

Ontologias em sistemas multi-agentes

Veruska R. Moreira

Universidade Estadual de Campinas (Unicamp)
Faculdade de Engenharia Elétrica e de Computação (FEEC)
veruska@decom.fee.unicamp.br

Abstract— This paper provides a comprehensive literature review on ontology as a mechanism for application integration and knowledge sharing in multi-agent system. Following the ontology life cycle, this paper compares several ontology languages, ontology building methodologies and tools. It reviews some recent work on applying to negotiation in multi-agent system.

Index Terms— Ontology, Application Integration, Multi-agent system.

I. INTRODUÇÃO

NOS últimos anos, a comunidade científica tem incentivado as pesquisas em ontologias aplicadas em sistemas multi-agentes. O uso de ontologias na área de agentes de negociação é um tema emergente.

Este artigo apresenta uma revisão de literatura sobre as ontologias computacionais. Na seção II é apresentada a metodologia de pesquisa empregada na revisão de literatura. A seção III mostra uma visão geral sobre as ontologias. A seção IV aborda a questão das ontologias aplicadas em sistemas multi-agentes. Essa seção aborda especialmente o uso de ontologias de negociação. A seção V mostra um exemplo de ontologia empregada em e-commerce e a seção VI apresenta a conclusão.

II. METODOLOGIA DE PESQUISA

Neste trabalho, foi realizado um levantamento bibliográfico na área de ontologias aplicadas em sistema multi-agentes. A base de dados do IEEE (IEEEExplore) foi utilizada para a pesquisa. A estratégia de busca realizada empregou os seguintes termos: *Ontology in abstract, knowledge sharing in abstract, multi-agent system, ontology languages and collaborative design*. A busca foi refinada para apresentar somente o conteúdo que foi publicado em *Conferences* e *Journals*, cujo assunto estava relacionado a *Computing & Processing (Hardware/Software)* ou *Bioengineering*. O ano de publicação dos artigos compreendeu o período de 2006 a 2010. A busca retornou dezessete artigos, dos quais cinco foram selecionados. Os artigos selecionados abordam questões relacionadas à área de pesquisa em ontologias de negociação aplicadas em sistemas multi-agentes.

III. ONTOLOGIAS

O corpo de conhecimento formalmente representado é baseado em uma conceituação. O corpo de conhecimento é formado por objetos, conceitos, outras entidades e suas relações que existe na área de interesse. A conceituação é uma abstração, uma visão simplificada do mundo que se deseja representar para atender algum propósito (1). Cada base do conhecimento ou sistema baseado em conhecimento está comprometida com alguma conceituação implícita ou explícita.

Uma ontologia é uma especificação explícita de uma conceituação (1). O termo é emprestado da filosofia, onde uma ontologia é um relato sistemático da existência. Para um sistema baseado no conhecimento, o que existe é exatamente o que pode ser representado. Quando o conhecimento de um domínio é representado em um formalismo declarativo, o conjunto de objetos que podem ser representados é chamado de universo de discurso (1). Este conjunto de objetos e suas relações são refletidos na descrição do vocabulário que representa o conhecimento em um sistema baseado em conhecimento. Assim, uma ontologia pode ser descrita definindo um conjunto de termos representacionais.

A estrutura de uma ontologia é composta de um vocabulário de termos utilizados no domínio e uma especificação semântica deste vocabulário (2). Isto inclui as regras para a criação e uso deste vocabulário. Essas regras indicam como os conceitos estão inter-relacionados. A tautologia também pode fazer parte da estrutura de uma ontologia. A tautologia é um conjunto de proposições válidas para o domínio em questão que restringe uma possível interpretação dos termos.

A. Classificação das ontologias

As ontologias podem ser classificadas em termos de tipos e de formalidades. De maneira geral, há quatro tipos (ontologia de domínio, ontologia de tarefa, ontologia de senso-comum e meta-ontologia) e quatro formalidades (informal, semi-informal, semi-formal e rigorosamente formal) (2).

