

# Utilização do Processo de Descoberta de Conhecimento Para Enriquecer um Banco de Dados Geográfico e Orientado a Objeto

Nara Martini Bigolin

Faculdade de Informática –ULBRA- Rua Miguel Tostes, 101 – Canoas – RS -Brasil

nara@ulbra.tche.br

## Resumo

*O processo de Descoberta de Conhecimento em Bancos de Dados Geográficos (DCBDG) busca encontrar padrões, informações e relacionamento espacial implícitos em dados descritivos e espaciais armazenados em bancos de dados geográficos. Este tipo de descoberta é muito importante para o entendimento dos dados espaciais, através da recuperação destes relacionamentos com o objetivo específico de enriquecer com dados descritivos o banco de dados geográficos. Isso motivou o estudo de um mecanismo para automatizar o processo de descoberta. Este artigo apresenta uma arquitetura, bem como um estudo de caso para validar a arquitetura, que permite automatizar as etapas do processo de descoberta de conhecimento em banco de dados espaciais e orientado a objetos objetivando enriquecer este banco.*

**Palavras chaves:** Banco de Dados, Mineração de dados, Sistemas de Informação Geográfica

## Abstract

*The Knowledge Discovery in Spatial Databases (KDSD) is the extraction of interesting spatial patterns and features, general relationship between spatial and nonspatial data which are not explicitly stored in spatial databases. Such discovery may play an important role to understanding spatial data, capturing intrinsic relationships between spatial and nonspatial data. This motivates the study and development of mechanisms to automates knowledge discovery for large object-oriented spatial databases. This article presents an architecture, as well as a study of case to validate the architecture, that allows to automatize the knowlegde discovery in object-oriented and spatial databases process.*

**Keywords:** Databases, Spatial Data Mining, Geographic Information Systems

## 1. Introdução

Descoberta de conhecimento em banco de dados (em inglês *Knowlegde Discovery in Databases*) é um conceito utilizado para denominar a exploração de informações implícitas em grandes volumes de dados. Essa tecnologia surgiu pela necessidade e dificuldade de explorar grandes bancos de dados. A Descoberta de Conhecimento em Bancos de Dados Geográficos (DCBDG) é um tipo particular de descoberta, pois está ligada à extração de características e padrões espaciais interessantes, à identificação de relacionamentos entre dados espaciais e não-espaciais ou

descritivos, restrições entre objetos geográficos e outras características não explicitamente armazenadas nestes bancos de dados. Os dados convencionais descrevem características como: nome e população de um município. Os dados espaciais representados por coordenadas x,y descrevem a localização de dados descritivos sobre a superfície da terra e são representados, geralmente, por três primitivas geométricas: ponto, linha e área.

Neste contexto, dois enfoques diferentes, porém complementares, foram desenvolvidos, na tentativa de solucionar esses problemas: um, pela comunidade de Inteligência Artificial e outro, pela comunidade de Banco de Dados. A comunidade de Banco de Dados desenvolveu Sistemas de Gerência de Banco de Dados (SGBD), que tem por objetivo oferecer ferramentas que possibilitam o armazenamento e a manipulação de grandes quantidades de informação estruturada e um acesso rápido a elas. A maior preocupação no desenvolvimento deste tipo de sistema é o aspecto de modelagem dos dados, as linguagens de consulta e a eficiência na recuperação da informação.

A comunidade de Inteligência Artificial interessou-se pela mineração de dados (extração de conhecimento a partir de pequenas quantidades de dados) e a aprendizagem a partir de um pequeno conjunto de informações. A combinação dessas duas abordagens originou a área de Descoberta de Conhecimento em Banco de Dados (DCBD) [Fay96]. As técnicas tradicionais de aprendizagem descobrem conhecimento a partir de dados extraídos do mundo real, enquanto as técnicas de mineração utilizam dados extraídos de bancos de dados. No primeiro caso, os dados de entrada para a mineração são representados em uma estrutura simples, e as informações relevantes são definidas pelo especialista. Já no segundo caso, além da estrutura e os tipos de dados serem complexos, o volume de informações é muito grande e, parte deste conjunto, pode ser irrelevante. Assim, é necessário efetuar dois tratamentos: um, para encontrar um subconjunto de informações apropriado e outro para adaptar essas informações em uma estrutura [Big97] aceitável pelas ferramentas de mineração de dados [Big98]. A seleção das informações relevantes é tão importante quanto o reconhecimento de padrões, pois sem os dados apropriados, dificilmente algum conhecimento útil será extraído do banco de dados [Hal00]. Para selecionar a amostra adequada é fundamental que haja um entendimento do domínio da aplicação e um prévio conhecimento dos dados relevantes, para que se possa estabelecer as metas do processo de descoberta de conhecimento do ponto de vista do usuário.

