

Proposta para um algoritmo de conversão de metadados em indivíduos de ontologias

Rodrigo Ribeiro
Universidade Federal do Rio Grande do Sul
9500 Av. Bento Gonçalves
Porto Alegre, Brasil
rzribeiro@inf.ufrgs.br

Tiago Primo
Universidade Federal do Rio Grande do Sul
9500 Av. Bento Gonçalves
Porto Alegre, Brasil
ttprimo@inf.ufrgs.br

ABSTRACT

Neste artigo será apresentada a proposta de um algoritmo para conversão de metadados de objetos de aprendizagem presentes em um repositório para os devidos fins, em indivíduos de ontologias para posterior armazenamento em um repositório baseado em triplas. Para apresentar a referida proposta, uma aplicação foi desenvolvida objetivando o acesso dos objetos de aprendizagem do repositório OBAA visando a posterior avaliação dos objetos de aprendizagem convertidos em indivíduos por mecanismos de inferência.

Keywords

Ontologias, Padrão de Metadados Educacionais, Web Semântica

Categories and Subject Descriptors

H.4 [Information Systems Applications]: Miscellaneous

1. INTRODUÇÃO

A Web Semântica potencializou o aparecimento de repositórios de informações com variadas finalidades. Dentre estas, destaca-se seu potencial para o uso no domínio educacional onde uma das suas possibilidades é relacionada ao compartilhamento de objetos de aprendizagem. A Web Semântica foi proposta por Berners Lee em [1] como uma mudança de paradigma da web atual, de uma internet voltada a documentos para uma internet voltada para informações, onde os computadores seriam capazes de cooperar com os usuários.

A proposta de Lee abriu caminho para técnicas que possibilitem interoperabilidade de objetos de aprendizagem por meio de metadados. Um objeto de aprendizagem é qualquer entidade, digital ou não, que é utilizada para fins educacionais, podendo ser usada, reusada ou referenciada durante algum processo educacional [5], levando isso em conta, os metadados tornaram-se uma boa opção para realizar esses objetivos [8].

Metadados são dados que descrevem outros dados, e servem como uma estrutura para a organização da Web Semântica, possibilitam uma interpretação semântica do documento pelo computador. Os Metadados de repositórios educacionais devem se preocupar em possibilitar que uma entidade tenha as características de um objeto de aprendizagem, ou seja, facilitar o uso, reuso e a referencia.

Segundo [6] existem três tarefas principais para se alcançar o reuso de objetos de aprendizagem: Desenvolver tecnologias que facilitem o seu reuso; Desenvolver ontologias para dar um significado pedagógico; Fazer objetos de aprendizagens que permitem o seu uso inteligente, como por exemplo, a sua recomendação personalizada. Este artigo apresenta uma proposta de algoritmo para a conversão automática de objetos de aprendizagem que estão em um repositório para um modelo em armazenamento de triplas que explora a primeira tarefa e a terceira tarefa para se alcançar o reuso de objetos de aprendizagem.

Foram reutilizados os objetos de aprendizagem conforme o padrão OBAA [9]. Foi desenvolvido um protótipo que através do acesso a uma interface de comunicação recupera e posteriormente identifica os metadados que compõem os objetos de aprendizagem, e com auxílio de estruturas que representam a ontologia, os metadados são então mapeados para propriedades de dados, e quando necessário propriedades de objetos.

Na primeira seção é feita uma revisão bibliográfica onde é apresentada uma visão geral sobre a proposta da Web Semântica, alguns conceitos sobre ontologias e a tecnologia OWL, os princípios dos objetos de aprendizagem e de metadados educacionais finalizando a revisão técnico/teórica com algumas observações sobre repositórios de conteúdos educacionais. Na seção *Solução Proposta*, é apresentado o algoritmo proposto para a conversão de metadados em indivíduos OWL, sendo apresentada uma visão sobre o problema e a explicação de cada uma das etapas relacionada a conversão a proposta de mapeamento. Na seção *Resultados*, são apresentados e analisados os resultados obtidos, e por fim, as conclusões e trabalhos futuros.

2. REVISÃO BIBLIOGRÁFICA

2.1 Web Semântica

A ideia da Web Semântica foi proposta por Berners Lee e colegas em [1] e tinha como objetivo criar uma expansão da web atual, visando o uso de aplicações inteligentes que

auxiliassem nas tarefas dos usuários. A principal motivação, veio do aumento do volume de informações na web e da popularização dos primeiros sistemas de banco de dados. Tamanha popularidade originou excesso de informação para o usuário, tanto para pesquisa como para interpretação dessas informações. A Web Semântica propõe que os computadores auxiliem os usuários em tal tarefa. Para isso é necessário uma reestruturação da web uma vez que ela é voltada para o ser humano não para o computador.