Tipos de ontologias

A ontologia de domínio fornece um vocabulário para descrever um dado domínio. Geralmente, ela inclui termos relacionados a objetos e seus componentes, um conjunto de verbos e sinônimos que descrevem as atividades, os processos e as relações e fórmulas que regem o domínio.

A ontologia de tarefa descreve a estratégia adotada para a resolução de um problema de um determinado domínio. A

ontologia de tarefa pode ser utilizada para descrever a resolução de problemas que utilizam lógica *fuzzy* ou redes neurais, por exemplo.

A ontologia de senso-comum não está vinculada a um problema particular do domínio, mas procura descrever entidades gerais como o tempo, o espaço, os eventos, etc.

A meta-ontologia fornece um núcleo básico de conceitos utilizados para codificar a ontologia de domínio, a ontologia de tarefa ou a ontologia de senso-comum em uma linguagem formal de representação do conhecimento.

Formalidade das ontologias

O grau de formalidade em que é criado e especificado um vocabulário varia consideravelmente. Em (3) e (4) são definidos em quatro graus de formalidades:

- Informal: é expresso livremente em linguagem natural. Por exemplo, catálogos e glossários.
- Semi-informal: é expresso em linguagem natural de maneira restrita e estruturado. Isso aumenta a clareza e diminui a ambigüidade. Por exemplo, as categorias do Yahoo.
- Semi-formal: expresso em linguagem artificial definida formalmente. Por exemplo, as hierarquias de subclasses e as relações de instância formal.
- Rigorosamente formal: os termos são definidos cuidadosamente com semântica formal, teoremas e provas de propriedades. Por exemplo, classes que inclui informação de propriedade, relacionamento parte-todo, relacionamento inverso, etc.

B. Linguagem de ontologias

A linguagem é essencial no desenvolvimento de uma ontologia. Esforços significativos têm sido realizados no desenvolvimento de várias linguagens de ontologia. Antes de apresentar algumas linguagens disponíveis, é necessário entender quais são os mínimos requisitos que uma linguagem de ontologia deve atender.

Uma lista de requisitos foi criada pelo *WebOntology* (WebOnt), o grupo de trabalho do W3C. Esses requisitos fornecem uma visão geral das funcionalidades que devem ser atendidas pela linguagem para serem úteis em uma aplicação da *web* semântica. Essa lista de requisitos aborda questões como o compartilhamento, a interoperabilidade, a evolução, a expressividade e a escalabilidade da ontologia, a detecção de inconsistência, a compatibilidade com outros padrões, a facilidade de uso e a internacionalização. Algumas linguagens que atendem esses requisitos são mostradas a seguir (2):

- XML e XML *Schema* (XML e XML *Schema*): XML fornece um conjunto de regras para a criação de vocabulários com base na estrutura de documentos e dados da *web*. O XML *schema* é uma linguagem de modelagem de metadados para a definição e compartilhamento de documentos XML.
- SHOE (*Simple HTML Ontology Extensions*) (5): É uma linguagem de representação do conhecimento com base na linguagem HTML. O SHOE incorpora a estrutura da linguagem HTML e adiciona *tags* necessárias para incorporar semântica de dados em páginas *web*.

- RDF (*Resource Description Framework*) (RDF) e o RDF *Schema* (RDFS): É uma linguagem de proposta geral para representar informação na *web*. Ela fornece um grau de interoperabilidade semântica entre aplicações que trocam informação na *web*. O RDFS é uma linguagem que específica como utilizar e descrever vocabulários RDF. Ela define o vocabulário, a estrutura, as restrições, as propriedades do domínio e as relações das classes e das propriedades.

- DAML (*DARPA Agent Markup Language*) e DAML-ONT (6): O DARPA é uma agência de pesquisa e desenvolvimento associada ao departamento de defesa dos Estados Unidos. O DARPA é responsável pelo projeto DAML que desenvolveu diversas ferramentas e linguagens para facilitar a aplicação do conceito de *web* semântica. A DAML-ONT é uma linguagem desenvolvida no âmbito deste projeto para a representação do conhecimento. Essa linguagem é uma extensão da linguagem RDF e utiliza os conceitos de frames e de orientação a objetos.