O segundo tratamento consiste em simplificar a estrutura complexa dos dados para um formato reconhecível pelas técnicas de mineração de dados. A simplificação é realizada em dois momentos: primeiro é simplificada a estrutura orientada a Objetos e após os dados espaciais, pois a mineração é realizada unicamente nos dados espaciais, sem utilizar as informações descritivas do banco. Neste contexto, o artigo descreve a arquitetura de um sistema que automatiza todas as etapas do processo de descoberta de conhecimento em banco de dados, dando ênfase ao pré-processamento. Apresenta, também, um estudo de caso realizado para validar a arquitetura proposta. O restante do artigo está organizado da seguinte forma: a Seção 2 apresentada as etapas do processo de DCBD. Na Seção 3, é descrita a arquitetura para automatizar este processo, suportando dados espaciais, armazenados em estruturas orientadas a objetos. A Seção 4 apresenta um estudo de caso realizado para validar a arquitetura e, finalmente, na Seção 5 são apresentadas as considerações finais.

## 2. Processo de Descoberta de Conhecimento em Estruturas e Tipos de Dados Complexos

O processo de DCBD [Fay96], conforme já mencionado, tem como principal objetivo extrair regras e informações implícitas de grandes banco de dados. Para isso, quatro etapas sucessivas foram definidas : seleção e transformação dos dados; mineração dos dados e avaliação e apresentação do conhecimento. Essas etapas podem ser generalizadas em três grandes fases: pré-processamento, mineração e pós-processamento de dados.

O *pré-processamento* é uma das fases mais demoradas do processo de DCBD, e, segundo pesquisas, consome cerca de 80 % dos esforços necessários para concluir todo o processo. Nessa fase são realizadas as seguintes tarefas:

- *seleção de dados*: os dados relevantes para a mineração de dados são identificados e agrupados, gerando uma amostra do banco de dados;
- *limpeza dos dados*: eliminação de ruído e inconsistência dos dados;
- *transformação de dados*: conversão dos dados para um formato interpretável pelas ferramentas de mineração de dados.

A *mineração de dados* é a etapa em que se definem os padrões a serem usados para a extração do conhecimento e selecionam-se os algoritmos de mineração, que serão aplicados sobre os dados selecionados. Essa etapa destaca-se de tal forma dentre as etapas do processo de KDD, que levou alguns autores a considerá-la como sinônimo de descoberta de conhecimento [BAS 01].

O *pós-processamento* é realizado através das etapas de avaliação e apresentação dos padrões, que são responsáveis pela identificação e análise dos padrões interessantes que representam conhecimento, bem como, a forma como o conhecimento extraído será apresentado ao usuário.

A *seleção de dados* a partir de grandes bancos de dados pode ser realizada através de linguagens de consulta. A mais comumente usada é o SQL (*Structure Query Language*) [ISO92]. Ginter [] estendeu essa linguagem para suportar dados espaciais, incorporando funções espaciais na cláusula WHERE. Em paralelo, Han [Han94] estendeu o SQL tradicional para realizar a mineração de dados armazenados em banco de dados relacional. Onde, além das cláusulas tradicionais SELECT, FROM e WHERE, duas cláusulas foram definidas: DataMining e Rule. A primeira especifica o algoritmo de mineração empregado e a segunda que tipo de regra será extraída.

Para a limpeza dos dados não existe nenhuma técnica específica, pois a limpeza depende do tipo de impureza dos dados. Nos dados espaciais, um tipo de limpeza que é realizada é a redução de coordenadas cartesianas consideradas irrelevantes.

