CDD: 005

MINERAÇÃO DE DADOS E PÓS-PROCESSAMENTO EM PADRÕES DESCOBERTOS

DATA MINING AND POS-PROCESSING IN PATTERNS DISCOVERABLE

Leonardo Aparecido de Almeida Calil[1], Deborah Ribeiro Carvalho[1,2], Celso Bilynkievycz dos Santos[3], Maria Salete Marcon GomesVaz[4]

[1] Universidade Tuiuti do Paraná, Curitiba-PR
[2] Instituto Paranaense de Desenvolvimento Econômico e Social - IPARDES, Curitiba-PR
[3] Universidade Estadual de Ponta Grossa - UEPG, UTFPR - Ponta Grossa-PR e Centro Universitário Claretiano, Batatais-SP
[4] UEPG. E-mails: leonardo_calil@hotmail.com, deborah@ipardes.pr.gov.br, bilynkievycz@globo.com, salete@uepg.br

Recebido para publicação em 15/10/2008
Aceito para publicação em 30/11/2008

RESUMO

Com o crescente aumento da quantidade de dados armazenados, acresce também a possibilidade de obtenção de informações preciosas pelos gestores, porém, muitas vezes, o seu volume inviabiliza a percepção e avaliação, por ultrapassar a capacidade humana de análise e interpretação. Uma das alternativas para facilitar esta atividade é a adoção do processo KDD – Knowledge Discovery in Database, que preve três etapas: Pré-processamento, Mineração de Dados e Pós-processamento. Este artigo apresenta e discute as etapas de Mineração de Dados e Pós-processamento, não apenas em nível conceitual, mas também facilitando a compreensão a partir de exemplos construídos em cada situação.

Palavras-chave: Descoberta de conhecimento. Avaliação. Apoio à decisão.

ABSTRACT

The possibilities of obtaining valuable information by managers is growing as the number of stored data is increasing. However, sometimes, the huge amount of data makes impracticable to evaluate and understanding it. One of the ways to facilitate this activity is to adopt the KDD (Knowledge Discovery in Database) process, which has three stages: previous data processing; data mining and post data processing. The present article has the aim to discuss the stages of data mining and post data processing not only on the conceptual level, but also facilitating the comprehension by demonstrating examples from each situation.

Keywords: Knowledge discovery. Evaluation. Decision support.

1 Introdução

O ser humano continuamente toma decisões ou simplesmente chega a determinadas conclusões baseadas no conhecimento que ele acumula ao longo de sua vida. A "era da informação" é caracterizada pela grande expansão no volume de dados gerados e armazenados (HAND, 1999). Uma grande parte desses dados está armazenada em bases de dados e podem ser facilmente acessadas pelos seus usuários.

Essa situação tem gerado demandas por novas técnicas e ferramentas que, com eficiência, transformem os dados armazenados e processados em conhecimento. Ou seja, viabilize que o usuário/gestor efetivamente se utilize destes como apoio ao processo decisório. Para facilitar a recuperação e uso destes dados uma das alternativas que pode ser utilizada é o processo de KDD – *Knowledge Discovery in Database*. O processo KDD é composto por diversas etapas, a saber (FAYYAD et al.1996):

Seleção de Dados: prevê a coleta e seleção dos dados;

Limpeza: prevê a análise dos dados coletados, verificando a existência de ruídos, tratamento de valores ausentes, entre outras;

Transformação ou Enriquecimento dos Dados: dedica-se à incorporação/criação de novos dados a partir dos já existentes;

Mineração de Dados: consiste na aplicação de um algoritmo que, efetivamente, procura por padrões/relações e regularidades, em um determinado conjunto de dados;

Interpretação e Avaliação: verifica a qualidade do conhecimento (padrões) descoberto, procurando identificar se o mesmo auxilia a resolução do problema original que motivou a realização do processo KDD.

Essas etapas podem ser agrupadas em três grandes grupos: pré-processamento, mineração de dados e pós-processamento. O pré-processamento inclui todas as etapas que consideram a preparação da base de dados, cujos dados serão fornecidos como entrada para o(s) algoritmo(s) de Mineração de Dados. O pós-processamento contempla a depuração e/ou síntese dos padrões descobertos. Na grande maioria dos processos KDD, a etapa de pós-processamento se justifica, pois o volume de conhecimento descoberto é tão grande/extenso que dificulta a sua análise e fundamentalmente inviabiliza o seu uso no apoio a tomada de decisão. Isso se deve a vários fatores, como por exemplo: padrões redundantes, relações irrelevantes, entre outros.

