

Meta-DM: Uma ontologia para o domínio de mineração de dados

Edmar Augusto Yokome e Flávia Linhalis Arantes

Tema – Ontologias e Mineração de Dados

Resumo – Ontologias vêm sendo utilizadas em várias áreas de pesquisa da computação, incluindo a mineração de dados. Este artigo descreve o desenvolvimento de uma ontologia de domínio para a área de mineração de dados. A ontologia Meta-DM fornece uma terminologia comum que pode ser compartilhada e processada por ferramentas de mineração de dados. A ontologia também indica as fases do KDD que requerem conhecimento humano, sendo esse seu principal diferencial. Com ontologias como a Meta-DM é possível adicionar semântica no processo de mineração de dados para melhorar a interatividade e a cooperação entre especialistas e sistemas de mineração.

Palavras-chaves – ontologia de domínio, mineração de dados, descoberta de conhecimento em bases de dados (KDD).

I. INTRODUÇÃO

Em inteligência artificial, uma ontologia pode ser definida como “uma especificação explícita e formal de uma conceitualização compartilhada de um domínio de interesse” [1]. As ontologias são comumente utilizadas como estruturas que representam o conhecimento sobre uma determinada área, ou domínio, por meio de conceitos relevantes e relacionamentos entre eles. Com as ontologias, a representação do conhecimento sobre determinado domínio pode ser mais facilmente compartilhada e compreendida entre agentes humanos e de software.

No cenário atual de sistemas de informação, as ontologias vem sendo utilizadas em vários domínios, como web semântica, bancos de dados, sistemas especialistas, dentre outros. Na área de mineração de dados, o uso das ontologias em projetos que envolvem a descoberta de conhecimento em bases de dados permanece pouco explorado na literatura [2].

O processo de descoberta de conhecimento em bases de dados (KDD, do inglês, *Knowledge Discovery in Databases*) consiste em descobrir informações interessantes em bases de dados, informações essas que podem ser utilizadas para melhorar ou solucionar determinado problema, como por exemplo, melhorar a campanha de *marketing* de um supermercado, ajudar a decidir o melhor momento para vender ou comprar ações da bolsa de valores, etc. Sua utilização é justificada pelo grande crescimento das bases de dados nos últimos anos e também pelo o fato das base de dados estarem cada vez mais heterogêneas (banco de dados relacional, série temporal,

multimídia, imagens, dentre outros) tornando-se inviável fazer uma análise manual.

O processo de KDD é realizado em diferentes fases: limpeza, integração, seleção, transformação, mineração, avaliação e apresentação. As ferramentas de mineração de dados, tais como a Weka [3], são puramente orientadas aos dados, isto é, trabalham os dados isoladamente em cada fase do KDD, sendo que o processo de descoberta do conhecimento em bases de dados é guiado por um especialista humano.

O processo de KDD pode ser beneficiado com abordagens orientadas a ontologias, as quais podem inserir semântica no processo de mineração com o objetivo de auxiliar ou guiar o minerador de dados durante o processo de descoberta de conhecimento em base de dados. As ferramentas de mineração de dados, quando combinadas com o conhecimento do domínio do problema representado em ontologias, podem auxiliar o minerador de dados em tarefas como entendimento e preparação dos dados, seleção de dados relevantes para serem minerados, especificação de restrições para guiar a escolha e execução de algoritmos de mineração, dentre outras [4].

Neste trabalho, é apresentado o desenvolvimento de uma ontologia de domínio para a área de mineração de dados. A ontologia Meta-DM oferece uma terminologia comum para mineração de dados, podendo ser utilizada com diversas ferramentas relacionadas a esse domínio e assim ajudar o minerador de dados durante o processo de descoberta de conhecimento em bases de dados.

Para que seja possível auxiliar efetivamente o minerador de dados durante o processo de KDD, as ferramentas de mineração de dados precisam incluir conhecimento humano e do domínio do problema, de maneira que a descoberta de conhecimento em bases de dados aconteça de modo interativo com o especialista humano [5], [6]. Por essa razão, a ontologia Meta-DM procura identificar conceitos e relacionamentos onde o conhecimento humano, utilizado para o entendimento do problema, se faz necessário durante o processo de KDD.

O minerador de dados ao utilizar uma ontologia de domínio como a Meta-DM para guiar o processo de KDD terá um conjunto de itens semânticos que permite auxiliá-lo nas diversas etapas deste processo, cujo objetivo é obter resultados mais expressivos.

Este artigo está organizado da seguinte maneira: na Seção II é apresentada a justificativa da escolha desse tema. Na Seção III é apresentado o problema que motivou o desenvolvimento desse trabalho, bem como a solução adotada. Na Seção IV é apresentada uma breve fundamentação teórica com conceitos considerados relevantes sobre mineração de dados e ontologias para o desenvolvimento desse trabalho. Na Seção V são

apresentados alguns trabalhos relacionados ao tema e feito um comparativo entre as propostas de desenvolvimento com a ontologia Meta-DM. Na seção VI a metodologia utilizada para o desenvolvimento da ontologia é apresentada. Na Seção VII é apresentado o processo de desenvolvimento da ontologia, com o detalhamento das tarefas executadas em cada fase do seu ciclo de vida. A Seção VIII apresenta algumas direções de trabalhos futuros para nossas pesquisas. Por fim, a Seção IX apresenta as conclusões deste artigo e a ontologia Meta-DM como principal resultado de nossas pesquisas até o momento.

II. JUSTIFICATIVA

Segundo Cao e Zhang [6], a mineração de dados praticada atualmente é um processo automatizado que fornece algoritmos e ferramentas, com pouco envolvimento humano e sem a capacidade para adaptar o processo às restrições do ambiente externo (domínio do contexto). A consequência disso é que os resultados da mineração de dados muitas das vezes não são interessantes aos objetivos do negócio para o qual o projeto de mineração de dados foi desenvolvido. Cao e Zhang propuseram uma metodologia chamada D³M, que considera conhecimento humano e informação de contexto sobre o problema em questão durante o processo de mineração de dados. A metodologia D³M possui as seguintes características:

- Restrição baseada em contexto: conhecer profundamente o ambiente em torno do domínio do problema, seus dados e objetivo. Possíveis maneiras de ter restrições baseadas em contexto são por meio da utilização de metadados de domínio ou ontologias relacionadas ao problema de mineração.
- Integração de conhecimento de domínio: diz respeito à maneira como o conhecimento do domínio do problema pode ser representado e integrado no processo de descoberta de conhecimento em bases de dados. Utilizar ontologias relacionadas ao domínio do problema é uma das abordagens adequadas para modelar e integrar conhecimento de domínio no processo de mineração de dados.
- Cooperação entre homem e máquina: fazer uma mineração cooperativa entre especialistas e sistemas de mineração em todo o processo.
- Mineração em profundidade: consiste em avaliar e refinar regras acionáveis. Antes do início da mineração de dados, o minerador pode definir regras que vão de encontro aos interesses do negócio. Essas regras, chamadas de “acionáveis”, podem ser disparadas durante o processo de mineração de dados.
- Melhorar a “acionabilidade” do conhecimento: padrões genéricos podem precisar de melhorias para gerar padrões acionáveis, isto é, padrões que vão de encontro aos interesses do problema de mineração.
- Processo de refinamento iterativo dos resultados: a avaliação e refinamento de resultados são baseados em *feedback* iterativo até chegar à fase final.
- Suporte de mineração interativa e paralela: consiste em obter pedidos dos usuários, gerenciar informações e usar algoritmos para processá-los em máquinas distintas.