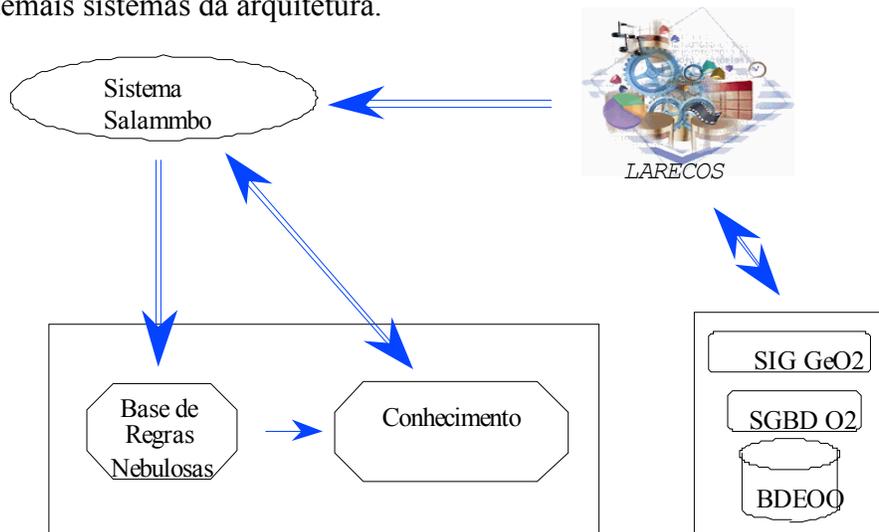
A complexidade da *transformação dos dados* depende da complexidade de estrutura. Por exemplo, a conversão da estrutura relacional é mais simples, visto que as informações são armazenadas em tabelas ou relações, formato este, entendido pelas ferramentas de mineração de dados. Entretanto, estruturas orientadas a objetos são mais complexas, devido suas características de relacionamento como herança, composição, agregação e dependência. Estes relacionamentos não são entendidos pelas ferramentas de mineração de dados. Outra dificuldade está na conversão

de dados espaciais (coordenadas cartesianas x,y) para um formato possível de extrair alguma informação útil.

Neste contexto, não existe nenhuma linguagem de consulta capaz de realizar a mineração a partir de um banco de dados espacial e orientado a objetos, devido a complexidade a nível semântico das duas estruturas. Assim, a seguir é descrita a arquitetura proposta que é composta de vários sistemas que permitem a descoberta de conhecimento totalmente automático a partir de um banco de dados espaciais e orientado a objetos.

### 3. Uma Arquitetura para Descoberta de Conhecimento em Bancos de Dados Geográficos

A arquitetura proposta, conforme ilustra a Figura 2, é formada por quatro módulos principais, que são sistemas independentes: um Sistema Gerenciador de Banco de Dados Orientado a Objetos (SGBDOO) O2 [Ban92], que auxilia na manipulação da estrutura orientada a objetos; um Sistema de Informações Geográficas Geo2 [Ray95], que manipula os dados espaciais; uma ferramenta de mineração de dados, chamada Salammbô [Mar98], para gerar uma árvore de decisão nebulosa e; uma linguagem de consulta denominada LARECOS [Big99], que integra os três sistemas anteriores. O LARECOS é uma linguagem de consulta estendida que permite selecionar dados de um banco de dados geográfico, tratar somente informações espaciais e extrair conhecimento dessas informações. Neste artigo, a ênfase será dada para o LARECOS, o qual foi construído para integrar os demais sistemas da arquitetura.



**Figura 2** - Arquitetura para automatização do processo de DCBDG

#### 3.1 SGBDOO O2

O SGBDOO O<sub>2</sub> oferece um ambiente de desenvolvimento de aplicações para Banco de Dados Orientados a Objetos que inclui um conjunto de ferramentas de interface e permite a construção de um ambiente de desenvolvimento de aplicações geográficas. Seu ambiente gráfico (O<sub>2</sub>Tools) e as ferramentas (*browsers* gráficos, editores gráficos, compilador, etc) reduzem o ciclo de desenvolvimento. As aplicações no O<sub>2</sub> podem ser desenvolvidas utilizando O<sub>2</sub>C (sua linguagem

de quarta geração orientada a objetos e extensão da linguagem C, que integra programação, manipulação do banco de dados e a geração da interface ao usuário), C, C++ e Smalltalk. Com este arsenal de linguagens disponíveis, o sistema possibilita a incorporação de novas classes, de acordo com as necessidades do usuário e a reutilização de códigos pré-existentes, criados em algumas das linguagens mencionadas.

O O2 é um ambiente que favorece a prototipação, pois oferece a ferramenta integrada O2Look para realizar o projeto e a implementação de interfaces gráficas. Ele permite criar, automaticamente, o *display* gráfico de qualquer objeto no banco de dados e oferece um menu através do qual os métodos de um objeto podem ser executados

O *toolbox* (O2Kit) é um componente que constitui uma biblioteca de classes pré-definidas que podem ser utilizadas dentro de contexto de uma aplicação para definir subclasses a partir de suas superclasses. Com relação aos dados geográficos, o O2Kit permite a manipulação de imagens e a construção de aplicações gráficas. O *O2Graph* fornece um editor gráfico que pode ser invocado para criar, exibir e editar qualquer tipo de gráfico, fornecendo diferentes *layouts* (*grid*, *tree*, *DAG*, *graph*, etc).

