgis, que fornece diversas funcionalidades geográficas além de possibilitar uma arquitetura integrada dos dados geográficos com os dados alfanuméricos do banco de dados [Ferreira et al. 2005].

Para a geração dos grupos, os dados geográficos associados são de fundamental importância para o processo de ETL(*Extraction Transform and Load).* Planos de informação clássicos são utilizados onde são feitas as consultas para verificar se o ponto pertence ou não a uma determinada região geográfica.

A Figura 4 (A) apresenta a ferramenta que permite realizar as consultas tendo como base o modelo multi-dimensional para construir os agrupamentos. As funcionalidades propostas e implementadas são: criar o agrupamento (permitir habilitar ou não o agrupamento sobre os planos de informação); duplicar o plano de informação (permitir que seja visualizado o mesmo plano de informação por agrupamentos diferentes); e criar restrições do tipo AND (criar grupos por mais de uma dimensão).

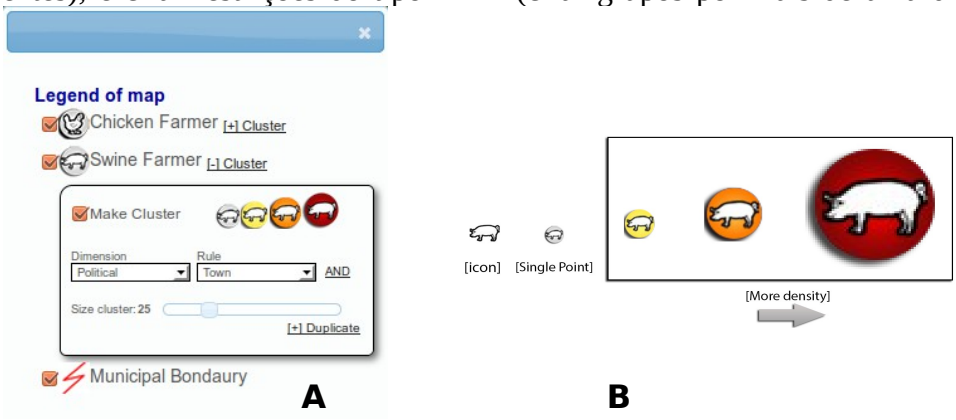


Figura 4. Ferramenta para a criação dos grupos

A solução para o plano de informação da Figura 1, é apresentado na Figura 5 (A) usando apenas a técnica de agrupamento. Neste caso, os grupos levaram apenas em consideração a distância entre os pontos já na Figura 5 (B) visualiza-se o grupo agregado por município.