Na mineração de dados, podem ser identificados algoritmos correspondentes a três tarefas principais: classificação, descoberta de regras de associação e *clustering*. Por exemplo, na tarefa de classificação, pode-se ter uma aplicação financeira na qual um banco poderia classificar seus clientes em duas classes: "crédito ruim" ou "crédito bom". Em uma aplicação de medicina, um médico poderia classificar alguns de seus pacientes em duas classes: "tem" ou "não tem" uma certa doença.

Neste artigo são discutidos conceitos de duas etapas do processo KDD: mineração de dados e o pós-processamento dos padrões (conhecimento) descobertos. Para tanto, este artigo está estruturado como segue. Na Seção 2, é abordada a conceituação inerente a mineração de dados. Na Seção 3, é descrita a etapa de pós-processamento. Na Seção 4, são discutidas as duas etapas de mineração de dados e o pós-processamento dos padrões. E, finalmente, na Seção 4, são feitas as considerações finais.

2 Mineração de dados

A *Mineração de Dados* consiste em um conjunto de conceitos e métodos com o objetivo de encontrar uma descrição, preferencialmente compreensível e interessante para o usuário, de padrões e regularidades em um determinado conjunto de dados.

Os termos Mineração de Dados e Descoberta de Conhecimento em Base de Dados – KDD muitas vezes são confundidos como sinônimos para identificar o processo de descoberta de conhecimento útil a partir de banco de dados. O termo KDD foi estabelecido no primeiro *workshop* de KDD, em 1989, para enfatizar que o conhecimento é o produto final de uma descoberta baseada em dados (*data-driven*). Desta forma, KDD se refere a todo o processo de descoberta de conhecimento, enquanto Mineração de Dados se refere a uma das etapas deste processo.

Um padrão é definido como um tipo de declaração (ou modelo de uma declaração) sobre o conjunto de dados que está sendo analisado. Uma instância de um padrão é uma declaração em uma linguagem de

alto nível que descreve uma informação interessante descoberta nos dados. A descoberta de relações nos dados compreende todas as instâncias de padrões selecionados no espaço das hipóteses que sejam suficientemente interessantes, de acordo com algum critério estabelecido (KLOSGEN, 1992).

As várias tarefas desenvolvidas em Mineração de Dados têm como objetivo primário a predição e/ou a descrição. A predição usa atributos para predizer os valores futuros de uma ou mais variáveis (atributos) de interesse. A descrição contempla o que foi descoberto nos dados sob o ponto de vista da interpretação humana (FAYYAD et al.1996).

O objetivo da descrição, bem como o da predição, é atendido através de algumas das tarefas principais de Mineração de Dados (ADRIAANS; ZABTINGE, 1996), (FAYYAD et al.1996), (FU, 1996). A seguir, são descritas três dessas tarefas: a de classificação, a de descoberta de regras de associação e a de agrupamento, a qual pode ser utilizada para análise inicial dos dados, possivelmente levando posteriormente à execução da tarefa de classificação.

Classificação

A classificação, por vezes chamada de aprendizado supervisionado (FISHER; HAPANYENGWI, 1993), parece ser a tarefa de Mineração de Dados que tem sido mais estudada ao longo do tempo. Essa tarefa consiste em classificar um item de dado (exemplo ou registro) como pertencente a uma determinada classe, dentre várias classes previamente definidas.

Cada classe corresponde a um padrão único de valores dos atributos previsores (demais atributos que caracterizam o exemplo). Esse padrão único pode ser considerado a descrição da classe. O conjunto de todas as classes é definido como C, e a cada classe C_i corresponde uma descrição D_i das propriedades selecionadas. Desta forma, utilizando estas descrições, é possível construir um classificador que descreva um exemplo e do conjunto de exemplos T como sendo um exemplo pertencendo à classe C_i, quando aquele exemplo satisfaz D_i.

O principal objetivo da construção de um classificador é descobrir algum tipo de relação entre os atributos previsores e as classes (FREITAS; LAVINGTON, 1998). Por exemplo, na Figura 1, o classificador em questão tem como objetivo a identificação da relação existente entre os atributos previsores A_1 e A_2 e os valores da classe ("+" e "-"). O procedimento de construção deste classificador é baseado em particionamentos recursivos do espaço de dados. O espaço é dividido em áreas e a cada estágio é avaliado se cada área deve ser dividida em subáreas, a fim de obter uma separação das classes.