### 3.2 Sistema de Informação Geográfica Geo2

Os Sistemas de Informação Geográfica (SIG) são sistemas capazes de armazenar e manipular dados descritivos e a espacialidade (coordenadas x,y) dos dados geográficos. Um SIG trata, também, os relacionamentos espaciais entre objetos geográficos, dentre os quais, podemos citar os métricos, os direcionais e os topológicos [Bor98].

O *Geo2* é um SIG que foi desenvolvido pelo Instituto Geográfico Nacional da França com suporte ao O2 [Dav93]. A escolha do *Geo2* deu-se pelo fato de ser um SIG acoplado a um SGBDOO.

### 3.3 Sistema de Construção de Árvore de Decisão Nebulosa: SALAMMBO

Árvore de decisão é uma técnica de aprendizagem que classifica exemplos em um número finito de classes [Qui86]. O objetivo desta técnica é elaborar e testar hipóteses sem a intervenção do usuário. A representação gráfica de uma árvore é dada por:

- um nó intermediário: corresponde a uma pergunta sobre a propriedade do objeto, representando um atributo.
- um arco: corresponde a um valor de atributo.
- um nó final corresponde a um conjunto da mesma classe.

Percorrendo a árvore (respondendo as questões dos nós intermediários e seguindo as ligações correspondentes), chega-se a um nó final que é a classe do objeto.

A arquitetura proposta baseia-se em árvores de decisão nebulosa, que são um tipo especial de árvore. Elas são baseadas na teoria de conjuntos nebulosos [Zad65],

diferenciando-se das árvores de decisão tradicionais em dois aspectos: utilizam critérios de partição imprecisos e, referem-se ao modelo de raciocínio aproximativo. As árvores de decisão tradicionais utilizam uma condição precisa, onde um valor satisfaz uma única condição. Nas árvores de decisão nebulosas, uma pergunta é feita sobre um valor nebuloso, gerando um resultado nebuloso. Como exemplo de valor nebuloso podemos citar: grande, próximo, longe, etc.

Os dados de entrada desta técnica são valores nebulosos e a saída é uma árvore de decisão nebulosa. Esta escolha foi feita devido a necessidade de tratar as imprecisões dos dados espaciais. O sistema SALAMMBO [Mar98] foi utilizado para implementar essa técnica.

### 3.4 Linguagem de Consulta LARECOS

O LARECOS é uma linguagem de consulta para a extração de conhecimento em bancos de dados espaciais orientados a objetos. Ele foi implementado na linguagem *O2C* (linguagem acoplada ao SGBD *O2*) e apresenta a seguinte sintaxe [Big99]:

```
DATAMINING Algoritmo de Mineração
FUZZY Função Nebulosa
ASSOCIATION Função Matemática
GENERALIZATION Função de Generalização
WITH
SELECT objetos
FROM x in Database1
WHERE Condição
```

A cláusula *DATAMINING* permite a especificação do algoritmo de mineração de dados a ser utilizado na extração do conhecimento.

A cláusula *FUZZY* é utilizada para determinar uma função da teoria dos conjuntos nebulosos [Zad65]. Ela serve para corrigir os erros gerados pelas cláusulas *ASSOCIATION* e *GENERALIZATION*.

A cláusula *ASSOCIATION* determina a associação dos objetos espaciais, através do uso de funções matemáticas como, por exemplo, função da distância euclidiana.

A cláusula *GENERALIZATION* permite especificar o algoritmo de generalização [Mul93], que é utilizado para reduzir a quantidade de informações espaciais, como, por exemplo, a função que reduz um conjunto de coordenadas cartesianas em uma única coordenada.

O resultado da cláusula *SELECT* é um conjunto de dados selecionados a partir de um banco de dados espacial e orientado a objetos. Neste caso, a linguagem de consulta precisa recuperar, além dos dados convencionais, os dados espaciais. Para isso, a extensão de Egenhofer foi empregada da seguinte maneira:

```
SELECT x.objeto  
FROM x in Banco de Dados Geográfico  
WHERE Condição Espacial
```

Esta consulta permite encontrar um conjunto de objetos a partir de um banco de dados geográfico, delimitado pela condição espacial especificado na cláusula WHERE.