- *Ontology Interchange Language* (OIL): A sintaxe da linguagem OIL é baseada na linguagem RDFS e acrescenta um conjunto de construtores que estão presentes na linguagem baseada em *frame*. Os desenvolvedores da linguagem OIL enfatizaram o rigor formal e a linguagem foi explicitamente projetada para que a semântica pudesse ser especificada através de um mapeamento bastante expressivo de descrição lógica.

- *Web Ontology Language* (OWL): É uma linguagem de marcação semântica para publicar e compartilhar ontologias na *web*. O OWL é desenvolvido como uma extensão de vocabulários da linguagem RDF e foi desenvolvida com base na linguagem DAML e OIL. A linguagem OWL fornece vocabulário adicional para descrever propriedades e classes com uma semântica formal. É bastante utilizada por aplicações que precisam processar propriedades e classes com uma semântica formal.

C. Metodologia para a construção de ontologias

Gomez-Perez (7) propôs uma metodologia para a construção de ontologias que envolvem o critério de projeto, as atividades integrais e as atividades de desenvolvimento.

Critério de Projeto

Com base no trabalho de Gruber (8) e Borgo, Gomez-Perez (7) resumiu os princípios e critérios de projetos conforme descrito a seguir:

- Clareza e objetividade: A ontologia deve fornecer uma definição objetiva de todos os termos e uma documentação dos termos em linguagem natural.
- Integralidade: A definição dos termos deve ser completa.
- Coerência: A ontologia deve permitir inferências que são consistentes com as definições.
- Extensível: Novos termos devem ser incluídos na ontologia de maneira que não seja necessário fazer uma revisão das definições existentes.
- Mínimo compromisso: Deve especificar somente o necessário sobre o significado de seus termos.
- Princípio da distinção: Classes em uma ontologia devem ser disjuntas. Este critério é utilizado para isolar o núcleo das

propriedades consideradas invariante de uma instância de uma classe. Também é conhecido como critério de identidade.

Atividades de Desenvolvimento

As tarefas listadas a seguir, descrevem as técnicas e métodos utilizados para desenvolver uma ontologia.

- Especificação: Uma ontologia não deve ser desenvolvida sem conhecer o porquê que ela deve ser construída, quais são as suas finalidades e quem são os usuários finais.
- Conceituação: O objetivo é construir um modelo conceitual que descreva os problemas e as soluções.
- Formalização: Esta atividade transforma um modelo conceitual em um modelo formal.
- Implementação: A ontologia é codificada em uma linguagem formal.
- Manutenção: Na ontologia as definições podem ser incluídas ou modificadas a qualquer momento. Por isso, é importante manter um guia ou manual atualizado sobre as ontologias.

Atividades Integradas

As atividades descritas, a seguir, interagem com as atividades de desenvolvimento e são importantes para o sucesso do desenvolvimento das ontologias.

- Aquisição do conhecimento: A aquisição do conhecimento é o primeiro passo para o compartilhamento do conhecimento.
- Integração: As ontologias são construídas para serem reutilizadas. Portanto, elas devem ser reutilizadas tanto quanto possível.
- Avaliação: Antes de disponibilizar uma ontologia para que outros possam utilizá-la, é necessário fazer um julgamento técnico. Uma plataforma para avaliação de ontologias está disponível em (9).
- Documentação: A ausência de uma documentação é um obstáculo importante na reutilização de uma ontologia. Portanto, é importante documentar uma ontologia o mais detalhadamente possível para que outras pessoas possam utilizar.