Berners Lee propôs uma “pilha” para estruturar a Web Semântica. A primeira camada Unicode/URI que garante uma padronização nos caracteres com o uso do Unicode como codificação de caracteres, e uma localização e identificação de forma única dos recursos com o uso de um identificador universal (URI). A segunda camada é onde há os metadados que descrevem as informações. Para essa camada, foi escolhida a linguagem XML para padronizar a troca de dados entre aplicações. Na terceira camada a semântica começa a ser apresentada. É utilizada a tecnologia RDF para estabelecer relações entre as coisas, com isso são utilizadas, por exemplo triplas. O RDF embora eficaz não carece em alguns aspectos semânticos, para isso foi proposta sua extensão RDF(s) que apesar de agregar mais semântica em suas definições, ainda tinha grandes limitações. Na quarta camada é utilizado o conceito de ontologia que permite, entre outras coisas, adicionar cardinalidade. A tecnologia mencionada nessa camada é a OWL que será mais explorada na subseção *Ontologias*. A quinta camada é a lógica, onde são definidas as regras pelas quais vão atuar os mecanismos de inferências nas instâncias para formar a sexta camada que é a camada prova. Por fim, camada de validação que vai verificar se a prova gerada está correta. Por exemplo, duas aplicações podem estar fazendo inferências sobre as mesmas regras, os resultados obtidos pelas duas aplicações podem ser comparados para verificar se as inferências foram feitas corretamente.

2.2 Objetos de Aprendizagem e Metadados Educacionais

A IEEE definiu que um metadado educacional é um dado sobre algum objeto digital ou físico [5], essa definição será usada nesse artigo. Os metadados promovem a interoperabilidade, ou seja, permitem que a informação originada em um contexto ser usada em outro. Um exemplo da utilização de metadados são os objetos de aprendizagem. Um objeto de aprendizagem(OA) é um material educacional, que pode ser qualquer coisa usada para fins educacionais, e seus metadados que o descrevem, podendo ser utilizado, reutilizado e referenciado durante o processo educacional [5].

Os padrões de metadados educacionais visam facilitar o compartilhamento e troca de objetos de aprendizagem levando em conta a diversidade cultural e linguística [5]. Entre os principais padrões de metadados educacionais podemos citar o *IEEE Learning Object Metadata*, *Dublin Core*, e o padrão brasileiro OBAA.

O IEEE LOM é um padrão educacional proposto pela IEEE em 2002 [5], possui uma estrutura hierárquica com nove categorias: *general*, *lifecycle*, *meta-metadata*, *technical*, *educational*, *rights*, *relation*, *annotation* e *classification*.

O padrão *Dublin Core (DC)*, surgiu em 1995, em um *workshop*,

trata-se de um padrão de metadados não especificamente educacional, mas contém características que podem ser utilizadas para objetos de aprendizagem [3].

Dentro do contexto educacional brasileiro, o padrão OBAA proposto em 2009 [10], trata-se de um padrão de metadados educacional multiplataforma baseado nos padrões LOM e IMS. No reuso do LOM, parte das modificações foram feitas especialmente na camada *Technical* e *Educational*, para se adequar às tecnologias e aos aspectos da educação no Brasil. O *IMS AccessForAll* preenche os requisitos para a utilização dos objetos de aprendizagem em diversos dispositivos por pessoas com necessidades especiais [4]. O OBAA foi escolhido como padrão de metadados a ser utilizado nesse trabalho, por ser multiplataforma, permitindo independência tecnológica e por se enquadrar na realidade educacional brasileira.

2.3 Repositórios

Os objetos de aprendizagem precisam ser armazenados de forma persistente para possibilitar o seu reuso. Tais repositórios auxiliam nessa tarefa. Além disso, repositórios tem papel importante na recuperação dos objetos de aprendizagem. O padrão OBAA conta com um repositório baseado em DSpace, onde os metadados são guardados em um banco de dados relacional. A recuperação de metadados padrão no DSpace é feita a tradicionalmente através do protocolo OAI-PMH, entretanto é possível acrescentar *web services* que permitem flexibilidade quanto ao acesso aos conteúdos de tal repositório.