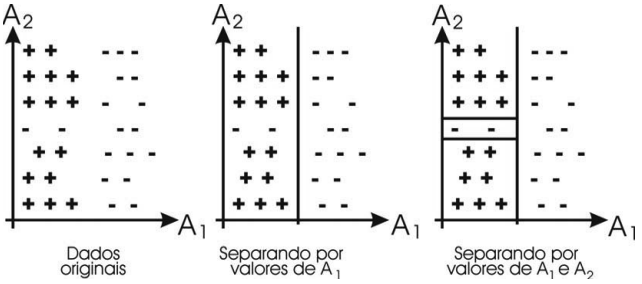

Figura 1 - Exemplo de classificação (FREITAS; LAVINGTON, 1998)

Segundo Breiman e seus colegas (1984), um classificador extraído de um conjunto de dados serve a dois propósitos: predição de um valor e entender a relação existente entre os atributos previsores e a classe. Para cumprir o segundo propósito, é exigido do classificador que ele não apenas classifique, mas também explicite o conhecimento extraído da base de dados de forma compreensível.

Como exemplo, pode-se considerar os dados da Tabela 1, que representam a receptividade de clientes que receberam a divulgação de um determinado produto via mala-direta em adquiri-lo ou não. Os atributos sexo, país e idade são ditos atributos previsores e o atributo compra corresponde ao atributo classe.

Tabela 1- Cadastro de clientes versus compra de livros.

Sexo	País	Idade	Compra
Masculino	França	Jovem	S
Masculino	Inglaterra	Jovem	S
Feminino	França	Jovem	S
Feminino	Inglaterra	Adulto	S
Feminino	França	Adulto	N
Masculino	Alemanha	Jovem	N
Masculino	Alemanha	Jovem	N
Feminino	Alemanha	Jovem	N
Feminino	França	Adulto	N
Masculino	França	Adulto	N

A partir deste conjunto de dados (Tabela 1) é possível extrair o classificador representado pela Figura 2. O algoritmo utilizado neste exemplo foi o C4.5 (Quinlan, 1993).

```
Decision Tree:

Pais = Inglaterra: S (2.0)
Pais = Alemanha: N (3.0)
Pais = Franca:
|   Idade = Jovem: S (2.0)
        |   Idade = Adulto: N (3.0)
```

Figura 2 - Exemplo de classificador gerado a partir de um conjunto de dados.

A fim de contribuir para a compreensibilidade do conhecimento descoberto (relação entre os atributos e as classes), esse conhecimento é geralmente representado na forma de regras "se"... (condições) ... "então"... (classe) ..., cuja interpretação é: "se" os valores dos atributos satisfazem as condições da regra, "então" o exemplo pertence à classe prevista pela regra.

Desta forma, o classificador (árvore de decisão), representado na Figura 2, pode ser transformado nas seguintes regras:

Se pais = Inglaterra então compra = S
Cobertura de 2 exemplos
Se pais = Alemanha então compra = N
Cobertura de 3 exemplos
Se pais = Franca e Idade = Jovem então compra = S
Cobertura de 2 exemplos
Se pais = Franca e Idade = Adulto então compra = N
Cobertura de 3 exemplos

A cobertura representa o número de exemplos da base de dados que satisfazem a condição que corresponde ao antecedente da regra.

Existem vários critérios para avaliar a qualidade das regras descobertas na tarefa de classificação. Os três critérios mais usados são precisão preditiva, compreensibilidade e o grau de interesse do conhecimento descoberto.

Precisão preditiva é normalmente medida com o número de exemplos de teste classificados corretamente, dividido pelo número total de exemplos de teste. Cabe ressaltar que há formas mais sofisticadas de se medir a precisão preditiva (HAND, 1997), mas a forma simples descrita anteriormente é, em sua essência, a forma mais usada na prática. A compreensibilidade geralmente é medida pela simplicidade, a qual por sua vez é medida em função do número de regras descobertas e do número de condições por regra. Quanto maiores estes números, menos compreensível é o conjunto de regras em questão. É importante ressaltar que, mesmo nos paradigmas que tradicionalmente têm a característica de descobrir conhecimento expresso sob uma forma dita compreensível, algumas vezes pode ser gerado um modelo muito complexo, o qual dificilmente satisfaz o requisito de compreensibilidade. Este fato pode decorrer da complexidade existente entre os atributos previsores e as classes, o nível de ruído existente nos dados etc. (FREITAS; LAVINGTON, 1998).

Mesmo que o conhecimento descoberto seja altamente correto do ponto de vista estatístico, ele pode não ser de fácil compreensão. Por exemplo, o conjunto de regras descobertas pode ser grande demais para ser analisado, ou conter redundâncias. Além disso, o conhecimento descoberto pode não ser interessante, representando algum relacionamento previamente conhecido. Desta forma, é importante que o conhecimento descoberto também seja avaliado do ponto de vista do quão interessante ele é para o usuário.