A justificativa deste trabalho vai de encontro à atual tendência na área de mineração de dados, que é a inserção

de conhecimento humano e de domínio durante o processo de descoberta de conhecimento em bases de dados. Esperamos contribuir com o estado da arte por meio do desenvolvimento de uma ontologia para o domínio de mineração de dados, de maneira a inserir conhecimento humano e de contexto para que ferramentas de mineração possam auxiliar interativamente o minerador de dados durante o processo de descoberta de conhecimento em bases de dados.

III. PROBLEMA E SOLUÇÃO PROPOSTA

A mineração de dados praticada atualmente é altamente orientada aos dados, isto é, existe pouca interação com o minerador de dados durante o processo de descoberta de conhecimento em bases de dados [6]. Por esse motivo, muitas vezes os resultados gerados não são interessantes para o problema em foco (objetivos do negócio). Isto é consequência da falta de semântica e de maior interação entre o minerador e ferramentas de mineração de dados durante o processo de KDD. Conforme explicado na seção anterior, metodologias mais atuais, como D³M, surgiram para contornar esse problema e inserir conhecimento humano e de domínio durante o processo de mineração.

O problema a ser abordado em nossas pesquisas é como inserir conhecimento humano e de contexto, relacionados ao problema de mineração em questão, em ferramentas de mineração de dados para que as mesmas possam trabalhar interativamente com o minerador durante o processo de KDD. Um passo na direção de contribuir com a solução desse problema é o desenvolvimento de uma ontologia para o domínio de mineração de dados. Neste trabalho propomos a Meta-DM, uma ontologia para ferramentas de mineração de dados, com o intuito de identificar pontos onde o conhecimento humano se faz necessário durante o processo de KDD e, conseqüentemente, permitir a inserção de semântica de domínio durante esse processo.

Há alguns trabalhos na literatura relacionados ao desenvolvimento de ontologias para o domínio de mineração de dados, como apresentado em V. No entanto, nenhum deles trata em específico o desenvolvimento de ontologias para ferramentas de mineração de dados, identificando pontos onde o conhecimento humano é necessário, para que as ferramentas de mineração possam trabalhar colaborativa e interativamente com o minerador durante o processo de descoberta de conhecimento em bases de dados.

IV. FUNDAMENTAÇÃO TEÓRICA

A fundamentação teórica desse trabalho envolve duas áreas da ciência da computação, que são ontologias e descoberta de conhecimento em base de dados. As subseções a seguir fazem uma breve descrição de cada uma delas.

A. Ontologias

Uma ontologia é uma maneira de se conceituar de forma explícita e formal os conceitos e restrições relacionadas a um domínio de interesse [7]. Elas são utilizadas para representar conhecimento sobre uma determinada área ou domínio, por meio de conceitos e relacionamentos.

Para disciplinar e apoiar o processo de criação de ontologias são utilizadas metodologias, tais como METHONTOLOGY [8], Noy e McGuiness [9], Grüninger e Fox [10], Uschold e King [11], dentre outras. Para o desenvolvimento da ontologia apresentada neste artigo foi utilizada a metodologia METHONTOLOGY [8], conforme apresentado na seção VI. Essa metodologia baseia-se na construção de ontologias a partir do zero, podendo utilizar outras ontologias ou não. A escolha dessa metodologia se deu pelo fato de suas fases serem bem delimitadas e documentadas, o que a torna didática para quem está desenvolvendo ontologias pela primeira vez.

Para formalizar ontologias diversas linguagens podem ser utilizadas, como OWL (Web Ontology Language) [12], [13], Ontolingua [14], LOOM [15], F-Logic [16], etc. Neste trabalho, a linguagem da web semântica OWL foi utilizada. Essa linguagem é usada em conjunto com RDF e RDF-S, que também são recomendações W3C para a web semântica. A OWL é utilizada para representar explicitamente o significado dos termos em vocabulários e as relações entre estes termos [13], a RDF é uma linguagem para representar informação sobre recursos [17] e a RDF Schema é uma extensão semântica do RDF, que fornece mecanismos para descrever grupos de recursos e as relações entre esses recursos [18]. Dentre os motivos levados em consideração para a escolha de OWL podemos dizer que OWL já é uma recomendação W3C desde 2004 e é considerada um padrão de fato para o desenvolvimento de ontologias, sua utilização pode facilitar a integração da Meta-DM com aplicações desenvolvidas no contexto da web semântica.

Ferramentas para desenvolvimento de ontologias são bastante úteis, pois aumentam a produtividade e fornecem vários recursos que facilitam o processo de desenvolvimento de ontologias. Algumas ferramentas disponíveis gratuitamente são: Protégé [19], Servidor Ontolingua [14], KAON [20], SWOOP [21], dentre outras. A ferramenta escolhida para o desenvolvimento da ontologia Meta-DM foi a Protégé [19]. Entre as vantagens dessa ferramenta podemos citar: bom suporte a OWL, boa documentação (inclusive com tutoriais), seu código é aberto, possui uma comunidade ativa de desenvolvedores e usuários, o que a torna uma ferramenta amplamente utilizada para o desenvolvimento de ontologias por diversos grupos de pesquisa. Além disso, Protégé oferece uma grande gama de *plugins* que podem ser adicionados conforme a necessidade da ontologia e da aplicação. Entre os *plugins* que foram utilizados durante o desenvolvimento da ontologia Meta-DM estão: OWLViz [22] para acompanhar graficamente o desenvolvimento da ontologia e a máquina de inferência Pellet [23] para verificar a consistência entre as classes declaradas na ontologia.

B. Descoberta de conhecimento em base de dados

O processo de descoberta de conhecimento em base de dados (KDD, do inglês, *Knowledge Discovery in Databases*) é uma ramificação da ciência da computação que tem como objetivo encontrar padrões interessantes em base de dados. O processo de KDD, ilustrado na Figura 1 é constituído das seguintes fases: Limpeza e Integração, Seleção

e Transformação, Mineração, Avaliação e Apresentação e Conhecimento. A mineração de dados é uma das fases do KDD, que consiste na extração ou mineração do conhecimento em um grande amontoado de dados [24].

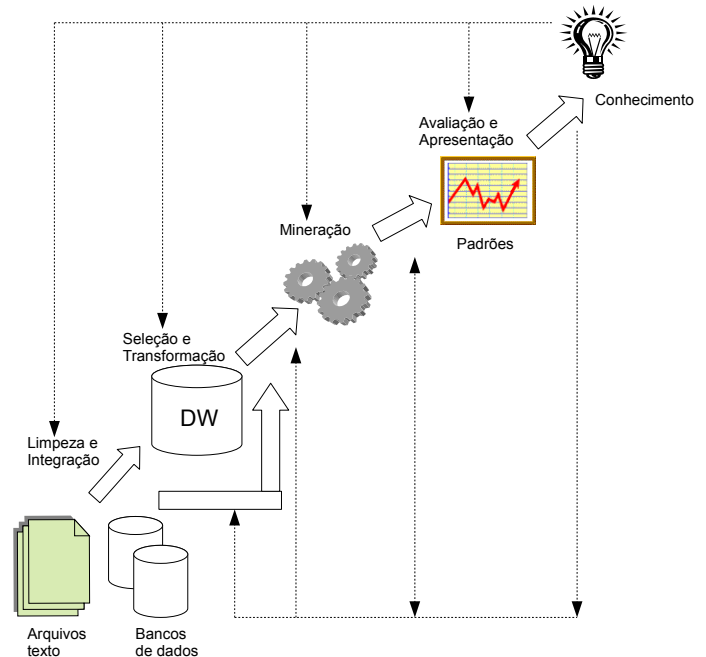


Fig. 1. Fases do processo de descoberta de conhecimento em bases de dados (KDD) [24].