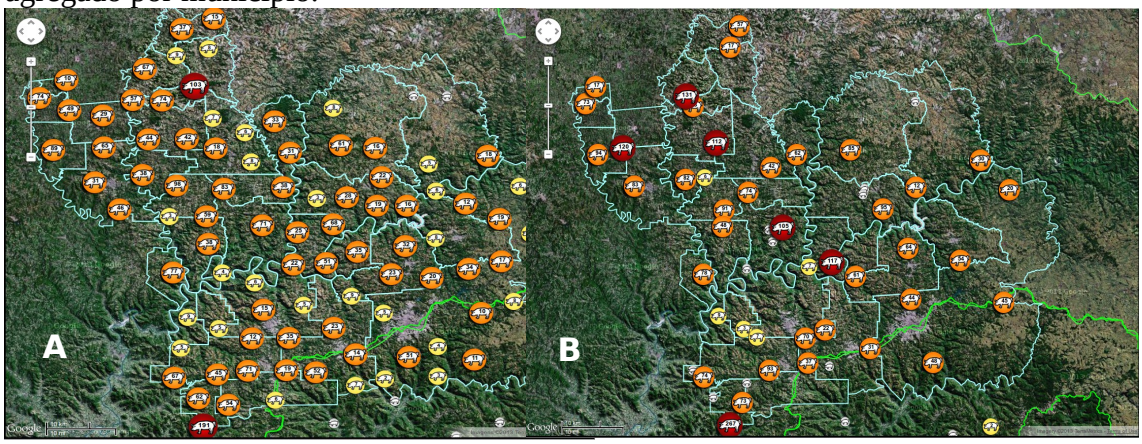


Figura 5. Mosaico de funcionalidades

4. Próximas Atividades

As próximas atividades incluem desenvolver uma amostragem com dois grupos distintos, para comprovar que o modelo utilizado é genérico. Um dos grupos será o GRID (Grupo de Gestão de Riscos e Desastres) da Universidade Federal do Rio Grande do Sul (UFRGS) e outro do ISAM (Instituto de Saneamento Ambiental) da UCS (Universidade de Caxias do Sul). Serão desenvolvidas pequenas aplicações com os dados provenientes destes grupos. Para o ISAM serão utilizados os dados de produtores rurais de região de abrangência do COREDE-SERRA. Para o GRID será utilizado os dados de desastres que ocorreram no estado do Rio Grande do Sul.

Para os dois estudos de caso, os experimentos incluem a análise de usabilidade com usuários reais das aplicações. Além disso, serão recolhidos depoimentos com vistas a analisar de forma qualitativa e quantitativa a real aplicação da proposta de utilização de uma ferramenta de web-mapa com a utilização de um modelo multi-dimensional como base para a tomada de decisão e geração de *cluster* em um mapa.

5. Considerações Finais

Este artigo apresentou um modelo de gerar grupos de pontos de forma automatizada e por dimensões. Por se tratar de informação geográfica, qualquer informação pontual, ou seja, qualquer domínio que utilizar um ponto georreferenciado poderá usar este método. O modelo multi-dimensional proposto apresenta-se como um modelo genérico para melhorar a visualização de um grande número de informação, independente da escala do mapa. É um modelo mais semântico, pois o mesmo visa dar ao usuário do mapa uma melhor visualização da informação controlada pelo contexto do usuário e suas escolhas, ou seja, permite visualização de dados mais direcionada à necessidade de exibição de cada aplicação e/ou especialista. Cabe destacar a necessidade da aplicações extras conforme o proposto visto a validar a aplicabilidade em outras áreas desse mesmo trabalho.

Referencias

Andrienko, G., Andrienko, N., Keim, D., MacEachren, A. M. and Wrobel, S. (aug

2011). Challenging problems of geospatial visual analytics. *Journal of Visual Languages & Computing*, v. 22, n. 4, p. 251–256.

Auer, T., MacEachren, A. M., McCabe, C., Pezanowski, S. and Stryker, M. (mar 2011). HerbariaViz: A web-based client–server interface for mapping and exploring flora observation data. *Ecological Informatics*, v. 6, n. 2, p. 93–110.

Bedard, Y., Rivest, S. and Proulx, M.-J. (2007). Spatial On-Line Analytical Processing (SOLAP): Concepts, Architectures, and Solutions from a Geomatics Engineering Perspective. *Data Warehouses and OLAP: Concepts, Architectures and Solutions*. p. 22.

De Chiara, D., Del Fatto, V., Laurini, R., Sebillo, M. and Vitiello, G. (jun 2011). A chorem-based approach for visually analyzing spatial data. *Journal of Visual Languages & Computing*, v. 22, n. 3, p. 173–193.

FEE (2011). Corede Serra. http://www.fee.tche.br/sitefee/pt/content/resumo/coredes_detalhe.php?corede=Serra, [accessed on Jun 22].

Ferreira, K. R., Casanova, M. A., Queiroz, G. R. De and Oliveira, O. F. De (2005). Arquiteturas e linguagens. *Bancos de Dados Geográficos*. Curitiba: MundoGEO. p. 33.