Um importante conceito da tarefa de classificação é a divisão dos dados entre dados de treinamento e dados de teste. Inicialmente, um conjunto de dados de treinamento é disponibilizado e analisado, e um modelo de classificação é construído, baseado nesses dados. Então, o modelo construído é utilizado para classificar outros dados, chamados dados de teste, os quais não foram contemplados pelo algoritmo durante a fase de treinamento. Cabe ressaltar que o modelo construído a partir dos dados de treinamento só será considerado um bom modelo, do ponto de vista de precisão preditiva, se ele classificar corretamente uma alta porcentagem dos exemplos (registros) dos dados de teste. Em outras palavras, o modelo deve representar conhecimento que possa ser generalizado para dados de teste, não utilizados durante o treinamento.

Regras de associação

Nesta tarefa, o objetivo é descobrir regras de associação, que são expressões $X \rightarrow Y$ (lidas como: SE (X) ENTÃO (Y)), onde X e Y são conjuntos de itens, $X \cap Y = \varnothing$. O significado de cada regra

desta natureza é de que os conjuntos de itens X e Y frequentemente ocorrem juntos em uma mesma transação (registro).

Um exemplo de uma regra do tipo $X \rightarrow Y$ poderia ser: 90% dos consumidores que compram pneus e acessórios automotivos também utilizam serviços de manutenção do carro. O valor 90% é dito a confiança da regra, ou seja, representa o número de consumidores que compraram pneus e acessórios automotivos e também utilizaram serviços de manutenção do carro, dividido pelo número de consumidores que compraram pneus e acessórios automotivos.

Uma outra medida para avaliar uma regra de associação é o valor do suporte da regra, que representa a frequência de ocorrência dos itens X e Y em relação à base de dados (AGRAWAL et al.,1993), (SRIKANT; AGRAWAL, 1995).

Formalmente, confiança e suporte são definidos da seguinte forma:

$$\text{Suporte} = | X \cup Y| / N \quad (1)$$

$$\text{Confiança} = | X \cup Y| / |X \quad (2)$$

em que N é o número total de exemplos.
|X| denota a cardinalidade do conjunto X

Como exemplo, pode-se considerar os dados da Figura 3, que representam o aproveitamento de alguns acadêmicos em relação à situação de aprovação nas disciplinas.

A partir deste conjunto de dados (Figura 3), é possível extrair um conjunto de 95 regras, sendo algumas representadas na Figura 4. O algoritmo utilizado neste exemplo foi o Apriori (Borgelt, 2004).

A partir da Figura 4, é possível perceber que as regras são na verdade representadas na forma $Y \leftarrow X$, em que X é o antecedente e Y o consequente. Interpretando a primeira regra, observa-se uma associação entre reprovações nas disciplinas de inglês e lógica, sendo que todos que reprovaram em lógica também reprovaram em inglês (confiança de 100%).

Classificação *versus* regras de associação

A principal diferença entre as tarefas de classificação e de descoberta de regras de associação envolve a questão da predição. A classificação é considerada uma tarefa não determinística, mal-definida, no sentido que, em geral, há vários classificadores diferentes que são igualmente consistentes com os dados de treinamento – mas, provavelmente, com diferentes graus de consistência com os dados de teste, não vistos durante o treinamento. Portanto, a tarefa de classificação envolve a questão da predição (analisa o "passado" para induzir o que ocorrerá no "futuro"). Em contraste, a tarefa de descoberta de regras de associação é considerada uma tarefa relativamente simples, bem-definida, determinística, que não envolve predição no mesmo sentido que a tarefa de classificação (FREITAS, 2000).

Outra distinção, facilmente identificada, diz respeito à questão sintática: regras de classificação têm

```
logica_aprovado calculo_aprovado introducao_inf_aprovado ingles_aprovado
logica_aprovado calculo_aprovado introducao_inf_aprovado ingles_aprovado
logica_reprovado calculo_reprovado calculo_aprovado introducao_inf_aprovado ingles_reprovado
logica_aprovado calculo_aprovado introducao_inf_aprovado ingles_aprovado
logica_aprovado calculo_aprovado introducao_inf_aprovado ingles_aprovado
logica_reprovado logica_aprovado calculo_reprovado introducao_inf_aprovado ingles_reprovado
logica_reprovado calculo_reprovado introducao_inf_aprovado ingles_reprovado
logica_aprovado calculo_aprovado introducao_inf_aprovado ingles_aprovado
logica_aprovado calculo_reprovado introducao_inf_aprovado ingles_aprovado
logica_aprovado calculo_aprovado introducao_inf_aprovado ingles_aprovado
```