D. Ferramentas para a construção de uma ontologia

Diversas ferramentas de *software* livre ou *software* comercial estão disponíveis para a construção e o desenvolvimento de ontologias. Elas podem ser utilizadas para construir uma ontologia desde o início ou para reutilizar uma ontologia existente. Além das funcionalidades de edição e de navegação, essas ferramentas incluem a documentação das ontologias, permite a importação e exportação de ontologias em diferentes formatos, a visualização gráfica das ontologias construídas e das bibliotecas existentes e também permitem anexar inferências.

Algumas ferramentas para a construção de ontologias que são bem conhecidas incluem: Apollo (10), LinkFactory (11), OilEd (12), OntoEdit, Ontolingua Server (13), OntoSaurus (14), Protégé (15), WebODE (16) e WebOnto (17). A Tabela 1 mostra parte de um estudo comparativo realizado por (18) sobre as ferramentas de construção de ontologias.

| Características | Apollo | LinkFactory | OilEd | OntoEdit Free | OntoEdit Profissional | Ontolingua | Ontosaurus | Protégé | WebODE | WebOnto |
|-------------------------------|-----------------------|-----------------------------|---|-------------------------------|--------------------------------------|--|--|--|---|------------------------------------|
| Desenvolvedor | KMI (Open University) | Language & Computing | University of Manchester | Ontoprise | Ontoprise | KSL (Stanford University) | ISI (University of Southern California) | SMI (Stanford University) | Ontology Group (UPM) | KMI (Open University) |
| Disponibilidade | Open Source | Licença no site | Open Source | Freeware | Software license | Acesso Web free | Open source e acesso Web free para a versão de avaliação | Open Source | software license e acesso Web free | acesso Web free |
| Arquitetura de Software | Standalone | 3 camadas | Standalone | Standalone | Standalone e cliente/servidor | cliente/servidor | cliente/servidor | Standalone | 3 camadas | cliente/servidor |
| Extensível | Plugins | yes | No | Plugins | Plugins | none | none | Plugins | Plugins | No |
| Importação das linguagens | Apollo meta-linguagem | XML, RDF(S), DAML+OIL | RDF(S), OIL, DAML+OIL | XML, RDF(S), Flogic, DAML+OIL | XML, RDF(S), Flogic, DAML+OIL | Ontolingua, IDL, KIF | LOOM, IDL, ONTO, KIF, C++ | XML, RDF(S), XML Schema | XML, RDF(S), CARIN | OCML |
| Exportação para as linguagens | OCML, CLOS | XML, RDF(S), DAML+OIL, HTML | OIL, RDF(S), DAML+OIL, SHIQ, doty, HTML | XML, RDF(S), Flogic, DAML+OIL | XML, RDF(S), Flogic, DAML+OIL, SQL-3 | KIF 3.0, CLIPS, CLIPS sentential, format, CML ATP, rule engine, EpiKit, IDL, KSL rule engine, LOOM, OKBC syntax, PROLOG syntax | LOOM, IDL, ONTO, KIF, C++ | XML, RDF(S), XML Schema, Flogic, CLIPS, Java, HTML | XML, RDF(S), OIL, DAML+OIL, CARIN, Flogic, Prolog, Jess, Java, HTML | OCML, Ontolingua, GXL, RDF(S), OIL |

Tabela 1: Comparação entre as ferramentas de edição de ontologias

IV. AGENTES E ONTOLOGIAS DE NEGOCIAÇÃO

A negociação é definida como dois ou mais agentes que tomam decisões sobre pontos de interesse em comum (19). Isso constitui uma estratégia eficiente para o *Business Intelligent* da empresa. Atualmente, as pesquisas em negociação estão centradas principalmente em estimular o comportamento de negociação entre os agentes. A ontologia, como uma tecnologia de modelagem em *web* semântica, é considerada um instrumento eficaz para melhorar a capacidade de negociação dos agentes, porque pode fornecer expressões semânticas mais ricas para o processo de negociação de agentes.