Goovaerts, P. (2012). Geostatistical analysis of health data with different levels of spatial aggregation. *Spatial and Spatio-temporal Epidemiology*,

Kimball, R. and Ross, M. (2002). *The data warehouse toolkit: guia completo para modelagem dimensional*. Rio de Janeiro: Campus. p. 494

MacEachren, A. M., Jaiswal, A., Robinson, A. C., et al. (oct 2011). SensePlace2: GeoTwitter analytics support for situational awareness. In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*. . IEEE. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6102456, [accessed on Apr 18].

Rivest, S. and Marchand, P. (2001). Toward better support for spatial decision making: defining the characteristics of spatial on-line analytical processing (SOLAP). *GEOMATICA*, v. 55, n. 4, p. 539 to 555.

Silva, R. and Santos, M. Y. (2011). Spatial Clustering to Uncluttering Map Visualization in SOLAP. In *ICCSA'11 Proceedings of the 2011 international conference on Computational science and its applications*.

Tyner, J. A. (2010). *Principles of Map Design*. 1. ed. New York, New York, USA: The Guilford Press. p. 259

Young, S. G. and Jensen, R. R. (may 2012). Statistical and visual analysis of human West Nile virus infection in the United States, 1999–2008. *Applied Geography*, v. 34, p. 425–431.

HSTB-index: A Hierarchical Spatio-Temporal Bitmap Indexing Technique

Cesar Joaquim Neto[1, 2], Ricardo Rodrigues Ciferri[1], Marilde Terezinha Prado Santos[1]

[1] Department of Computer Science – Federal University of São Carlos (UFSCar)
Rodovia Washington Luís, km 235 - SP-310 – 13565-905 – São Carlos – SP – Brazil

{cjneto, ricardo, marilde}@dc.ufscar.br

[2]International Business Machines (IBM)
Rodovia Jornalista Francisco Aguirre Proença, km 09 (SP-101) Chácaras Assay – 13186-900 – Hortolândia – SP – Brazil

cesarjn@br.ibm.com

Nível: Mestrado

Ingresso no Programa: Março de 2013

Exame de Qualificação: Março de 2014

Data Esperada de Conclusão: Março de 2015

Abstract. *Through years, the database group at the Federal University of São Carlos has been researching new ways of optimizing queries in data warehouses containing non-conventional data. The need for such research comes from a number of partners dealing with spatial and temporal data where a simple query can demand huge levels of computational efforts. Advances were carried out by creating new index structures that deal with join bitmap indexes, minimum boundary rectangles and temporal data (in separate or meshed together) and some variant proposals. This work intends to explore and extend the STB-index (Spatio-Temporal Bitmap Index) by creating a new hierarchical index of the temporal data called HSTB-index (Hierarchical Spatio-Temporal Bitmap Index). The attempt is to better prune the number of false candidates in temporal predicates that might arise before the final result can be calculated and fetched.*

Keywords: data warehouse**,** spatio-temporal databases, spatio-temporal data warehouses, bitmap index

1. Introduction

A data warehouse (DW) is a specialized multidimensional database adjusted for querying data among large integrated, subject-oriented, and historical datasets. Its structure, usually a star or a snowflake schema, allows data to be seen in distinct points of view as well as in different levels of granularity. In such model, the fact and dimension tables (through the defined dimension hierarchies) allow users to perform many OLAP (on-line analytical processing) operations such as 'slice and dice', pivot, drill-across, drill-down, and drill-up.

For years, the concept that dimension data could not be changed was defended by researchers until they realized such data was not as immutable as they thought. The concept of slow changing dimensions has been carved by Kimbal (2002) and soon researchers started to think changes could also be needed to the fact table data. In this line of thought, the temporal data warehouses concepts started to be shaped due to the need of capturing the changes a data warehouse can suffer through time being those related to data instantiation or their own structure. According to Jensen et al. (1994), there are two main mechanisms a temporal data warehouse can use to capture data instantiation changes: the validity time (also called "real world time" – the time a given information is valid for the business) and transaction time (also called "database system time" – the time a given information is valid for the database). Regarding structural changes, Roddick (1995) classifies them as schema modification, schema evolution, or schema versioning. Querying such databases has also become a challenge as structure additions that support evolving data and schema needed to be considered. Queries could fail by fetching invalid data or spanning different schema versions without proper processing. Solutions for such problem came in the form of additions to the SQL query syntax or pre-processing of queries (using the captured schema metadata) so schema versions and data evolution could be respected.