Figura 3 - Relação das situação de aprovação das disciplinas segundo cada discente.

```
ingles_reprovado <- logica_reprovado  (30.0%, 100.0%)
calculo_reprovado <- logica_reprovado  (30.0%, 100.0%)
introducao_inf_aprovado <- logica_reprovado  (30.0%, 100.0%)
introducao_inf_aprovado <- ingles_reprovado  (30.0%, 100.0%)
introducao_inf_aprovado <- calculo_reprovado  (40.0%, 100.0%)
ingles_aprovado <- calculo_aprovado  (60.0%, 85.7%)
calculo_aprovado <- ingles_aprovado  (60.0%, 85.7%)
logica_aprovado <- calculo_aprovado  (60.0%, 85.7%)
calculo_reprovado <- logica_reprovado ingles_reprovado  (30.0%, 100.0%)
ingles_reprovado <- logica_reprovado calculo_reprovado  (30.0%, 100.0%)
logica_reprovado <- ingles_reprovado calculo_reprovado  (30.0%, 100.0%)
calculo_reprovado <- logica_reprovado ingles_reprovado calculo_aprovado  (10.0%, 100.0%)
ingles_reprovado <- logica_reprovado calculo_reprovado calculo_aprovado  (10.0%, 100.0%)
logica_reprovado <- ingles_reprovado calculo_reprovado calculo_aprovado  (10.0%, 100.0%)
logica_reprovado <- calculo_aprovado ingles_reprovado  (10.0%, 100.0%)
```

Figura 4 - Algumas regras de associação descobertas a partir de um conjuto de dados.

apenas um atributo em seu consequente, enquanto regras de associação podem ter mais de um item no seu consequente. Adicionalmente, a classificação é dita assimétrica com relação aos atributos a serem minerados, uma vez que os atributos previsores podem ocorrer apenas no antecedente da regra e o atributo meta pode ocorrer apenas no consequente da regra. Em contraste, a tarefa de associação pode ser considerada como simétrica com relação aos itens a serem minerados, uma vez que cada item pode ocorrer ou no antecedente ou no consequente da regra.

Agrupamento

A tarefa de agrupamento, às vezes chamada de classificação não-supervisionada [20], consiste na identificação de um conjunto finito de classes ou *clusters*, baseada nos atributos de objetos não previamente classificados. Um *cluster* é basicamente um conjunto de objetos agrupados em função de sua similaridade ou proximidade. Os objetos são agrupados de tal forma que as similaridades intraclusters (dentro de um mesmo *cluster*) sejam maximizadas e as similaridades interclusters (entre *clusters* diferentes) sejam minimizadas. A Figura 5 mostra um exemplo do resultado de uma tarefa de *clustering*, em que quatro *clusters* foram identificados.

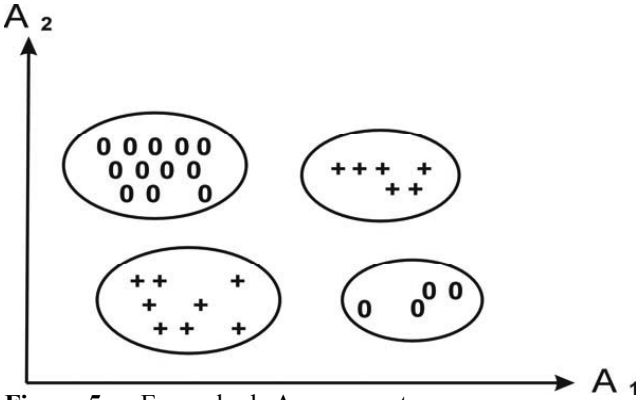

Figura 5 - Exemplo de Agrupamento.

Uma vez definidos os *clusters*, os objetos são identificados com seu *cluster* correspondente, e as características comuns dos objetos no *cluster* podem ser sumarizadas para formar a descrição da classe. Por exemplo, um conjunto de pacientes pode ser agrupado em várias classes (*clusters*), baseado nas similaridades dos seus sintomas, e os sintomas comuns aos pacientes de cada *cluster* podem ser usados para descrever a que *cluster* um novo paciente pertencerá. Assim, um dado paciente seria atribuído ao *cluster* cujos pacientes têm sintomas o mais parecido possível com os sintomas daquele dado paciente. Dessa forma, a tarefa de *clustering*, cujo resultado é a identificação de novas classes, pode ser realizada como pré-processamento para realização da tarefa de classificação (KUBAT et al., 1998).

3 Pós-processamento

Existem várias medidas propostas na literatura para pós-processar o conhecimento descoberto, porém este artigo prioriza a discussão sobre a avaliação do grau de interesse (*interestingness*) dos padrões descobertos. As diversas medidas descritas em geral são organizadas em dois grupos, ditas *user-driven* e *data-driven* (SILBERSCHATZ; TUZHILIN, 1996), (FREITAS, 1998).

A ideia básica das medidas *user-driven* é que o usuário especifica suas crenças, ou conhecimento prévio sobre o domínio da aplicação, e o sistema utiliza esse tipo de informação para selecionar regras interessantes. Uma regra é considerada interessante se ela representar alguma novidade com relação às crenças ou conhecimento prévio do usuário.