A. Pesquisas em ontologia de negociação

As pesquisas atuais em negociação estão concentradas nos protocolos e nas estratégias de negociação. Os protocolos de negociação definem regras e regulamentos para que os novos participantes possam adaptar-se ao ambiente de negociação. A estratégia de negociação aborda as diferentes ações que podem ser tomadas para responder aos diferentes cenários de negociação. A maioria das ontologias de negociação é projetada para gerar protocolos para a negociação no contexto de e-business.

Alguns pesquisadores estão tentando resolver o problema da heterogeneidade que existe no processo de negociação, em termos de engenharia de ontologia. Bravo (20) abordou a questão da incompatibilidade de protocolos instalados nos agentes em diferentes ontologias de negociação. Ele propôs uma nova ontologia que envolvia um vocabulário controlado dos termos e das mensagens da negociação e um módulo de desambiguação semântica para evitar que houvesse um mal entendido entre os agentes que utilizava diferentes protocolos.

Malucelli (21) abordou a questão da heterogeneidade semântica no processo de negociação. Na ontologia de negociação tradicional, quando um novo agente quer participar do processo de negociação, ele tem que ser reprogramado para atender os protocolos utilizados no processo. Malucelli apresentou uma solução para esse problema utilizando a plataforma JADE (*Java Agent Development Framework*) e a linguagem de ontologia OWL para construir uma arquitetura de ontologia composta de duas camadas. Na primeira camada, encontra-se a ontologia chamada de *e-commerce* que contém os termos e as mensagens utilizados no processo de negociação que são compartilhados entre todos os agentes. Para uma negociação específica, o agente é codificado com a segunda camada, chamada de ontologia específica de domínio. Portanto, este projeto pode ser visto como uma solução para o problema de heterogeneidade semântica no processo de negociação. Esse projeto será apresentado em mais detalhes na próxima seção.

V. EXEMPLO DE ONTOLOGIA

Uma empresa montadora de carros compra vários componentes de diferentes fornecedores para montar o seu carro. Para cada componente que será comprado existem vários potenciais fornecedores. Apesar de o setor de automobilismo utilizar uma terminologia específica, o mesmo termo pode assumir diferentes significados (21).

A Figura 1 (a) e (b) mostram os diagramas UML de parte das ontologias existentes para dois dos agentes propostos. A Figura 1 (a) representa a ontologia do CEAg, enquanto a Figura 1 (b) representa a ontologia do SEAg. Ambas as ontologias são compostas por um conjunto de conceitos e cada conceito tem relação com outros conceitos e com um conjunto de atributos. Cada atributo tem um tipo e um valor atribuído a ele.

Através deste exemplo podemos observar algumas diferenças que pode gerar um problema de interoperabilidade durante o processo de negociação. Por exemplo, no CEAg há um conceito chamado de “motor” e na SEAg há um conceito correspondente chamado de “engine”. Observe que o conceito

é o mesmo em ambas as ontologias, mas utilizam termos, características e relacionamentos diferentes.

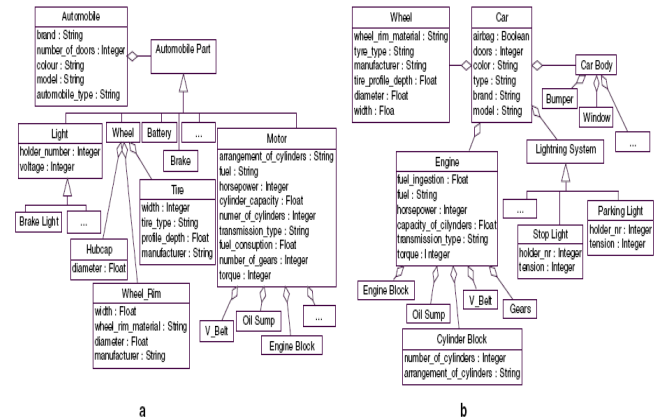


Figura 1: (a) CEAg – Ontologia de montagem de automóvel (b) SEAg – Ontologia de montagem de carro.