Spatial data warehouses (SDWs) had their own line of research with the inclusion of at least one spatial data type (like points, lines, circles and polygons) in one or more dimension tables or as a spatial measure in the fact table. Queries over SDWs have a spatial predicate, such as, "retrieve the profit per month per store for stores that intersect a given query window". The above mentioned inclusion brought immediate difficulties as the large volumes of data found in DWs together with the inherent complexity of dealing with spatial data made processing times to grow a lot and made clear the need for new index and query processing techniques. New indexing techniques and arrangements like the SB-index and its evolution, the HSB-index, were developed so they could be made available and used in future works (Siqueira et al. 2012).

As time evolved, researches found that for some applications there was a need for capturing changes to spatial data as well. As temporal data warehouses already had what was needed to comply with keeping data and structural history, it seemed natural to combine both temporal and spatial data warehouse concepts thus arising a new data warehouse type called spatio-temporal data warehouse (STDW). With such new data warehouse type, difficulties while querying data were also combined and a new line of research was found trying to speedup queries using both concepts and this is where this work fits in.

2. Related Work

Bitmap join indexes (BJI) are an evolution of the common bitmap indexes and they have found their application in data warehouses due to their ordered organization and ease to process. It can map rows from a fact table directly to their relative dimension rows without the need of the inner join operation. Queries using BJIs can be processed using bit operations which are resolved by processors in a single clock cycle.

Although, due to DWs' own property of having huge volumes of data, even bitmap indexes have problems to scale and such difficulty could be addressed by three different techniques to be presented below: Binning, compression and codification.

The binning technique proposed by Wu, Stockinger & Shoshani (2008) consists of grouping various keys in common bitmap structures called "bins" and, therefore, demanding less bitmaps to code a certain column. As example we can have a numeric column with values ranging from 0 to 100. Instead of having 100 bit vectors representing its joint to the fact table, we can create 10 vectors (or bins) containing the mapping bits for values in the intervals (0, 10], (10, 20], and so on. This arrangement makes it possible to exclude many bins in advance thus saving processing time. The drawback is that false candidates can be introduced requiring further refinement of the results but the gains obtained by excluding bins overpass the refinement step loss.

Wu, Otoo & Arie (2006) details the WAH (Word Aligned Hybrid code) technique which is a specialized compression algorithm. Due to the own DW nature of having huge number of rows to be mapped between fact and dimension tables, the resultant bit vectors are often sparse meaning many sequences of consecutive '0's and '1's can be found. The WAH algorithm makes use of such property by separating the bit vector in equal pieces (31 or 63 bit in length – according to the processor's architecture) and making two types of "runs" called 'tail' and 'filling'. Tails have "0" as starting bit indicating all remaining bits need to be respected as they are. Fillings have "1" as starting bit, with the second bit indicating the bit to be repeated, and the remaining bits indicating the number of piece repetitions being compressed. In an example found in Stockinger & Wu (2007), a bit vector containing 5456 bits (and separated in 31 bit pieces) could be represented using 96 bits (one tail piece followed by one filling piece indicating 174*31 "0" bits followed by another tail piece).

The coding technique presented by Wu & Buchmann (1998) has as central point the use of binary codes representing all the domain values found in the attribute to be indexed. Such coding is then used as base for the mapping vector resulting in a reduction on the number of vectors needed for the coding itself. As example we can have a column with 5 character values ("a" to "e") which would require 5 bit vectors to be mapped against the fact table. The same values could be mapped using 3 bits (000 for "a", 001 for "b", and so on) and the same mapping could be done using 3 codified bitmap vectors instead of 5. Extrapolating the idea we could have a 1 million domain values attribute coded by 20 vectors (instead of the original 1 million). The drawback is that the result needs to be de-codified before being returned but such cost would be acceptable taking in account the savings in number of mapping vectors and the calculations that can be done using them.