Para o desenvolvimento de projetos de mineração de dados há metodologias que disciplinam e auxiliam seus projetistas, entre elas, a CRISP-DM e a D^3M foram utilizadas neste trabalho.

CRISP-DM é uma metodologia orientada aos dados e contempla as fases do KDD [25]. Ela é composta por seis etapas: compreensão do negócio, compreensão dos dados, preparação dos dados, modelagem, avaliação e aplicação. Esta metodologia foi importante para a definição dos conceitos utilizados na ontologia Meta-DM.

A metodologia D^3M foi desenvolvida com o objetivo de permitir uma mineração de dados mais interativa, levando em consideração o conhecimento do domínio do problema durante a realização de um projeto de mineração de dados [6]. Esta metodologia foi importante para identificar os pontos onde o conhecimento humano é necessário para o entendimento do problema durante o processo de KDD.

A utilização de ferramentas de mineração de dados em um projeto de KDD permite alcançar resultados mais eficientes, pois devido ao crescimento e à diversificação dos tipos de bases de dados, é praticamente inviável para o ser humano realizar a mineração de dados sem uma ferramenta adequada. Como exemplos de ferramentas estão: Kira [26], Weka [3], Tanagra [27], Oracle Data Mining [28], dentre outras. As ferramentas Weka e Kira foram estudadas com o intuito de conhecer a terminologia utilizada em ferramentas

de mineração de dados. A primeira delas pelo fato de ser amplamente utilizada em projetos de mineração, e a segunda por ser resultado de um trabalho de pesquisa que visa guiar o minerador durante a mineração de dados (conforme explicado em VIII).

V. TRABALHOS RELACIONADOS

Para o desenvolvimento da ontologia Meta-DM foram pesquisados outros trabalhos existentes na literatura que também desenvolveram ontologias para o domínio de mineração de dados.

Estes trabalhos serviram em um primeiro momento para ter uma ideia do que poderia ser realizado, depois que contribuições extras poderíamos dar em relação a ontologia Meta-DM e por fim quais trabalhos poderiam ser reutilizados.

Nesta seção, apresentamos os principais trabalhos relacionados comparados com a Meta-DM.

A ontologia de Sharma e Osei-Bryson foi desenvolvida para representar a fase de *business understanding* da metodologia CRISP-DM [25] para mineração de dados. Os autores fizeram o levantamento de questões relacionadas a essa etapa da mineração de dados com o objetivo de ajudar o minerador no entendimento do negócio [29]. Diferente da ontologia de Sharma e Osei-Bryson, a Meta-DM tem como objetivo representar todas as etapas de um projeto de mineração de dados.

A DM Ontology [30] é uma ontologia que contempla a mineração de dados aplicada ao negócio de *marketing*, voltado ao financiamento. Isto é, essa ontologia foi desenvolvida considerando um problema que a mineração de dados vai ajudar a resolver. O propósito da ontologia Meta-DM é contemplar as etapas da mineração de dados com um todo, sem se voltar a um problema de mineração particular. Entretanto, outras ontologias podem ser utilizadas juntamente com a Meta-DM para prover conhecimento de domínio necessário para resolver determinado problema de mineração de dados.

A ontologia DMO [31] foi desenvolvida com o intuito de guiar um projeto de mineração de dados em grade. Seu maior objetivo é realizar a mineração de dados utilizando serviços semânticos espalhados pela web. Para isso, utiliza a ontologia OWL-S para descrever serviços web semânticos. A ontologia utiliza os conceitos das fases do KDD, entretanto tem objetivos diferentes da ontologia Meta-DM.

A ontologia de Pinto e Santos [4] utiliza alguns conceitos da DMO [31] e foi desenvolvida com o intuito de contemplar exclusivamente as fases do KDD seguindo a metodologia METHONTOLOGY [8]. Alguns conceitos da ontologia de Pinto e Santos foram utilizados na Meta-DM. Esta ontologia foi a que mais assemelhou com a proposta da Meta-DM, fato que justifica uma maior atenção neste trabalho. Entretanto, a ontologia Meta-DM leva também em consideração a metodologia CRISP-DM [25] e tem o objetivo de ser uma ontologia para ferramentas de mineração de dados, identificando também os momentos onde o conhecimento humano relacionado ao problema de mineração se faz necessário, com o intuito de buscar mais interatividade e cooperação entre o minerador de dados e ferramentas de mineração. Esse fato

justifica a criação da ontologia Meta-DM ao invés de utilizar a ontologia de Pinto e Santos na íntegra.

A ontologia Exposé [32] foi criada para gravar experimentos de aprendizado de máquina e o fluxo de trabalho realizado durante o processo de descoberta de conhecimento em bases de dados, onde estas informações podem ser coletadas, compartilhadas e reutilizadas, utilizando um vocabulário comum, experimentos em mineração de dados e detalhes de uso de algoritmos e conjuntos de dados. A ontologia Exposé trabalha em rede e utiliza outras ontologias do domínio de mineração de dados. Seu objetivo é, portanto, gravar experimentos relacionados à mineração de dados e poder compartilhar esses experimentos com outras aplicações. A ontologia Meta-DM foi construída com o intuito de guiar, de maneira genérica, ferramentas de mineração de dados durante o processo de descoberta de conhecimento em bases de dados. Outras ontologias relacionadas ao domínio do problema de mineração podem ser utilizadas em conjunto com a Meta-DM com o intuito de fornecer informação de contexto, entretanto, a Meta-DM não tem como objetivo utilizar outras ontologias do domínio de mineração de dados em suas aplicações.

A ontologia OntoDM [33] tem como objetivo criar um conjunto de definições de termos para o domínio de mineração de dados, como por exemplo, tipo de dados, conjunto de dados, tarefas de mineração de dados, algoritmos de mineração de dados, dentre outros. Desse modo, projetos de desenvolvimento de ontologias para esse domínio podem utilizar suas definições, evitando assim ambiguidades na interpretação de alguma definição do domínio.

No ano 2009, a ontologia OntoDM [34] passou por um processo de atualização onde foram incluídas definições de entidades base de mineração de dados para permitir a definição de entidades mais complexas, como por exemplo, restringir-se a um contexto de domínio ou descrever diferentes aspectos de mineração de dados. Segundo Panov et al. [35], a ontologia é baseada em um *framework* que representa as entidades de um projeto de mineração de dados. A OntoDM tem o propósito de unificar o domínio da mineração de dados e de formalizar as definições e resultados produzidos a partir da mineração de dados. Essa ontologia é classificada como ontologia peso-pesado, cujo objetivo é representar todos os componentes que fazem parte de um projeto de mineração e assim oferecer uma terminologia comum para projetos de mineração de dados.

Diferentemente da OntoDM, a ontologia Meta-DM tem a característica de ser uma ontologia peso-leve abordando todo o processo de mineração de dados, sem se aprofundar em uma etapa específica. Como esse aprofundamento depende, em grande parte, do problema de mineração a ser abordado, a ontologia Meta-DM pode ser especializada ou utilizada em conjunto com uma ontologia do domínio do problema de mineração em questão, conforme descrito na seção VIII. O objetivo de servir como terminologia comum para o domínio de mineração de dados é contemplado por ambas ontologias, entretanto, a OntoDM não representa, formalmente, a necessidade de conhecimento humano no processo de KDD, sendo esse o maior diferencial da ontologia Meta-DM com relação a outras ontologias para mineração de dados encontradas na literatura. Isso possibilita que ferramentas de mineração adi-

cionem interfaces para facilitar a cooperação e interatividade com especialistas humanos.