2.4 Ontologias

A definição mais utilizada de ontologia, que também será utilizada durante o artigo, é a definida em [2]. Define-se ontologia como sendo uma especificação formal explícita de uma conceitualização compartilhada. De acordo com a definição as ontologias precisam ser formais para que um computador possa interpretar. O conhecimento expresso também deve ser comum a todos, ou seja uma ontologia deve transmitir um conhecimento consensual.

No modelo da Web Semântica descrita por Berners Lee as ontologias ocupam a quarta camada. A criação de uma ontologia envolve vários aspectos, definir classes, subclasses, propriedades e restrições. Sendo que estas podem ser representadas por triplas. Uma tripla é composta por um sujeito, um verbo e um objeto. Por exemplo, *UsuárioX conhece UsuárioY*.

Em suma representam relações binárias, que podem ser realizadas por uma propriedade de dado ou uma propriedade de objeto. Uma propriedade de dado é uma relação entre um indivíduo e um valor de dado, na tripla o sujeito pode ser o indivíduo, o verbo é a propriedade e o objeto pode ser o valor de dado. Uma propriedade de objeto relaciona dois indivíduos, o objeto e o sujeito são indivíduos e o verbo representa a propriedade.

Uma alternativa para a representação de uma ontologia é através da linguagem OWL. Existem vários níveis dessa linguagem que delimitam o poder de representação, a ontologia mencionada nesse artigo faz uso da OWL-DL, visando expressividade sem perder a computabilidade (computadores

tem capacidade para seu processamento) e conclusividade (seu processamento se dá em tempo finito).

3. SOLUÇÃO PROPOSTA

Essa seção apresenta o algoritmo proposto pelo artigo para a conversão de OAs baseados em metadados para OAs baseados em OWL. O processo de conversão está organizado em quatro etapas: Captura de metadados, onde os metadados são recuperados do repositório; Criação do dicionário¹, onde o esquema da ontologia é mapeado em um dicionário; Conversão dos metadados, onde os metadados são convertidos em propriedades de um indivíduo através de buscas no dicionário; Montagem do indivíduo, onde as propriedades geradas na etapa anterior se agrupam para formar um indivíduo.

3.1 Captura de metadados

Pelo fato do repositório OBAA ser compatível com a tecnologia REST, sendo esta baseada em requisições por HTTP, e compatível com a Web Semântica, tal interface de comunicação, que caracteriza um *web service* foi utilizada, sendo assim, os OAs e os metadados obtidos através das requisições feitas ao *web service*. Após a obtenção dos metadados, eles passam por um processo de categorização. Esse processo agrega os metadados que se referem aos mesmos objetos de aprendizagem. Ele é feito para facilitar o processo de criação do indivíduo posteriormente.

O processo de conversão inicia fazendo uma requisição para recuperar os metadados do repositório OBAA ao *web service* que retorna todos os metadados do repositório. Os metadados passam pelo processo de categorização descrito anteriormente, onde são agrupados em objetos. Cada objeto contém metadados relacionados a apenas um objeto de aprendizagem e todos os metadados relacionados a um objeto de aprendizagem estão agrupados em apenas um objeto. Após essa etapa, existe um objeto para cada objeto de aprendizagem, onde há os seus metadados. Os objetos dos OAs são agrupados em uma lista. A estrutura escolhida foi uma lista pois permite um acesso sequencial aos objetos.

3.2 Criação do dicionário

Paralelamente a etapa anterior, pode-se fazer a criação do dicionário. O dicionário é uma estrutura que representa o esquema de uma ontologia para facilitar a conversão dos metadados em indivíduos OWL. Na implementação do algoritmo foi utilizado uma árvore n-ária para representar a ontologia do padrão OBAA. É possível usar esse tipo de estrutura pois a ontologia do padrão OBAA não possui ciclos, de modo mais abrangente, poderia se definir a estrutura do dicionário como um grafo, entretanto a pesquisa exigiria algoritmos para controlar ciclos.

O mapeamento do esquema de ontologia para o dicionário é direto. Cada nodo no dicionário representa uma classe, que pode conter propriedades de dados e propriedades de objetos e pode conter uma lista de nodos filhos (subclasses). As propriedades de dados e as propriedades de objeto têm estruturas de representação próprias. Essas estruturas que

¹Um dicionário é uma estrutura de dados que representa a ontologia de forma hierárquica.

representam o dicionário também serão utilizadas para compor um ou mais indivíduos que irão compor uma ontologia. Pode-se interpretar o dicionário como sendo um indivíduo completo que tem todas as propriedades possíveis para um indivíduo, mas com suas propriedades sem valores ainda definidos.