With all these bitmap functionalities at hand (gathered by their authors in a software called Fastbit) and aiming to improve query performance in SDWs, Siqueira (2009) introduced a new index structure called Spatial Bitmap Index (SB-index). The index is represented by a vector where its index match the keys for a specific spatial column of a dimension table. The vector contents are the minimum boundary rectangles (MBRs) of the spatial column value and a pointer linking to the corresponding BJI. The idea is to have a first step scanning through the MBRs contained in the index structure and filtering out the rows outside the query window. As MBRs are used for this first scan, results have to be further evaluated as false candidates might have been introduced. This index' aim is to trade from many "costly" polygon intersection operations to many "cheaper" square intersection operations plus a few polygon intersection operations (during false candidates filtering).

Although gains were soundly verified by the use of SB-index, advances on it were thought by the same author and the results were outlined in Siqueira et al. (2012). His idea was to improve the index scan by exchanging the vector based structure to a hierarchical structure (for instance, an R*-tree). Such tree-like organization made possible to further group MBRs into broader groups and prune tree branches not pertaining to a particular MBR group. A main memory buffering technique was also used so to have data loaded for processing as long as possible. Such approach was called Hierarchical Spatial Bitmap index (or HSB-index).

As further development of the SB-index, Tsuruda (2013) has broadened its concept to contemplate STDWs. The outcome of her work was a new index called Spatio-Temporal Bitmap index (STB-index) which idea is the inclusion of initial and final validity times to the index' MBRs. Such temporal information allow temporal clauses to be evaluated first and, therefore, based on the time related to MBRs, the STB-index can discard invalid MBRs in advance.

3. Our Proposal

3.1. Description

In the same way the SB-index has been evolved into a hierarchized structure (the HSB-index), this work's proposal is to evolve the STB-index into a hierarchized structure for the valid times and, thus, decrease the number of candidates to be explored by pruning candidates in early steps. It seemed natural the new index to be called Hierarchical Spatio-Temporal Bitmap index, or HSTB-index. Table 1 below summarizes the four indexes and their main differences.

Table 1. Index features comparison

Index	Collection organization	Run method	Type of DW
SB-index	Vector	Analyze all MBRs	SDW
HSB-index	Tree	Prune MBR records	SDW
STB-index	Vector	Analyze all MBRs	STDW
HSTB-index	Tree / Other	Prune MBR records	STDW

Although the main idea is already set, its execution is generating majority of discussions. The first big question is regarding the most recommended structure to hold the MBRs. So far a B+-tree had been considered but it was also thought that a bitmap index using the binning technique might lead to good results too. Another pending question regards the granularity of the temporal data for the index organization where the "YYYYMM" format seems to be the most natural so far. A third point to be considered is which spatial index to use so the best results can be found (SB-index or HSB-index) regarding the selectivity of the queries. The last, yet very important, question is regarding the execution order of the index meaning that the decision whether to limit candidates using the temporal attributes earlier than the spatial attributes or vice-versa is still to be investigated. Such questions and discussions have led to some variants described below and which should be evaluated throughout the work.