VI. METODOLOGIA PARA DESENVOLVIMENTO DA META-DM

A metodologia METHONTOLOGY [8] foi utilizada para o desenvolvimento da Meta-DM. Essa metodologia baseia-se na construção de ontologias a partir do ponto zero, podendo utilizar outras ontologias ou não. Os autores fazem uma comparação do ciclo de vida de uma ontologia com o processo de desenvolvimento de um software tradicional e ressaltam que é bem complicado levantar todos os requisitos necessários antes de começar o processo de desenvolvimento. As fases do ciclo de vida de uma ontologia para esta metodologia são: especificação, aquisição do conhecimento, conceituação, integração, implementação, avaliação e documentação (ver Figura 2). A seguir, uma breve descrição de cada uma das fases é apresentada.

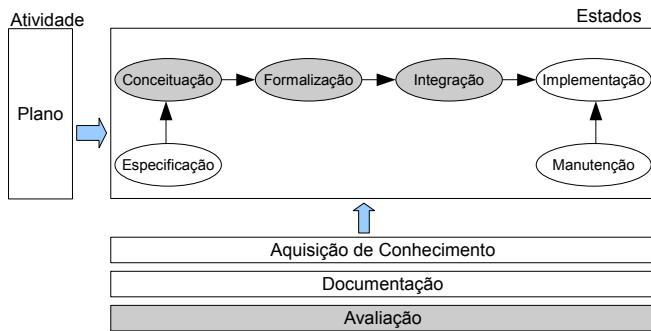


Fig. 2. Fases da metodologia METHONTOLOGY, que foi utilizada para o desenvolvimento da ontologia Meta-DM. As fases de aquisição de conhecimento, documentação e avaliação estão presentes durante todo o processo de desenvolvimento da ontologia [8].

- 1) **Especificação:** A meta desta fase é produzir um documento de especificação da ontologia, escrito em uma linguagem natural, usando um conjunto de representação intermediária ou usando questões de competência [10]. Nesta fase é proposto que no mínimo as seguintes informações sejam incluídas:
 - Propósito da ontologia: incluindo usuários, cenários de uso, usuários finais, etc.
 - Nível de formalidade da ontologia implementada, que pode ser altamente formal, semi-formal ou rigorosamente formal.
 - Escopo que a ontologia irá cobrir: inclui um conjunto de termos a ser representado, suas características e granulosidade.
- 2) **Aquisição de conhecimento:** Esta fase é realizada simultaneamente com a fase de especificação, e está relacionada a adquirir conhecimentos necessários para começar o processo de criação da ontologia. Para adquirir o conhecimento necessário são utilizados consultas a especialistas, livros, manuais, figuras, tabelas e mesmo outras ontologias como fonte de conhecimento,

e que pode ser elucidado com técnicas como: *brainstorming*, entrevistas, análise de texto formal e informal e ferramentas de aquisição de conhecimento.

- 3) **Conceituação:** Nesta fase será estruturado o conhecimento de domínio em um modelo conceitual que irá descrever o problema e suas soluções em termos do vocabulário de domínio identificado na atividade de especificação da ontologia. A primeira atividade a ser realizada é construir um completo Glossário de Termos (conceitos, instâncias, verbos e propriedades), que irá resumir tudo o que é útil e potencialmente utilizável no conhecimento de domínio e seu significado. Uma vez completado o glossário de termos, deve-se agrupar os termos em conceitos (dicionário de dados que descreve os conceitos, seus significados, atributos e instâncias) e verbos (ações no domínio). No final dessa fase será produzido um modelo conceitual expresso como um conjunto de conceitos bem definidos, que permitirá ao usuário final: verificar se a ontologia será ou não útil para uma da aplicação sem inspecionar seu código fonte; e comparar o escopo e plenitude de outras ontologias, sua reusabilidade e compatibilidade pela análise do conhecimento.
- 4) **Integração:** Para acelerar o processo de desenvolvimento de uma ontologia, pode-se considerar o reuso de definições já desenvolvidas, dentro de outras ontologias, ao invés de começar a construção do seu início.
- 5) **Implementação:** Consiste em implementar a ontologia em uma linguagem formal, como: OWL [13], Ontolingua [14], LOOM [15], F-Logic [16], dentre outras. Nesta fase é requerido um ambiente de desenvolvimento de ontologias e que deve pelo menos incluir: uma análise léxica e sintática, um tradutor, um editor, um navegador, realizar pesquisa de termos e apresentação dos resultados produzidos.
- 6) **Avaliação:** Realizar um julgamento técnico da ontologia, seu ambiente de software e documentação das fases do seu ciclo de vida. A avaliação inclui verificação e validação. Verificação refere-se ao processo técnico que garante a correção de uma ontologia e validação garante que a ontologia desenvolvida representa o domínio do conhecimento definido na fase de especificação.
- 7) **Documentação:** Para cada uma das fases anteriores é feito um documento descrevendo o que foi realizado. A documentação é parte integrante do desenvolvimento da ontologia. Assim esta etapa está presente em todas as anteriores.

O conjunto dessas etapas forma o ciclo de vida de desenvolvimento de uma ontologia. Esta metodologia foi adotada neste trabalho para registrar cada um dos passos do desenvolvimento da Meta-DM.

VII. DESENVOLVIMENTO DA ONTOLOGIA META-DM

A ontologia Meta-DM tem como objetivo guiar o processo KDD e identificar pontos onde o conhecimento humano relacionado ao problema de mineração se torna necessário. Com isso, pretende-se inserir semântica no processo de mineração

de dados e aumentar a participação do minerador de dados durante o processo de descoberta de conhecimento em bases de dados.

A construção da ontologia Meta-DM foi realizada a partir da metodologia CRISP-DM, onde a mesma foi utilizada para identificar as diversas etapas do processo de mineração de dados e assim representa-los na ontologia. Foram utilizadas ainda as ferramentas Weka e Kira com o objetivo de identificar as tarefas essenciais que deveriam fazer parte da ontologia Meta-DM.

A ontologia Meta-DM é apresentado na Figura 3 onde a mesma é composta por elipses (representado um determinado conceito), setas (representando um relacionamento), linhas (representando uma dependência), ligação com um círculo preenchido (representando um atributo) e uma seta tracejada (representando ponto onde o conhecimento humano se faz necessário).

A Figura 2 mostrou as fases do ciclo de vida de desenvolvimento da ontologia Meta-DM. As próximas subseções detalham como cada fase foi realizada, dentro da metodologia METHONTOLOGY.

A. Especificação

Nessa fase, o domínio e o escopo da ontologia foram determinados. Isto é, foi definido que a ontologia iria representar o domínio de mineração de dados, com o objetivo de guiar o processo de mineração de dados em ferramentas de mineração. Foi definido também que a metodologia CRISP-DM deveria ser levada em consideração com o intuito de levantar as etapas que constituem um projeto de mineração de dados.

Nesta fase, foi definido ainda que a ontologia deveria contemplar os momentos em que há a necessidade de conhecimento humano, relacionado ao problema de mineração, durante o processo de mineração de dados.

B. Aquisição de Conhecimento

Para desenvolver uma ontologia de domínio é preciso conhecer bem o domínio em questão, isto é, ser um especialista na área. Antes de começar o desenvolvimento da Meta-DM, foi necessário estudar mineração de dados. Para isso foram utilizados: livros, aulas, trabalhos práticos, artigos e a ajuda de especialistas da área.

Ter estudado a área de mineração de dados foi importante para saber quais conceitos se enquadravam com a proposta da ontologia e como eles se relacionavam.

C. Integração

Para o desenvolvimento da Meta-DM a reutilização de ontologias já existentes foi considerada. Foram analisadas várias ontologias, conforme apresentado nos trabalhos relacionados (seção V).