Em contrapartida, as medidas ditas *data-driven* tentam estimar o quanto as regras podem ser surpreendentes ao usuário de uma forma mais automática e indireta, sem exigir que esse especifique suas crenças ou conhecimento prévio.

Grande parte da literatura usa os termos "medidas subjetivas" e "objetivas", em vez de medidas *user-driven* e *data-driven*. Entretanto, neste artigo está se optando pela última terminologia, tendo em vista que o termo medida subjetiva pode não ser suficientemente claro. As crenças de um usuário certamente são subjetivas, porém elas constituem uma entrada para um método que computa medidas de interesse. Essas medidas tipicamente consistem em fórmulas matemáticas que irão atribuir um grau de interesse para essas regras. Esse grau é, em geral, computado de forma objetiva, dado o uso de uma fórmula matemática para computar o valor da medida. Sendo assim, o termo *user-driven* parece ser mais apropriado.

As medidas *user-driven* têm a vantagem de considerarem diretamente as crenças do usuário, porém têm as desvantagens de serem fortemente dependentes de conhecimento do domínio da aplicação e serem menos automáticas do que as medidas

data-driven, exigindo uma participação intensiva do usuário na tarefa de tornar explícitas as suas crenças ou avaliações. De fato, pode-se afirmar que estas medidas não são apenas dependentes do domínio de aplicação, mas também do usuário, uma vez que, mesmo considerando um mesmo domínio de aplicação, dois ou mais usuários podem ter crenças ou conhecimento do domínio bastante diversos.

As medidas *data-driven* têm a desvantagem de serem uma estimativa indireta do quão surpreendentes serão as regras para o usuário, ignorando suas crenças ou conhecimento prévio. Porém têm algumas vantagens, como por exemplo, maior independência do domínio da aplicação e serem mais automáticas, liberando o usuário da tarefa de explicitar as suas crenças ou conhecimento prévio, o que em geral consome muito tempo.

Desta forma, intuitivamente as medidas *user-driven* são mais indicadas quando um usuário específico está disponível, tem tempo e experiência suficientes para gerar uma especificação de boa qualidade de suas crenças e conhecimento prévio; enquanto as medidas *data-driven* são mais indicadas para situações nas quais existe um grande número de usuários ou mesmo quando o(s) usuário(s) não tiver(em) nem tempo, nem experiência suficientes.

Cabe ressaltar que os dois grupos de medidas não são mutuamente exclusivos, ou seja, é possível que sejam usadas medidas oriundas de ambos os grupos em uma determinada aplicação.

Este artigo instancia cinco medidas *data-driven* de interesse de regras que estão definidas pelas fórmulas (3)–(7), e maiores detalhes sobre essas e outras medidas podem ser encontrados nos trabalhos de Tan et al. (2002), (2004). As fórmulas são expressas usando a seguinte notação:

A denota o antecedente da regra;

C denota o consequente (classe) da regra;

P(A) denota a probabilidade de *A*, isto é, o número de exemplos satisfazendo o antecedente *A* dividido pelo número total de exemplos;

P(C) denota a probabilidade de *C*;

$\neg A$ e $\neg C$ denotam a negação lógica de *A* e *C*.

Quanto maior o valor da medida para uma dada regra, maior o grau de interesse estimado para aquela regra.

$$\phi\text{-Coeficiente} = \frac{P(AC)\ P(A)P(C)}{\sqrt{P(A)P(C)(1-P(A))(1-P(C))}} \quad (3)$$

$$\text{Piatetsky-Shapiro's} = P(AC) - P(A)\,P(C) \quad (4)$$

$$\text{Jaccard} = \frac{P(AC)}{P(A)+P(C)-P(AC)} \quad (5)$$

$$\text{Cosine} = \frac{P(AC).}{\sqrt{P(A)P(C)}} \quad (6)$$

$$\text{Interest} = \frac{P(AC)}{P(A)P(C)} \quad (7)$$

As Figuras (6) até a (10) apresentam as cinco regras de associação descobertas a partir do conjunto representado na Figura 3, com as maiores avaliações para cada uma das medidas instanciadas neste artigo.

logica_aprovado <- introducao_inf_aprovado (100.0%, 80.0%)	80,91
introducao_inf_aprovado <- logica_aprovado (80.0%, 100.0%)	80,91
ingles_aprovado <- logica_aprovado introducao_inf_aprovado (80.0%, 87.5%)	70.95
logica_aprovado <- ingles_aprovado introducao_inf_aprovado (70.0%, 100.0%)	70,95
ingles_aprovado <- logica_aprovado (80.0%, 87.5%)	70,95

Figura 6 - Avaliação das regras pela medida ϕ-coefficient.