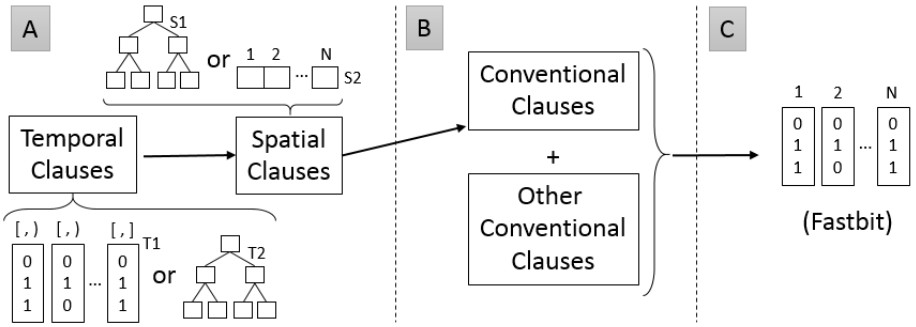


Figure 1. First strategy evaluating temporal clauses before spatial clauses

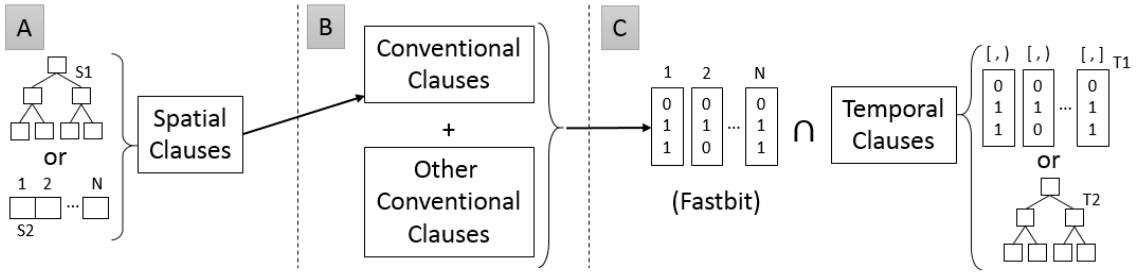


Figure 2. Second strategy evaluating spatial clauses before temporal clauses

Figure 1 depicts how the three main query execution phases can be executed. Phase A is the index processing until we can have the results in the form of conventional clauses which will be used in phase B so to incorporate all other conventional clauses for execution in Fastbit at phase C. It is important to mention the processing options that can be used in phase A were named T1 (temporal bitmap vectors using binning technique), T2 (B+-tree option for organizing records), S1 for HSB-index, and S2 for SB-index.

Figure 2 above outlines the second approach where the roles are inverted while running spatial and temporal query predicates. Phase A would have a first index processing by running option S1 (HSB-index) or S2 (SB-index) and the formed results would be incorporated to the other conventional clauses in phase B. Phase C would then come to limit data according to temporal clauses intersected to Fastbit results. It is worth to mention that we still have to investigate T1 (temporal bitmap vectors using binning

technique) and T2 (B+-tree organization for temporal data). Furthermore, T1 option has a promise of good performance as it can be intersected directly with bit-to-bit operation in Fastbit.

Considering the picture, we can name temporal predicate types as T, spatial predicate types as S, S1 being HSB-index, S2 being SB-index, T1 being the temporal bitmap using binning technique, and T2 being the B+-tree organization for temporal data. If we combine those abbreviated names, we have the eight combinations that should be evaluated in performance tests: T1_S1 (Temporal predicate is processed first using bitmapped temporal organization, Spatial predicate is processed second using HSB-index), T1_S2, T2_S1, T2_S2, S1_T1, S1_T2, S2_T1, and S2_T2.

Following the other indexes, the implementation of each alternative should use as development platform the C++ language under a Linux machine, PostgreSQL as database, and Fasbit as bitmap indexing solution.

3.2. Validation

Tests of the implemented alternatives should be performed so to exercise data with different query window sizes as well as different temporal spans. The GQM methodology should be used to define the needed performance tests and all the data that needs to be gathered while performing them. The plan is to use data generated by data warehouse benchmarks, such as SSB (Star Schema Benchmark) (O'NEIL, P. E. et al., 2009), Spatial SSB (Spatial Star Schema Benchmark) (Nascimento et al., 2011), and Spadawan Benchmark (Siqueira et al., 2010) and adapt the data generated to include spatio-temporal characteristics. The gathered results would then be evaluated so conclusions on which index strategy works best for each situation simulated by the tests can be taken.