As classes *Algorithm* e *Data*, bem como suas subclasses, foram reutilizadas da ontologia de Pinto e Santos, como mostra a próxima seção.

D. Conceituação

Na fase de conceituação foram enumerados os termos importantes para a ontologia e definidas as classes e hierarquias entre elas. As classes e seus relacionamentos começaram a ser representados na ferramenta Protégé.

O diagrama da ontologia Meta-DM é apresentado na Figura 3. Entre os elementos que constituem a ontologia estão classes, relacionamentos e atributos. O desenvolvimento da ontologia foi bastante cíclico, novos conceitos e relacionamentos eram identificados conforme a ontologia foi evoluindo, até chegar ao resultado apresentado na Figura 3.

A ontologia representa também, por meio de uma linha tracejada, os relacionamentos onde existe a necessidade do conhecimento humano (ou intervenção de um especialista humano) para descoberta de conhecimento em bases de dados. Há portanto, dois tipos de relacionamento entre os conceitos da ontologia – os que envolvem conhecimento humano (linhas pontilhadas) e os que são automáticos no processo de KDD (linhas contínuas).

A ontologia pode ser dividida em cinco partes: Dados, Entendimento do Problema, Tratamento dos Dados, Tarefa de Mineração e Resultados. Os conceitos da ontologia envolvidos em cada uma dessas partes são descritos a seguir.

1) *Dados*: A classe *Data* é responsável por abrigar todos os dados e a estrutura referente a eles. Suas subclasses – *Value* e *Structure* – especificam valores e estrutura da base de dados.

2) *Entendimento do Problema*: O problema de mineração deve ser analisado e compreendido com relação aos dados (*Data Understanding*) e com relação ao negócio da mineração (*Business Understanding*). Este último é feito a partir da definição do objetivo (*Objective*) e da descrição do problema (*Problem*) que o projeto de mineração de dados deve resolver. Note que, nessa fase da mineração de dados, o conhecimento humano sobre o problema de mineração em questão se faz necessário, o que está representado na ontologia por linhas tracejadas.

3) *Tratamento dos Dados*: Essa parte da ontologia representa o tratamento da base de dados de uma forma que se adéque ao projeto de mineração de dados. A principal classe é *Table for Analysis*, que representa os dados prontos para passar pelo processo de mineração. Os relacionamentos *hasIntegration*, *hasTransformation*, *hasSelection* e *hasCleaning* representam ações que precisam ser executadas para o tratamento da base de dados, onde respectivamente dizem respeito a integração da base de dados, a transformação da base de dados, a seleção dos dados considerados relevantes para a mineração e a limpeza dos dados ruidosos ou com inconsistências. Em todas essas fases, o conhecimento de um especialista sobre o problema de mineração em questão é necessário.

4) *Tarefa de Mineração*: Representa a definição da tarefa de mineração de dados por meio da classe *Task* e suas subclasses. Juntamente com a tarefa de mineração de dados, é preciso definir a técnica que será utilizada (classe *Technic* e suas subclasses) e seu respectivo algoritmo (classe *Algorithm* e suas subclasses). Nessas fases, o conhecimento humano também se faz necessário.

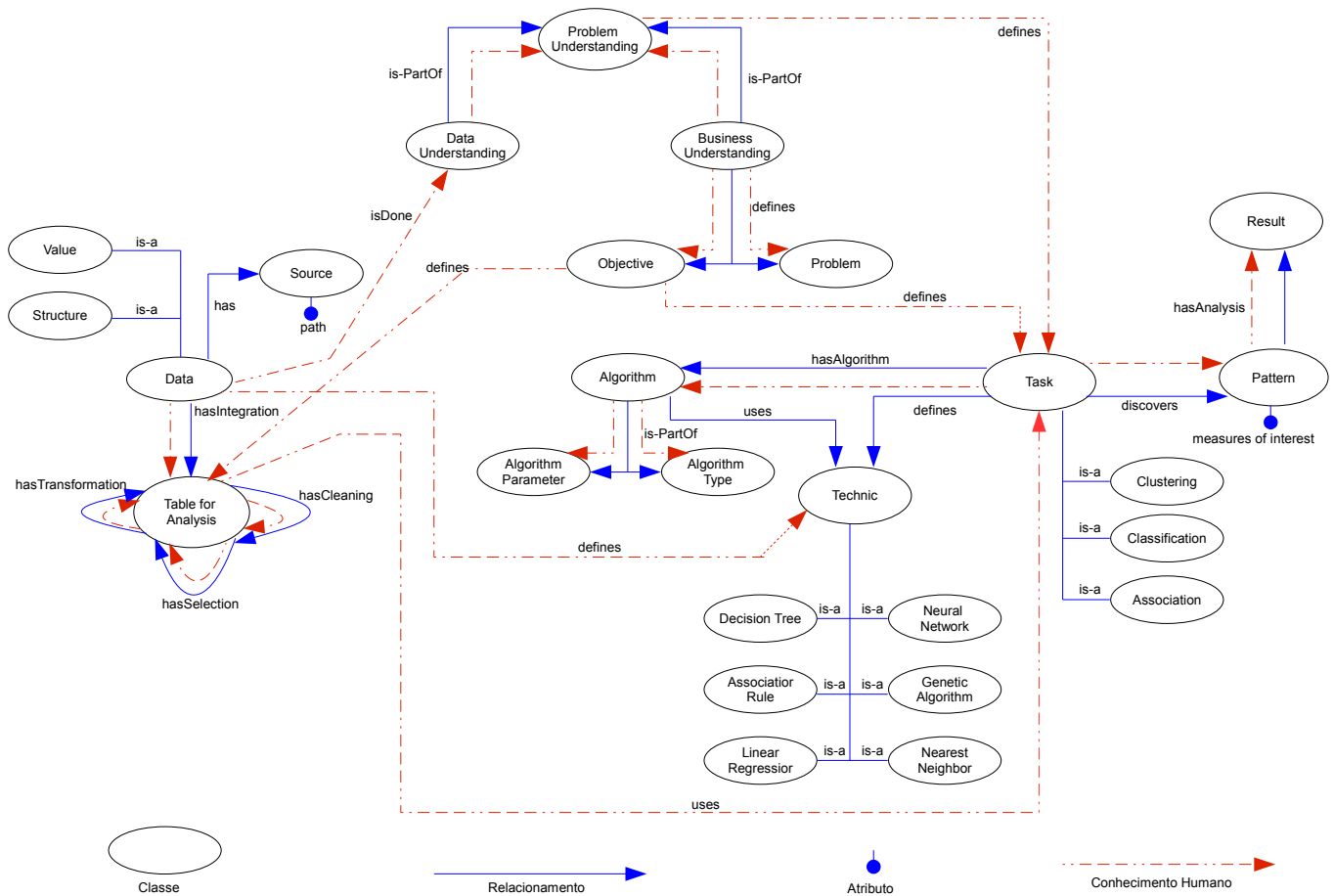


Fig. 3. Diagrama da Ontologia Meta-DM mostrando classes, relacionamentos, atributos e pontos onde o conhecimento humano é necessário durante o processo de KDD.

A tarefa de mineração de dados é pré-definida com o entendimento do negócio e entendimento dos dados. Nessa fase define-se qual tarefa melhor se enquadra ao objetivo do projeto de mineração de dados. Com a definição da tarefa é definida também uma ou mais técnicas conforme o tipo de dados que são utilizados por um determinado algoritmo de mineração.

5) *Resultados*: A última parte da ontologia representa os padrões encontrados após a execução do algoritmo de mineração de dados (classe *Pattern*) e os resultados obtidos (classe *Result*). O conhecimento humano é necessário para interpretar os resultados gerados e fazer uma descrição desses resultados para verificar se o projeto de mineração de dados atende ou não ao propósito que foi estabelecido.