logica_reprovado <- ingles_reprovado calculo_reprovado calculo_aprovado (10.0%, 100.0%)	-7.25
ingles_reprovado <- logica_reprovado calculo_reprovado logica_aprovado introducao_inf_aprovado (10.0%, 100.0%)	-7,25
ingles_reprovado <- logica_reprovado calculo_reprovado logica_aprovado (10.0%, 100.0%)	-7,25
logica_reprovado <- ingles_reprovado calculo_reprovado logica_aprovado (10.0%, 100.0%)	-7,25
logica_reprovado <- ingles_reprovado calculo_reprovado calculo_aprovado introducao_inf_aprovado (10.0%, 100.0%)	-7,25

Figura 7 - Avaliação das regras pela medida Piatetsky-Shapiro's.

introducao_inf_aprovado <- logica_aprovado (80.0%, 100.0%)	80,00
logica_aprovado <- introducao_inf_aprovado (100.0%, 80.0%)	80,00
logica_aprovado <- (100.0%, 80.0%)	80,00
introducao_inf_aprovado <- ingles_aprovado logica_aprovado (70.0%, 100.0%)	70.00
introducao_inf_aprovado <- ingles_aprovado (70.0%, 100.0%)	70,00

Figura 8 - Avaliação das regras pela medida Jaccard.

ingles_reprovado <- logica_reprovado calculo_reprovado introducao_inf_aprovado (30.0%, 100.0%)	1.00
ingles_reprovado <- logica_reprovado (30.0%, 100.0%)	1,00
logica_reprovado <- ingles_reprovado (30.0%, 100.0%)	1,00
logica_reprovado <- ingles_reprovado calculo_reprovado introducao_inf_aprovado (30.0%, 100.0%)	1,00
logica_reprovado <- ingles_reprovado introducao_inf_aprovado (30.0%, 100.0%)	1,00

Figura 9 - Avaliação das regras pela medida Cosine.

logica_reprovado <- ingles_reprovado calculo_reprovado calculo_aprovado introducao_inf_aprovado (10.0%, 100.0%)	0.03
ingles_reprovado <- logica_reprovado (30.0%, 100.0%)	0,03
logica_reprovado <- ingles_reprovado (30.0%, 100.0%)	0,03
ingles_reprovado <- logica_reprovado calculo_reprovado calculo_aprovado introducao_inf_aprovado (10.0%, 100.0%)	0,03
ingles_reprovado <- calculo_reprovado calculo_aprovado introducao_inf_aprovado (10.0%, 100.0%)	0,03

Figura 10 - Avaliação das regras pela medida Interest.

Hussain et al. (2000) apresentam um método que identifica, a partir de um conjunto de padrões descoberto, um subconjunto de regras que representam regras de exceção. A Tabela 1 mostra a estrutura geral das regras de exceção, considerando uma regra de "bom senso", ou "senso comum" (common sense), e uma regra de referência. Nesta tabela, A e B são conjuntos não-vazios de pares de atributo-valor, e C representa a classe predita pela regra. O símbolo "¬" denota a negação lógica. É importante observar que uma regra de exceção é uma especialização de uma regra de senso comum, e uma regra de exceção prediz uma classe distinta da classe prevista pela regra de senso comum. Este método assume que regras de senso comum representam padrões conhecidos pelo usuário, tendo em vista que aquelas regras têm uma grande cobertura, ao contrário das regras de exceção, que, em geral, são desconhecidas, uma vez que elas têm baixa cobertura. Sendo assim, as regras de exceção tendem a ser surpreendentes, dado o fato de representarem uma contradição em relação à regra de senso comum. É importante observar que a regra de referência auxilia na explicação da causa da regra de exceção.

$A \rightarrow C$ regra de senso comum (alta cobertura e alta precisão)
$A, B \rightarrow \neg C$ regra de exceção (baixa cobertura, alta precisão)
$B \rightarrow \neg C$ regra de referência (baixa cobertura e/ou baixa precisão)

Figura 11 - Estrutura das regras de exceção.

Pós-processando o conjunto de 95 regras descobertas a partir deste conjunto de dados (figura 3), é possível identificar alguns pares de conjuntos de regras gerais e suas respectivas regras de exceção. A partir da figura 12, é possível perceber que, em geral,

Regra Geral:
logica_aprovado <- calculo_aprovado introducao_inf_aprovado (70.0%, 85.7%)
Excecoes:
logica_reprovado <- calculo_aprovado introducao_inf_aprovado calculo_reprovado (10.0%, 100.0%)
logica_reprovado <- calculo_aprovado introducao_inf_aprovado ingles_reprovado (10.0%, 100.0%)

Figura 12 - Regras gerais e suas respectivas regras de exceção

os alunos aprovados em cálculo e introdução à informática também são aprovados em lógica. A exceção ocorre quando alunos que também foram aprovados em cálculo e introdução à informática, mas foram reprovados em cálculo ou inglês, revertem a situação de aprocação em lógica, passando a reprovados.