Cada uma dessas cláusulas é validada no estudo de caso, descrito na próxima seção, segundo as fases do processo de descoberta de conhecimento.

#### 4. Estudo de Caso

O estudo de caso realizado para validar a arquitetura proposta na Seção 3 foi baseado de um banco de dados cedido pelo Instituto Geográfico Nacional (IGN) da França. O objetivo do estudo de caso foi classificar objetos espaciais (casas) considerando apenas as coordenadas x,y em áreas urbanas e não urbanas.

Com base no *LARECOS*, definiu-se a seguinte estrutura:

```
DATAMINING Algoritmo de construção de árvore de decisão nebulosa  
FUZZY Função de pertinência  
ASSOCIATION Função distância  
GENERALIZATION Função de centralização  
WITH  
SELECT x.casa  
FROM x in Banco de Dados 1  
WHERE  
x.casa ∈ NaArea(CoordPtMin, CoordPtMax)
```

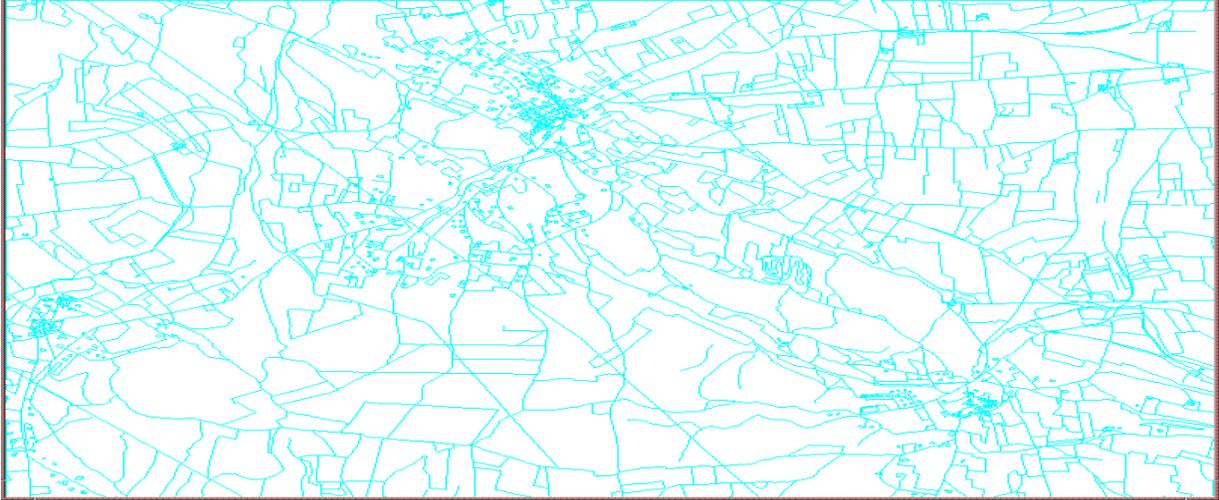
A cláusula DATAMINING especifica o algoritmo de construção da árvore de decisão nebulosa, que neste estudo, foi o sistema SALAMMBO.

A cláusula FUZZY utiliza uma função de pertinência que permite identificar se um objeto pertence, ou não, a um determinado grupo de objetos espaciais. Esta função elimina as imprecisões resultantes dos outros processamentos (cláusula ASSOCIATION e GENERALIZATION). A cláusula ASSOCIATION está vinculada a uma função métrica, que permite estabelecer a distância entre dois pontos, definindo assim, um relacionamento espacial entre os mesmos. A cláusula GENERALIZATION agrupa um conjunto de pontos em um ponto único.

De acordo com as fases do processo de DCBD, as cláusulas do LARECOS serão detalhadas nas próximas seções:

## 4.1 Pré-processamento

A Figura 3 ilustra uma região da França sobre a qual foi realizado este estudo de caso.

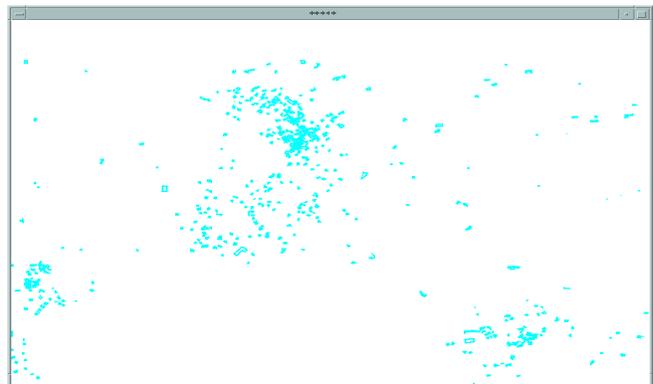


**Figura 3:** Mapa de uma região da França

Com base no mapa da Figura 3, uma consulta espacial para selecionar todos os objetos do tipo área foi realizada, utilizando o Geo2. Essa consulta e seu resultado estão representados na Figura 4.