E. Formalização/Implementação

A formalização foi feita na ferramenta Protégé, utilizando a linguagem OWL. O trecho de código apresentado nesta seção, serializado em Turtle [36], mostra a codificação da classe *Task*, suas subclasses, e os relacionamentos que saem da classe *Task* e chegam nas classes *Pattern*, *Technic* e *Algorithm*. A Figura 4 mostra a parte da ontologia que está sendo representada no trecho de código OWL.

As demais partes da ontologia Meta-DM (classes e relacionamentos) foram formalizados de maneira similar e podem ser encontrados na íntegra em [37].

```

01 :Task rdf:type owl:Class .
02
03 :Technic rdf:type owl:Class .
04
05 :Pattern rdf:type owl:Class .
06
07 :Algorithm rdf:type owl:Class .
08
09 :Association rdf:type owl:Class ;
10   rdfs:subClassOf :Task .
11
12 :Classification rdf:type owl:Class ;
13   rdfs:subClassOf :Task .
14
15 :Clustering rdf:type owl:Class ;
16   rdfs:subClassOf :Task .
17
18 :discovers rdf:type owl:ObjectProperty ;
19   rdfs:domain :Task ;
20   rdfs:range :Pattern ;
21   rdfs:subPropertyOf owl:topObjectProperty .
22
23 :discovers_human rdf:type owl:ObjectProperty ;
24   rdfs:domain :Task ;
25   rdfs:range :Pattern ;

```

```

26   rdfs:subPropertyOf owl:topObjectProperty .
27
28   :hasAlgorithm rdf:type owl:ObjectProperty ;
29     rdfs:domain :Task ;
30     rdfs:range :Algorithm ;
31     rdfs:subPropertyOf owl:topObjectProperty .
32
33   :hasAlgorithm_human rdf:type owl:ObjectProperty ;
34     rdfs:domain :Task ;
35     rdfs:range :Algorithm ;
36     rdfs:subPropertyOf owl:topObjectProperty .
37
38   :defines rdf:type owl:ObjectProperty ;
39     rdfs:domain :Task ;
40     rdfs:range :Technic ;
41     rdfs:subPropertyOf owl:topObjectProperty .

```

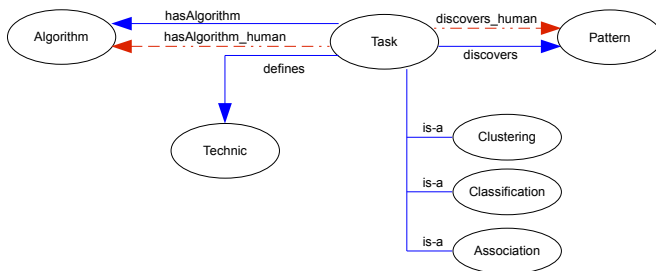


Fig. 4. Trecho da Ontologia Meta-DM mostrando a classe Task, suas subclasses e relacionamentos com as classes Pattern, Technic e Algorithm.

As linhas 01 até 07 mostram a criação das classes Task, Technic, Pattern e Algorithm. Essas classes são conceitos da ontologia que representam, respectivamente, a tarefa de mineração de dados a ser utilizada, a técnica de mineração de dados que será trabalhada, os padrões gerados e o algoritmo de mineração de dados a ser utilizado.

Nas linhas 09 até 16 são criadas as classes Association, Classification e Clustering como subclasses de Task. Essas subclasses têm a função de representar as tarefas base da mineração de dados.

Conforme já mencionado na seção VII-D, há dois tipos de relacionamento – os que envolvem conhecimento humano (linhas pontilhadas) e os que são automáticos no processo de KDD (linhas contínuas). Na Figura 3, esses relacionamentos aparecem com o mesmo nome para não sobrecarregar o diagrama. Mas na formalização da ontologia eles precisam ser diferenciados. Por exemplo, o relacionamento *discovers*, entre as classes Task e Pattern, foi formalizado em OWL como *discovers_human* e *discovers*, o que indica para as ferramentas de mineração que utilizarem a ontologia que o relacionamento *discovers_human* envolve conhecimento humano e possivelmente informação de contexto, enquanto o relacionamento *discovers* indica uma atividade que pode ser realizada automaticamente pela ferramenta de mineração, possivelmente com base em informações já obtidas em outras fases.

Nas linhas 18 até 21 o relacionamento *discovers*, citado no parágrafo anterior, é formalizado. Em OWL, os relacionamentos entre classes são chamados de Object Property e todos eles herdam as propriedades de *owl:topObjectProperty*, como indicado nas linhas

18 e 21, respectivamente. *rdfs:domain* e *rdfs:range* (linhas 19 e 20) indicam quais são as classes envolvidas em um relacionamento. No exemplo, o relacionamento *discovers* tem a classes Task como *domain* (domínio) e a classe Pattern como *range* (alcance), o que significa que “Task discovers Pattern” ou “a tarefa de mineração é usada para descobrir padrões”. De maneira similar, o relacionamento *discovers_human* é formalizado nas linhas 23 a 26.

Nas linhas 28 até 36 são criados os relacionamentos *hasAlgorithm* e *hasAlgorithm_human*, sendo que o domínio dessas propriedades é a classe Task e o alcance é a classe Algorithm. Semanticamente, podemos ler esse relacionamento como “a tarefa de mineração possui um algoritmo”.

Nas linhas 38 até 41 o relacionamento *defines* é formalizado entre as classes Task e Technic. Nesse relacionamento não houve a necessidade de representar a participação humana, visto que a técnica de mineração será definida (ou automaticamente sugerida pelo sistema) com base na tarefa de mineração e no algoritmo definidos em etapas anteriores.

Conforme já mencionado, a ontologia Meta-DM foi formalizada em OWL, que utiliza lógica de descrições (*Description Logics - DL*) para representar conhecimento [38].

Mais informações sobre a sintaxe e o formalismo de OWL podem ser encontradas em [12] e [13].

F. Avaliação

De acordo com a metodologia METHONTOLOGY [8], adotada para o desenvolvimento da Meta-DM, a avaliação da ontologia consiste em realizar um julgamento técnico da mesma, o que inclui validação e verificação.

A validação garante que a ontologia desenvolvida representa o domínio do conhecimento definido na fase de especificação. Para validar uma ontologia utiliza-se usualmente um sistema de software associado a ela. No caso da ontologia Meta-DM, o uso de um sistema de software em conjunto com a ontologia está previsto em nossos trabalhos futuros (conforme descrito na seção VIII). Portanto, apenas a parte de verificação foi realizada como avaliação da Meta-DM.

A verificação refere-se ao processo técnico que garante a correção de uma ontologia. Tal correção pode ser analisada com o uso de uma máquina de inferência como Pellet [23], *framework* Jena [39], FaCT++ [40], dentre outras. Para a verificação da ontologia Meta-DM, a máquina de inferência Pellet foi utilizada por ser uma ferramenta já consolidada e considerada madura para verificar inconsistências em ontologias. Além disso, Pellet pode ser instalada a partir de um *plugin* na ferramenta Protégé, é totalmente compatível com a linguagem OWL e com a linguagem para definição de regras SWRL (*Semantic Web Rule Language*) [41].

Com a utilização da Pellet, foi verificado se a ontologia possuía alguma inconsistência. As inconsistências podem estar relacionadas à disposição de classes (classes de mesma hierarquia e classes disjuntas), ao relacionamento entre classes (*range* e *domain*), ao tipo de atributo (literal, numérico e outros) e à aplicação de regras (consulta) na ontologia. O trecho de código a seguir mostra uma inconsistência com relação ao relacionamento e ao tipo de atributo.