As ontologias de domínio mencionadas na Figura 1 (a) e (b) formam a base para que os agentes possam obter informação sobre qualquer item negociado. No entanto, para que os agentes possam participar do processo de negociação é necessário criar uma ontologia genérica, chamada ontologia para o domínio de *e-commerce*, conforme mostra a Figura 2.

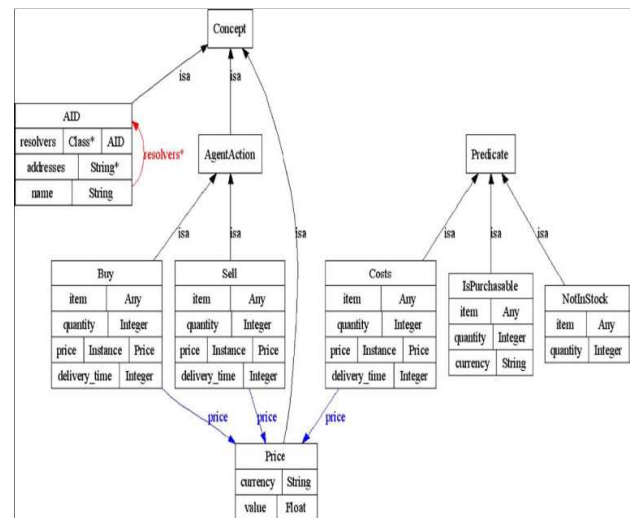


Figura 2: Ontologia e-commerce

Todas as ontologias foram criadas com a ajuda da ferramenta Protégé. Para permitir que essa ontologia seja utilizada por sistemas multi-agentes, é necessário considerar os seguintes termos que definem a ontologia de *e-commerce*: Conceitos (por exemplo, “price”), ação do agente (por exemplo, “buy”) e atributos ou predicados (“IsPurchasable”). Esses termos descrevem os conceitos básicos e os relacionamentos utilizados durante o processo de negociação garantindo a comunicação, uma vez que todos os agentes interpretarão todas as mensagens trocadas de maneira uniforme. A ontologia do *e-commerce* pode ser aplicada em qualquer domínio.

A Figura 3 mostra um exemplo de definição das subclasses da classe conceito utilizando a ferramenta Protégé.

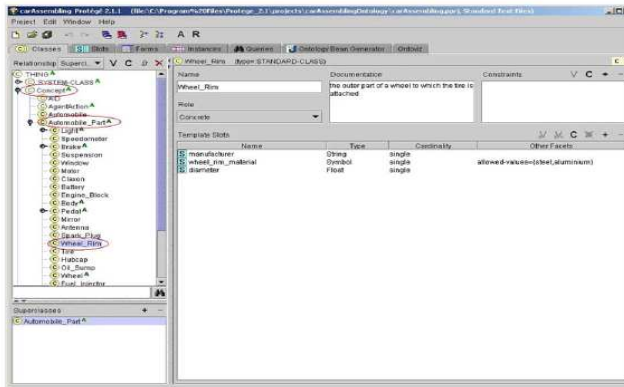


Figura 3: Hierarquia da classe conceito definida no Protégé

Caso as ontologias de domínio não estejam em conformidades, o problema de interoperabilidade semântica e sintática irá ocorrer. Para evitar isso, existem dois tipos de ontologias, a ontologia geral que é representada pela ontologia *e-commerce* e a ontologia de domínio específica, representada pela SEAg e CEAg. As mensagens são trocadas com base na ontologia de *e-commerce*, enquanto que os agentes representam a sua visão do domínio em termos da ontologia OWL, conforme mostra a Figura 4. Todos os agentes dispostos a negociar têm que concordar em utilizar a ontologia *e-commerce*, o que implica em um acordo sobre o vocabulário definido nessa ontologia. As classes da ontologia devem ser fornecidas para cada agente. Assim, todos os agentes são capazes de interpretar as mensagens da mesma forma, evitando a ambigüidade. O conteúdo diferente é expresso em termo do OWL.