A função *igual*, especificada na cláusula WHERE, determina a seleção dos objetos espaciais onde o primeiro ponto é igual ao último, requisito necessário para determinar um objeto espacial do tipo superfície.

```
SELECT x.objetos
FROM x in Cidades
WHERE igual
(CoordPremier,CoordDernier)
```



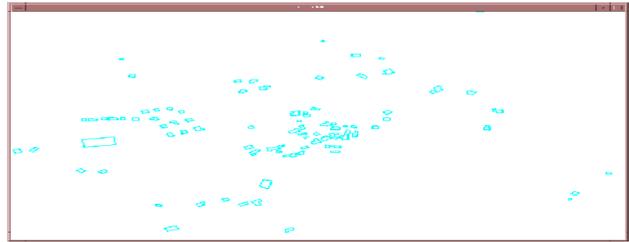
**Figura 4:** Representação de casas de uma cidade

Após, uma consulta foi realizada para selecionar apenas uma das três cidades, ou seja, foi selecionado um conjunto de casas pertencentes a uma região.

```

SELECT x.casa
FROM x in Database1
WHERE
x.casa ∈ NaArea(CoordPtMin, CoordPtMax)

```



**Figura 5 : Consulta que seleciona uma área**

Isso foi determinado na condição da cláusula WHERE, onde a função *NaArea* (que faz cálculos espaciais) foi utilizada. Essa função foi implementada na linguagem O2C. Ela determina se o objeto *casa* está numa área definida pelas coordenadas *CoordPtMin* e *CoordPtMax* (veja Figura 5).

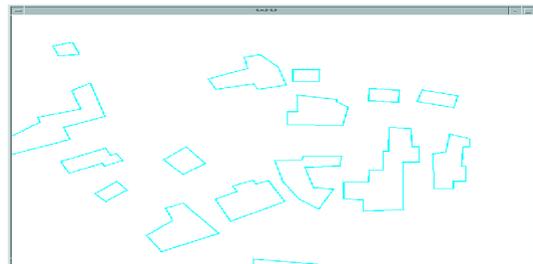
Uma vez selecionados os dados, os mesmos devem ser preparados para a mineração de dados. Esta preparação é realizada durante o pré-processamento, que na nossa proposta, é realizada pela cláusula GENERALIZATION. Ela utiliza funções de generalização, que permitem reduzir todos os pontos que representam as casas em um único ponto, diminuindo a quantidade de informações espaciais.

A consulta e o resultado são apresentados na Figura 6.

```

GENERALIZATION centro(p.objeto)
WITH
SELECT x.objetos
FROM x in Cidadee
WHERE
x.objetos inRegião
(CoordPtMin,CoordPtMax)

```



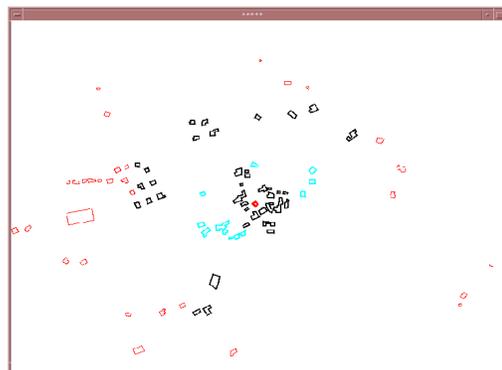
**Figura 6: Consulta usando generalização**

A cláusula ASSOCIATION determina o relacionamento entre os objetos espaciais. Para identificar o relacionamento entre os objetos selecionados anteriormente foi criada a função matemática *d*, que calcula a distância Euclidiana entre dois pontos *p1* e *p2*. A sintaxe da consulta e seu resultado são apresentados na Figura 7.

```

ASSOCIATION distancia (centro(p.objeto))
GENERALIZATION centro(p.objeto)
WITH
SELECT x.objetos
FROM x in Cidades
WHERE
x.objetos inRegião
(CoordPtMin,CoordPtMax)