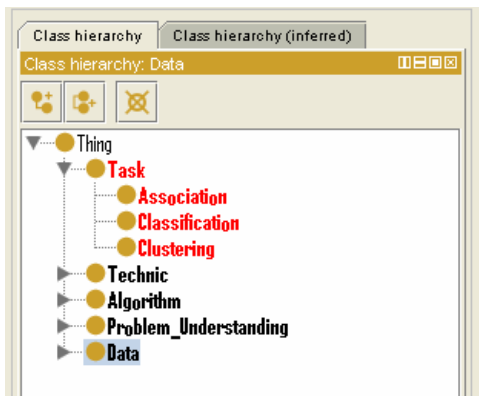


Fig. 5. Classes verificadas (em preto) e não verificadas (em vermelho) pela máquina de inferência Pellet, utilizada na ferramenta Protégé.

```

01 :Source rdf:type owl:Class .
02 :path rdf:type owl:DatatypeProperty ;
03 rdfs:domain :Source ;
04 rdfs:range xsd:decimal ;
05 rdfs:subPropertyOf owl:topDataProperty .
06 :path_instance rdf:type :Source ,
07 owl:NamedIndividual ;
08 :path "C:\dados\banco.dbc"^^xsd:decimal .

```

Na linha 01 a classe `Source` é criada. Na linha 02 a propriedade `path` é criada. Na linha 03 é definido que o *domain* da propriedade `path` é a classe `Source` (isto é, ela é uma propriedade da classe `Source`). Na linha 04 é definido que o *range* da propriedade `path` é o tipo decimal. As linhas 06 a 08 mostram uma instanciação da propriedade `path`, mas ao invés de ser armazenado um valor numérico foi armazenado um valor de *string*. Como resultado, a máquina de inferência Pellet acusou uma inconsistência. Ao mudar o valor de *string* para decimal essa inconsistência desapareceu.

A Figura 5 apresenta parte da implementação das classes da ontologia Meta-DM. As classes com tom mais escuro foram verificadas com sucesso. As classes em tom mais claro ainda não foram verificadas pela Pellet. Quando há alguma inconsistência, a máquina de inferência Pellet apresenta uma mensagem. Poucas inconsistências foram detectadas na Meta-DM e todas foram corrigidas com sucesso.

G. Documentação

A metodologia de desenvolvimento da Meta-DM prevê documentos formais para as várias fases do ciclo de vida da ontologia. Para a Meta-DM foram produzidos os seguintes documentos:

- Diagrama da ontologia, que é o mesmo apresentado na Figura 3;
- Dicionário das classes, relacionamentos e atributos, com uma descrição de cada elemento da ontologia;
- Descrição do que foi reutilizado na ontologia;
- O código OWL gerado com a formalização da ontologia;
- Documento de avaliação.

Todos esses documentos podem ser obtidos na íntegra em [37].

VIII. DIREÇÕES FUTURAS

As metodologias de mineração de dados fornecem poucos detalhes para o minerador de dados sobre como realmente realizar um passo no processo de descoberta de conhecimento em bases de dados [42]. Conforme afirmado por Vieira et al. [43], ferramentas para mineração de dados amplamente utilizadas na academia, como a Weka [3], se concentram principalmente no tratamento e visualização de dados e não incluem recursos instrucionais e interativos sobre as etapas do processo de KDD.

Com o objetivo de auxiliar o processo de mineração de dados uma ferramenta chamada Kira foi desenvolvida [43]. A ferramenta propõe um conjunto de guias para apoiar o usuário em cada fase da mineração de dados. Essas guias abstraem grande parte do conhecimento necessário pelo usuário para executar esse processo. Os módulos da Kira oferecem facilidades para ajudar o usuário a preparar os dados, executar algoritmos de mineração e avaliar os dados obtidos. No final de cada fase, o usuário é informado sobre o próximo passo do processo. O objetivo da Kira é orientar o usuário para executar os passos do KDD, mesmo que ela/ele não tenha conhecimento detalhado sobre o processo de mineração de dados [43]. A ferramenta foi utilizada em experimentos com os funcionários de uma empresa e em sala de aula por estudantes de graduação e pós-graduação.

Conforme já mencionado na Seção III, nossas pesquisas têm como objetivo investigar como inserir conhecimento humano e de domínio em ferramentas de mineração de dados para que as mesmas possam trabalhar interativamente com o minerador durante o processo de KDD. Apesar dos bons resultados da Kira para orientar os usuários no processo de mineração de dados, a ferramenta não considera a inteligência humana ou o conhecimento de domínio nesse processo. Nossos trabalhos futuros terão seu foco em adicionar a ontologia Meta-DM em uma arquitetura baseada em ontologias para a ferramenta de Kira. O objetivo é considerar o conhecimento de domínio (por meio de uma ontologia que representa o domínio do problema de mineração a ser tratado) e a cooperação humana no processo de mineração de dados e, conseqüentemente, (a) enriquecer as capacidades da Kira para orientar os usuários no processo de mineração de dados, e (b) considerar informações de contexto para (semi-)automatizar algumas tarefas que são inteiramente deixadas para o minerador de dados.

IX. CONCLUSÕES

Neste artigo, foi apresentado o desenvolvimento de uma ontologia de domínio para mineração de dados. O principal resultado apresentado neste artigo é a ontologia Meta-DM, sua conceituação e implementação. A ontologia Meta-DM tem o objetivo de fornecer uma terminologia comum, que pode ser compartilhada e compreendida por ferramentas de mineração de dados. Diferente de outras ontologias para o domínio de mineração de dados encontradas na literatura, a Meta-DM identifica e formaliza em quais fases da mineração de dados o conhecimento humano deve ser inserido durante o processo de KDD. Esse diferencial é importante para que o conhecimento humano sobre o domínio do problema possa ser

inserido em ferramentas de mineração e, conseqüentemente, possa ajudar ou guiar o minerador de dados durante o processo de descoberta de conhecimento em bases de dados.

Tendo em mãos a implementação da ontologia, nossos próximos passos irão focar na integração da ontologia Meta-DM com a ferramenta de mineração de dados Kira [26], com o intuito de melhorar a questão da participação humana na ferramenta. Para tando, pretende-se utilizar a Meta-DM para identificar e implementar na ferramenta Kira interfaces mais interativas com o minerador de dados.