4. Performed Activities

4.1. Current activities

So far, we can list as current activities: a) literature review so to get involved in the activities already performed and up to the challenge for the new activities to be performed; and b) proposal discussion taking in account the alternatives that can be taken.

4.2. Future activities

Future activities comprise: a) reproduce the workbench to run the three other indexes (SB-index and HSB-index to be used as steps in the new index, STB-index to be used as comparison); b) implement the options proposed in the solution; c) plan and execute tests with results collection; and d) compare results and conclude on the best alternatives fitting each situation.

5. Final Considerations

Nowadays, indexing DWs became a major concern with all the promised data volumes and even more in the face of the new non-conventional data demands. STDWs are important on their side as they can map fast changing phenomena as well as many geographical happenings worth studying after data collection. The inclusion of the

mentioned indexes in a working database environment (like Postgresql) can be considered as future work as well as moving the whole spatio-temporal with indexing idea to a No-SQL database where data growth restrictions are lower.

References

Jensen, C., Clifford, J., Elmasri, R., Gadia, S. K., Hayes, P. J., & Jajodia, S. (1994). A Consensus Glossary of Temporal Database Concepts. ACM SIGMOD Record, 23(1), 52-64.

Kimball, R.; Ross, M. The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling. 2nd ed. New York, NY: Wiley, 2002. 464 p.

O'Neil, P. E.; O'Neil, E. J.; Chen, X. Star Schema Benchmark. Disponível em: <http://www.cs.umb.edu/~poneil/StarSchemaB.PDF>. Acesso em: Jul. 2013

Nascimento, S. M.; Tsuruda, R. M.; Siqueira, T. L. L.; Times, V. C. T.; Ciferri, R. R.; Ciferri, C. D. A. The Spatial Star Schema Benchmark. In: Brazilian Symposium ON Geoinformatics (GEOINFO), 12nd, 2011, Campos do Jordão, SP. Proceedings... São José dos Campos, SP: MCT/INPE, 2011. p.73-84.

Roddick, J. (1995). A Survey of Schema Versioning Issues for Database Systems. Information and Software Technology, 37(7), 383-393.

Siqueira, T. L. L. SB-index: um índice espacial baseado em bitmap para data warehouse geográfico. 2009. 118 f. Dissertação (Mestrado em Ciência da Computação) - Departamento de Computação, Universidade Federal de São Carlos, São Carlos. 2009

Siqueira, T. L. L.; Ciferri, R. R.; Times, V. C.; Ciferri, C. D. A. Benchmarking spatial data warehouses. In: International Conference On Data Wareahousing and Knowledge Discovery (DaWaK), 12th, 2010, Bilbao, Spain. Proceedings... Berlin / Heidelberg: Springer, 2010. p. 40-51.

Siqueira, T. L. L.; Ciferri, C. D. A.; Times, V. C.; Ciferri, R. R. The SB-index and the HSB-index: efficient indices for spatial data warehouses. Geoinformatica, Hingham, MA, v. 16, n. 1, p. 165-205, jan. 2012.

Tsuruda, R. M. STB-index: Um Índice Baseado em Bitmap para Data Warehouse Espaço-Temporal. 2013. 81 f. Dissertação (Mestrado em Ciência da Computação) – Departamento de Computação, Universidade Federal de São Carlos, São Carlos. 2013

Wu, K.; Stockinger, K.; Shoshani, A. Breaking the Curse of Cardinality on Bitmap Indexes. International Conference on Scientific and Statistical Database Management, 20., 2008, Hong Kong. **Proceedings…** Berlin/Heidelberg: Springer-Verlag, 2008. p.348-365

Wu, K.; Otoo, E. J.; Arie, S. Optimizing Bitmap Indices With Efficient Compression. ACM Transactions on Database Systems, v.31, n.1, p.1-38, 2006

Wu, M.-C.; Buchmann, A. P. Research Issues in Data Warehousing. In: International Conference on Data Engineering, 14., 1998, Orlando. **Proceedings…** Washington: IEEE Computer Society, 1998. p.220-230