```



### Figura 7: Utilização da cláusula Association

A cláusula NEBULOSA, representada no LARECOS por FUZZY, permite tratar as imprecisões geradas nas etapas anteriores. Isso devido à utilização da função de generalização, que transforma um conjunto de pontos em um único ponto. As distâncias geradas entre os pontos, são analisadas novamente, utilizando as modalidades nebulosas (perto, longe, muito longe). A sintaxe da cláusula FUZZY tem a seguinte estrutura:

```
FUZZY pertinência (distancia(centro(p.objeto)))
ASSOCIATION distancia (centro(p.objeto))
GENERALIZATIO centre(p.objeto)
WITH
SELECT x.objetos
FROM x in Cidades
WHERE
x.objetos inRegião(CoordPtMin,CoordPtMax)
```

Dada uma casa  $c$ , a função de pertinência calcula o número de casas que pertencem a três áreas nebulosas: *próximo*, *longe* e *muito longe*, ou seja, para cada casa  $c'$  diferente de  $c$ , a distância  $d(pc, pc')$  é analisada. Assim, o número de casas na área definida pela modalidade *próximo*, é dada pela medida nebulosa de cardinalidade:  $Nr\_próximos = \sum_{c' \in C} \mu_{próximos}(c')$  e assim para todas as outras modalidades.

### 4.3 Mineração de dados

A cláusula DATAMINING possibilita a definição do algoritmo de mineração de dados que, neste exemplo, é o algoritmo de construção de árvores de decisão nebulosa. Ele descobre o número de casas que pertencem, ou não, a uma zona urbana, considerando um intervalo impreciso definido pela teoria dos conjuntos nebulosos. A entrada dos dados foi gerada pelas cláusulas acima descritas, e alguns exemplos da base de exemplos são mostrados na Tabela 1.

**Tabela 1: Exemplos a partir da base de aprendizagem**

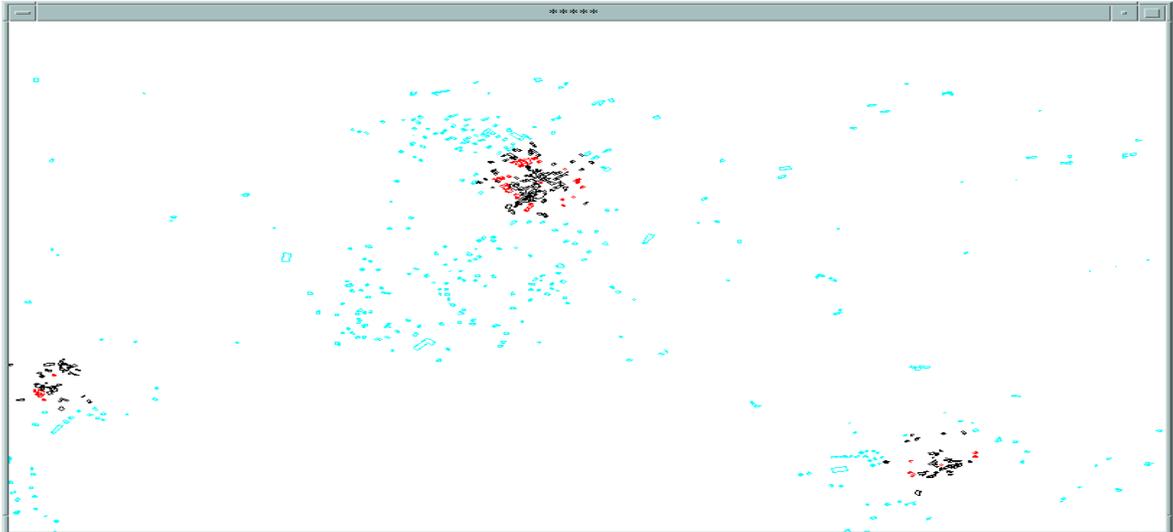
<i>Casa</i>	<i>Nr próximos</i>	<i>Nr longe</i>	<i>Nr super longe</i>	<i>Urbana</i>
Casa1	3.2	5.4	4.3	Não
Casa2	3.5	3.2	1.2	Não
Casa3	1.3	4.2	3.5	Não
Casa4	6.7	1.2	3.6	Sim
Casa5	3.4	3.2	5.5	Sim
Casa6	4.3	3.3	2.2	Sim

O resultado da aplicação do sistema SALAMMBO gera uma árvore de decisão nebulosa, que pode ser interpretada em forma de regras.