REFERENCES

- [1] T. R. Gruber, "A translation approach to portable ontology specifications," *KNOWLEDGE ACQUISITION*, vol. 5, pp. 199–220, 1993.
- [2] C.-C. Shen and H.-M. Chuang, "A study on the applications of data mining techniques to enhance customer lifetime value," *WSEAS Trans. Info. Sci. and App.*, vol. 6, pp. 319–328, February 2009. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1512690.1512706>
- [3] H. WITTEN, I and E. FRANK, *Data Mining - Practical Machine Learning Tools and Techniques*, Elsevier, Ed. Second Edition, 2005.
- [4] F. M. Pinto and M. F. Santos, "Considering application domain ontologies for data mining," *WSEAS Trans. Info. Sci. and App.*, vol. 6, pp. 1478–1492, September 2009. [Online]. Available: <http://portal.acm.org/citation.cfm?id=1718151.1718156>
- [5] L. Cao, "Domain driven data mining (d3m)," *Data Mining Workshops, International Conference on*, vol. 0, pp. 74–76, 2008.
- [6] L. Cao and C. Zhang, "Domain-driven data mining: A practical methodology," *IJDWM*, vol. 2, no. 4, pp. 49–65, 2006.
- [7] N. Guarino, "Formal ontology and information systems," In: *Proceedings. Amsterdam: IOS on Formal Ontology and Information Systems (FOIS'98)*, pp. 3–15, 1998.
- [8] M. Fernandez-Lopez, A. Gomez-Perez, and N. Juristo, "Methontology: from ontological art towards ontological engineering," in *Proceedings of the AAAI97 Spring Symposium*, Stanford, USA, March 1997, pp. 33–40.
- [9] N. F. Noy and D. L. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology," Online, 2001. [Online]. Available: <http://www.ksl.stanford.edu/people/dlm/papers/ontology101/ontology101-noy-mcguinness.html>
- [10] M. Grüninger and M. Fox, "Methodology for the Design and Evaluation of Ontologies," in *IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing, April 13, 1995*, 1995. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.44.8723>
- [11] M. Uschold and M. King, "Towards a methodology for building ontologies," in *Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95*, Montreal, Canada, 1995.
- [12] W3C, "Owl 2 web ontology language – document overview," W3C Recommendation, October 2009, available at <http://www.w3.org/TR/owl2-overview/>.
- [13] D. L. McGuinness and F. van Harmelen, "Owl web ontology language overview," W3C Recommendation. Disponível em: <http://www.w3.org/TR/owl-features/>, Fevereiro 2004.
- [14] K. S. L. KSL, "Ontolingua," Disponível online <http://www.ksl.stanford.edu/software/ontolingua/>, 2005, stanford University. Acesso em: 12 jul. 2011.
- [15] Loom, "Loom project home page," Disponível online <http://www.isi.edu/isd/LOOM/>, 2007, university of Southern California – Information Sciences Institute. Acesso em: 12 jul. 2011.
- [16] M. Kifer, G. Lausen, and J. Wu, "Logical foundations of object-oriented and frame-based languages," *J. ACM*, vol. 42, pp. 741–843, July 1995. [Online]. Available: <http://doi.acm.org/10.1145/210332.210335>
- [17] E. MANOLA, F.; MILLER, "Rdf primer," *W3C Recommendation*, 2004.
- [18] R. V. BRICKLEY, D.; GUHA, "Rdf vocabulary description language 1.0: Rdf schema," *W3C Recommendation*, 2003.
- [19] Protégé, "Software protégé," <http://protege.stanford.edu/>, 2011, stanford University. Acesso em: 12 jul. 2011.
- [20] KAON, "Kaon tool suite," Disponível online: <http://kaon.semanticweb.org/>, 2011, acesso em: 12 jul. 2011.
- [21] SWOOP, "Swoop – semantic web ontology editor," Disponível online: <http://code.google.com/p/swoop/>, 2011, acesso em: 12 jul. 2011.
- [22] M. HORRIDGE, "Owlviz," <http://www.co-ode.org/downloads/owlviz/>, August 2009, the University Of Manchester.
- [23] clark and parseia, "Pellet: Owl 2 reasoner for java," disponível em: <http://clarkparsia.com/pellet/>, 2 2011, acesso em: 12 jul. 2011.
- [24] M. HAN, J.; KAMBER, *Data Mining: Concepts and Techniques*, E. Elsevier, Ed., 2006.
- [25] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth, "Cross industry standard process for data mining (crisp-dm 1.0) – step-by-step data mining guide," <http://www.crisp-dm.org/CRISPWP-0800.pdf>, 2000.
- [26] E. F. MENDES, "Kira: Uma ferramenta instrucional para apoiar a aplicação do processo de mineração de dados," Master's thesis, Faculdade de Ciências Exatas e da Natureza, da Universidade Metodista de Piracicaba – UNIMEP, 2009.
- [27] R. RAKOTOMALALA, "Tanagra: a free software for research and academic purposes," <http://eric.univ-lyon2.fr/~ricco/tanagra/en/tanagra.htm>, 2005.
- [28] Oracle, "Oracle data mining," Disponível em <http://www.oracle.com/us/products/database/options/data-mining/index.html>, 2011.
- [29] S. Sharma and K.-M. Osei-Bryson, "Organization-ontology based framework for implementing the business understanding phase of data mining projects," *Hawaii International Conference on System Sciences*, pp. 1–10, 2008.
- [30] L. Zheng and X. Li, "An ontology reasoning architecture for data mining knowledge management," *Wuhan University Journal of Natural Sciences*, vol. 13, pp. 396–400, 2008, 10.1007/s11859-008-0403-y. [Online]. Available: <http://dx.doi.org/10.1007/s11859-008-0403-y>
- [31] P. Brezany, I. Janciak, and A. M. Tjoa, "Ontology-based construction of grid data mining workflows," in *Data Mining with Ontologies: Implementations, Findings, and Frameworks*. Hershey, 2007. [Online]. Available: <http://eprints.cs.univie.ac.at/472/>
- [32] J. Vanschoren and L. Soldatova, "Exposé: An ontology for data mining experiments," in *International Workshop on Third Generation Data Mining: Towards Service-oriented Knowledge Discovery (SoKD-2010)*, Sep. 2010, pp. 31–46. [Online]. Available: <https://lirias.kuleuven.be/handle/123456789/273222>
- [33] P. Panov, S. Dzeroski, and L. Soldatova, "Ontodm: An ontology of data mining," *Data Mining Workshops, International Conference on*, vol. 0, pp. 752–760, 2008.
- [34] P. Panov, L. N. Soldatova, and S. Dzeroski, "Towards an ontology of data mining investigations," in *12th International Conference on Discovery Science (DS'09)*, ser. Lecture Notes in Computer Science, J. Gama, V. S. Costa, A. M. Jorge, and P. Brazdil, Eds., vol. 5808. Springer, 2009, pp. 257–271. [Online]. Available: <http://dblp.uni-trier.de/db/conf/dis/dis2009.html#PanovSD09>
- [35] P. Panov, S. Dzeroski, and L. N. Soldatova, *Inductive Databases and Constraint-Based Data Mining*. Springer, 2010, ch. Representing Entities in the OntoDM Data Mining Ontology, pp. 27–58.
- [36] D. Beckett and T. Berners-Lee, "Turtle – terse rdf triple language," W3C Team Submission. Disponível em: <http://www.w3.org/TeamSubmission/turtle/>, Janeiro 2008.
- [37] E. A. Yokome, "Uma ontologia para inserir conhecimento humano em ferramentas de mineração de dados," Master's thesis, Faculdade de Ciências Exatas e da Natureza, da Universidade Metodista de Piracicaba – UNIMEP, 2011.
- [38] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Prentice-Hall, Englewood Cliffs, NJ, 2003.
- [39] Jena, "Jena – a semantic web framework for java," Disponível online: <http://jena.sourceforge.net/>. Acesso em 26 de julho de 2011., 2011.
- [40] FaCT++, "Fact++ owl-dl reasoner," Disponível online: <http://owl.man.ac.uk/factplusplus/>. Acesso em 26 de julho de 2011., 2011.
- [41] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosf, and M. Dean, "Swrl: A semantic web rule language combining owl and ruleml," W3C Member Submission, May 2004, disponível em <http://www.w3.org/Submission/SWRL/>. Acesso em 26 de julho de 2011.
- [42] M. Charest, S. Delisle, O. Cervantes, and Y. Shen, "Intelligent data mining assistance via cbr and ontologies," in *Proceedings of the 17th International Conference on Database and Expert Systems Applications (DEXA'06)*, 2006, pp. 1–5.
- [43] M. T. P. Vieira, A. E. A. Silva, C. Peixoto, E. F. Mendes, and R. S. Gomide, "Kira – a tool based on guides and domain knowledge to instruct data mining," in *Proceedings of the IADIS International Conference Applied Computing*, vol. 2. Rome, Italy: IADIS, 2009, pp. 12–16.