4 Conclusões

O processo KDD envolve uma série de etapas, desde a preparação dos dados e a extração dos padrões até a avaliação do quanto estes padrões descobertos agregam valor ao que o gestor já conhecia sobre o problema em questão. Este artigo, além de apresentar estas etapas, também detalhou o comportamento de alguns algoritmos/processos para as etapas de mineração de dados e pós-processamento sobre conjuntos de dados de pequena magnitude, permitindo, assim, que o leitor entenda o seu funcionamento. Para a etapa de mineração de dados foram simulados algoritmos para a descoberta de padrões na forma de regras de associação, classificadores e agrupamento. Para a etapa de pós-processamento, foram simulados processos de atribuição de grau de interesse para as regras descobertas, bem como identificados pares de regras (regra geral e suas respectivas regras de exceção) sobre o conjunto total descoberto.

Como trabalho futuro, poderiam ser incluídos neste relato processos/algoritmos para exemplificar a etapa de pré-processamento, incorporar outras formas de pós-processamento.

REFERÊNCIAS

ADRIAANS, P.; ZABTINGE, D. **Data mining.** England, Addison Wesley Longman. 1996.

AGRAWAL, R.; IMIELINSKI, T.; SWAMI, A. Mining associations between sets of items in massive databases. In: *ACM-SIGMOD, 1993.* **Proceedings…** *Int'l Conference on Management of Data,* Washington D.C., May 1993, p.207-216.

BREIMAN, L. et al. **Classification and regression trees.** Wadsworth and Brooks, Monterey, Ca. 1984.

FAYYAD, U. et al. **Advances in knowledge discovery and data mining**. American Association for Artificial Intelligence. Menlo Park, CA: MIT Press. 1996.

FISHER, D.; HAPANYENGWI, G. Database management and analysis tools of machine induction, **Journal of Intelligence Information Systems**, **2.** Boston, Kluwer Academic Publishers,. 1993, p. 5-38

FREITAS, A.A.; LAVINGTON, S.H. **Mining very large databases with parallel processing,** MA: Kluwer Academic Publishers. 1998.

_____. On objective measures of rule surprisingness. Principles of Data Mining & Knowledge Discovery (Proc. 2nd European Symp., PKDD'98. Nantes, France, Sep. 1998). **LNAI** v.1510, 1-9, Springer-Verlag, 1998.

_____. Understanding the crucial differences between classification and discovery of association rules - a position paper. **ACM SIGKDD Explorations**, v.2, n.1, 2000. p.65-69.

FU, Y. **Discovery of multiple-level rules from large databases,** Ph.D. Thesis of Doctor of Philosophy. Faculty of Applied Sciences, Simon Fraser University, British Columbia, Canada. 1996, 184p.

HAND, D.J. **Construction and assessment of classification rules**. New York: John Wiley & Sons. 1997.

_____. Introduction. In: Berthold, M.; Hand, D.J. (Eds.) **Intelligent Data Analysis**. Berkeley, CA: Springer- Verlag. 1999, p.1-14.

HUSSAIN, F.; LIU, H.; LU, H. Exception rule mining with a relative interestingness measure. PAKDD-2000, **LNAI** v.1805, 2000. p. 86-96.

KLOSGEN, W. **Patterns for knowledge discovery in databases**. Proc. Of Machine Learning. UK. 1992, p. 1-9.

KUBAT, M.; BRATKO, I.; MICHALSKI, R.S. A review of machine learning methods. In: Michalski, R.S., Bratko, I. and Kubat, M. (Eds.) **Review of machine learning methods, machine learning and data mining:** methods and applications. London, John Wiley & Sons, 1998, p.3-70.

SILBERSCHATZ, A.; TUZHILIN, A. What makes patterns interesting in knowledge discovery systems. **IEEE Trans. Knowledge & Data Eng.,** v.8, n.6, 1996.

SRIKANT, R.; AGRAWAL, R. Mining Generalized Association Rules. In: INT. CONF. VERY LARGE DATABASES, 21. **Proceedings…** 1995, p.407-419.

TAN P.; KUMAR, V.; SRIVASTAVA, J. Selecting the right interestingness measure for association patterns. In: ACM SIGKDD INT'L CONF. ON KNOWLEDGE DISCOVERY AND DATA MINING (KDD-2002), 8. **Proceedings…** p.32-41, 2002

_____. Selecting the right objective measure for association analysis. **Information System,** v.29, n.4, p. 293-313, 2004.