### 4.4 Pós-processamento

Uma vez encontrada a base de regras, elas devem ser validadas. Existem vários métodos para essa validação. No estudo de caso, as regras foram validadas utilizando a técnica de base/teste da seguinte forma: as regras são aprendidas a partir de uma cidade, e após, essas mesmas regras são utilizadas em outras cidades (conforme Figura 8).

No estudo de caso, a região analisada (Figura 3) é formada de três cidades *T1*, *T2* e *T3*. A base de exemplos foi gerada a partir de *T1* (Figura 4) e o conjunto de teste foi gerado a partir das outras duas cidades. A média de erros na classificação das cidades *T2* e *T3* foi de 10.1%, ou seja, dadas 413 casas a partir de uma nova zona, 371 casas foram bem classificadas em urbana ou não urbanas.



**Figura 8: Resultado final da classificação de casas em perto, longe e muito longe**

## 5. Conclusão

Neste artigo foi apresentada uma proposta para automatizar o processo de DCBD em um banco de dados real. Inicialmente, foi descrito todo o processo de descoberta para situar a proposta. Em seguida, foi apresentada uma arquitetura composta de quatro sistemas e um estudo de caso para validar a abordagem.

A originalidade deste artigo situa-se na proposta de uma arquitetura que automatiza todas as etapas de processo de DCBDG e, principalmente, a etapa de pré-processamento que é a mais carente de ferramentas.

A tecnologia de descoberta de conhecimento em banco de dados é uma área de grandes perspectivas, onde o mais importante é a manipulação de técnicas multidisciplinares, que visam tratar todo o processo de uma maneira eficaz e correta.

## Referências Bibliográficas

- [Agr93] R. Agrawal and T. Imielinski and A. Swami. Database Mining: A Performance Perspective. *IEEE Transactions on Knowledge and Data Engineering*, December, 6(5): 914-925, 1993.
- [Ban92] F. Bancilhon, C. Delobel, and P. Kanellakis. *Building an Object-Oriented Databases Systems: The story of O2*. Morgan Kaufmann, 1992.
- [Big97] N.M. Bigolin and P. Brezillon. An Experiment using Context in Translation from System's Requirements to Conceptual Model. *Proc. International and Interdisciplinary Conference on Modeling and Using Context*. Rio de Janeiro. Fevrier, 1997.
- [Big98] N.M. Bigolin and C. Marsala. Fuzzy Spatial OQL for Fuzzy Knowledge Discovery in Databases. *2nd European Symposium on Principles of Data Mining and Knowledge Discovery*. J.M. Zytkow and M.Quafafou (Eds.) Springer Verlag Nantes, France September. pages 246-254. 1998. Lecture Notes in Computer Science. 1510.
- [Big99] N.M. Bigolin. *Découverte de Connaissances a partir de Base de Données Spactiales Orientées Objets : Le Système LARECOS*. Tese de Doutorado. Université de Paris VI. Paris. 1999.
- [Bor97] Borges, K. A. V. *Modelagem de dados geográficos: uma extensão do modelo OMT para aplicações geográficas*. Belo Horizonte: Fundação João Pinheiro, 1997. Dissertação de Mestrado.
- [Dav93] B. David and L. Raynal and G. Shorter and V. Mansart. GeO2: Why Objects in a Geographical DBMS. *Advances in Spatial Databases, Third International Symposium*, Springer Verlag. 1993. Singapore p.264-276
- [Ege94] M. Egenhofer. Spatial sql: A query and presentation language. *IEEE Transactions on Knowledge and Data Engineering*, 6(1):86--95, 1994.
- [ISO92] ISO. *Database language sql*. ISO/IEC 9075, 1992.
- [Hal2000] Halmenschlager, C. *Utilização de agentes na descoberta de conhecimento*. Porto Alegre: PPGC da UFRGS, 2000. (TI-955).
- [Han01] J. Han, M. Kamber, A. Tung. *Spatial clustering methods in data mining: a Survey*. *School of Computing Science*, Simon Fraser University, Burnaby, BC Canada, v5a156, 2001.
- [Mar98] C. Marsala. *Apprentissage inductif en présence de données imprécises: construction et utilisation d'arbres de décision flous*. Tese de doctorado, Université Pierre et Marie Curie, Paris, France, Janvier 1998.
- [Qui86] J.R. Quilain. Induction of decision Trees. *Machine Learning*, 1:81-106,1986.
- [Ray95] L. Raynal and G. Schorter. *Geo2. Technical report*, COGIT - IGN, 1995.
- [Zad65] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338--353, 1965.