3.3 Conversão dos metadados

O processo de conversão é dependente do formato pelo qual os OAs são recuperados. No momento de criação de um OA cada um de seus metadados deveria ter pelo menos um *path*, um *schema*, um *value* e um *individualId*. O *path* é um indicador para a propriedade de dado que o metadado se refere, normalmente é composto pelo nome da propriedade de dado concatenado com o nome de algumas classes que ajudam na busca durante as consultas no dicionário.

O campo *schema* representa o esquema de metadados onde a propriedade se refere à propriedade de dado. Na implementação esse atributo é concatenado com o *path*, pois a única ontologia utilizada é o OBAA, ou seja, um dicionário é pesquisado, mesmo que este agregue outras ontologias, entretanto pode existir mais de uma ontologia para pesquisa, e esse campo se faz necessário. *Value* é o valor que será atribuído à propriedade de dado do indivíduo OWL. O atributo *individualId* serve para identificar o indivíduo ao qual pertence a propriedade de dado. Esse atributo é muito importante para propriedades de dado com cardinalidade maior que um. Propriedades de dado com cardinalidade maior que um geram um novo subindivíduo, que se relaciona com o indivíduo principal através de uma propriedade de objeto. É esperado que vários subindivíduos sejam gerados, e que esses subindivíduos tenham algumas vezes propriedades iguais, o campo *individualId* deve relacionar a propriedade ao indivíduo principal ou a algum subindivíduo.

Um problema encontrado durante a implementação do algoritmo foi a representação dos metadados no repositório DSpace, como os dados estão representados em um esquema *flat* não era possível representar a pertinência de uma propriedade de dado a um subindivíduo comprometendo propriedades de dados com cardinalidade maior que um.

O DSpace não representa hierarquicamente os metadados, não há como saber no retorno se o *lifecycle.contribute.role* com valor *author* pertence ao mesmo subindivíduo que a propriedade *lifecycle.contribute.entity* com valor *NUTED, Núcleo de Tecnologia Digital aplicada à Educação*, mesmo os metadados estando agrupados por indivíduos, deveria ter alguma forma de diferenciá-los por subindivíduos. No algoritmo estão previstos esses tipos de propriedades e uma alternativa para a representação do DSpace que consiga representar semanticamente as características das propriedades de dados da ontologia ainda está sendo trabalhada.

A conversão dos metadados ocorre com a pesquisa no dicionário. É feita uma busca em profundidade com um metadado de um objeto na árvore e o caminho bem sucedido é copiado do dicionário e acrescentado um valor, essa estrutura representa a propriedade.

O *schema* é concatenado com o *path*, assim é obtida a *string lom.general.title* para pesquisar no dicionário. O primeiro

elemento da *string* que está concatenado é *lom*. Esse elemento é utilizado para selecionar um nodo filho do dicionário para continuar a pesquisa, caso não for encontrado nenhum nodo filho que tenha aquele nome, todos os nodos filhos serão pesquisados recursivamente, caso não achar em nenhum nodo do dicionário, o elemento seguinte é concatenado com anterior (a *string* passaria a ser *lom.general*), isso se deve ao fato de existirem propriedades de dados que tem o nome da classe que pertencem concatenado ao delas, para fins de diferenciação. Se mesmo assim nenhum nodo for achado, o metadado é eliminado do conjunto que representa os metadados do indivíduo. Caso o elemento *lom* achou um nodo que referencia, esse nodo e todo o caminho feito até achar ele são copiados e o próximo elemento da *string* passa ser procurado, no caso *general*. O elemento é procurado a partir do nodo atual (*lom*). Caso o nodo que está sendo procurado existir, o caminho feito até achá-lo é copiado e agregado com o caminho que foi copiado anteriormente. Isso ocorre até a string acabar. Quando o último elemento for encontrado, o valor da propriedade, encontrado no campo *value* do metadado (no exemplo é o valor "Outras Infâncias"), é atribuído ao último nodo. No final dessa etapa de pesquisa há uma lista encadeada de nodos que representam o caminho feito da classe mais geral da ontologia até a propriedade, sendo que no último nodo há o valor da propriedade de dados. Com essa representação é possível analisar se é necessário criar algum subindivíduo. Caso haja alguma propriedade de objeto no caminho copiado do dicionário essa propriedade pertence a um subindivíduo, o atributo *individualId* garante que a propriedade de dados será inserida no subindivíduo correto.

3.4 Construção dos indivíduos

A montagem dos indivíduos consiste em criar relações entre as propriedades de dados e subindivíduos com o indivíduo principal. Através da análise dos caminhos obtidos da fase anterior é possível decidir se a propriedade de dados que será gerada pertence ao indivíduo principal ou a um subindivíduo. Caso pertença ao indivíduo basta inserir a propriedade. Caso pertença a um subindivíduo há uma pesquisa para ver se o subindivíduo já existe de acordo com o *individualId* do metadado, se já existe basta inserir a propriedade, caso contrário deve-se criar o subindivíduo, buscar a propriedade de objeto no caminho copiado do dicionário, e relacionar o indivíduo principal com o subindivíduo usando a propriedade de objeto. Após a criação do subindivíduo basta inserir a propriedade de dado.

4. RESULTADOS

Ao todo foram convertidos 60 objetos de aprendizagem do repositório OBAA. Para verificar se os indivíduos foram gerados corretamente eles passaram por um *reasoner*, que é um mecanismo de raciocínio que identifica automaticamente inconsistências nos indivíduos gerados da conversão. O processo revelou inconsistências nos indivíduos, entretanto ao fazer uma análise sobre os metadados retornados do repositório, foi constatado que havia algumas inconsistências da definição dos metadados no repositório. Os erros de inconsistência foram corrigidos manualmente nos indivíduos após gerados. Os indivíduos foram novamente submetidos ao *reasoner* e não foi encontrada nenhuma inconsistência. Percebeu-se também, que objetos de aprendizagem, compostos por muitos indivíduos provenientes de ontologias como a LOM, onde as propriedades são compostas por informa-

ções semânticas como cardinalidade e tipo de dado, acabaram por influenciar em um maior tempo de processamento do *reasoner*.

5. CONCLUSÃO

Esse artigo apresentou uma proposta de algoritmo para a conversão de metadados de objetos de aprendizagem seguindo o padrão OBAA para indivíduos. Foram feitos testes com os indivíduos gerados e com isso foi possível achar inconsistências na implementação do esquema de metadados no repositório. Como trabalhos futuros está previsto achar alguma alternativa para corrigir o esquema *flat* do DSpace: ou acrescentar um campo nos metadados que retornam do DSpace para representar um identificador para o subindivíduo ou fazer a transição do DSpace para uma base de dados baseada em triplas. Outro trabalho futuro seria reaproveitar o *Reasoner* por outros aplicativos, como por exemplo, verificação de inconsistências, explain e derivação de conhecimento através de axiomas para auxiliar sistemas de recomendação como encontrado em [7].

6. REFERENCES

- [1] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5):34–43, May 2001.
- [2] W. N. Borst. *Construction of Engineering Ontologies for Knowledge Sharing and Reuse*. PhD thesis, Enschede, September 1997.
- [3] DCMI Usage Board. DCMI metadata terms. DCMI recommendation, Dublin Core Metadata Initiative, December. Published online on December 18th, 2006 at <http://dublincore.org/documents/2006/12/18/dcmi-terms/>.
- [4] IMS. IMS Access For All. http://www.imsglobal.org/accessibility/accmdiv1p0/imsaccmd_infov1p0.html. Accessed: June, 2013.
- [5] Learning Technology Standards Committee of the IEEE. Draft standard for learning technology - learning object metadata. Technical report, IEEE Standards Department, New York, July 2002.
- [6] P. Mohan and C. A. Brooks. Learning objects on the semantic web. In *ICALT*, pages 195–199. IEEE Computer Society, 2003.
- [7] T. T. Primo and R. M. Vicari. A recommender system that allows reasoning and interoperability over educational content metadata. In *Proceedings of the 2011 IEEE 11th International Conference on Advanced Learning Technologies*, ICALT '11, pages 598–599, Washington, DC, USA, 2011. IEEE Computer Society.
- [8] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors. *Recommender Systems Handbook*. Springer, 2011.
- [9] R. Vicari, A. Ribeiro, J. Silva, and E. Santos. Brazilian Proposal for Agent-Based Learning Objects Metadata Standard-OBAA. *Metadata and Semantic ...*, 108:300–311, 2010.
- [10] R. M. Vicari, J. C. Gluz, L. Passerino, E. Santos, T. Primo, L. Rossi, A. Bordignon, P. Behar, R. Filho, and V. Roesler. The OBAA Proposal for Learning Objects Supported by Agents. *AAMAS*, May 2010.