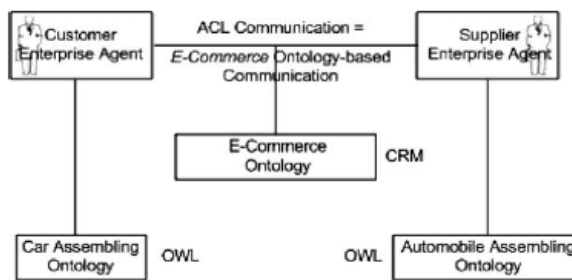


Figura 4: O uso da ontologia *e-commerce* para permitir a comunicação entre agentes.

A implementação de uma rodada de negociação combina o protocolo de interação de rede de contrato da FIPA e um protocolo adicional chamado protocolo de interação de ontologia, como mostra a Figura 5. O protocolo FIPA é utilizado no cenário geral de negociação envolvendo agentes. Ele fornece uma estrutura completa que permite o desenvolvimento do processo de negociação. O protocolo de interação de ontologia implementa o fluxo de mensagem necessária para resolver o problema da interoperabilidade que inclui a interação entre o agente fornecedor e o agente cliente.

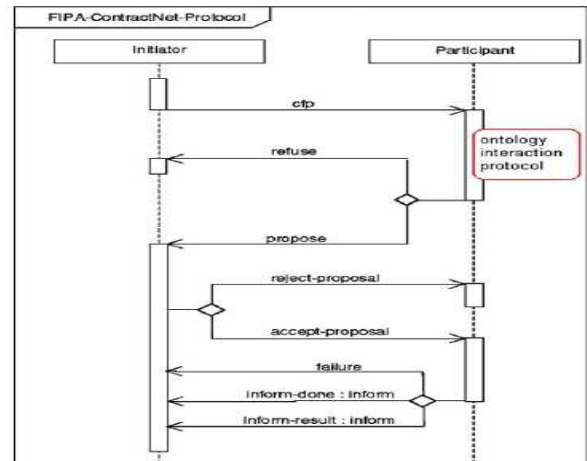


Figura 5: Protocolo de interação de rede de contrato da FIPA com o "protocolo de interação de ontologia" embutido.

O diagrama de seqüência da Figura 6 mostra a implementação do "protocolo de interação de ontologia" que visa encontrar os conceitos correspondentes nas duas ontologias (SEAg e CEAg). O agente fornecedor, depois de ter recebido um CFP como parte do protocolo FIPA e não ser capaz de interpretar o item solicitado, ele envia uma mensagem com a performativa NOT_UNDERSTOOD para o agente mediador (*ontology service agent*) indicando que recebeu um CFP e o nome do item desconhecido. O agente mediador envia o nome do item que recebeu para o agente cliente QUERY_REF a fim de obter mais informações sobre o pedido. O agente mediador irá analisa o pedido e enviará de volta os atributos do conceito, seus tipos, preços e descrição, ou seja, todas as informações sobre este item INFORM. Os fluxos 5 e 6 referem-se ao processo de pré-seleção. O processo de seleção visa obter conceitos candidatos que pode representar o produto solicitado.

Depois de ter recebido INFORM o agente mediador conhece o preço do produto e envia para o agente fornecedor (QUERY_REF. Após a seleção, o agente fornecedor responde com uma lista que contém os nomes, as documentações e os atributos dos potenciais conceitos correspondentes INFORM. Depois de receber todas as informações sobre o item em negociação e uma lista de possíveis itens correspondente, o agente mediador é capaz de aplicar os métodos necessários para verificar quais são as condições que coincidem. Este método de mapeamento de ontologia visa detectar a similaridade semântica e sintática dos termos. Na etapa 8 o agente mediador informa ao agente fornecedor o resultado de todos os itens que estão presentes no pedido ou uma mensagem indicando que o item não pode ser descoberto. O agente fornecedor então é capaz de responder ao agente cliente se aceita a proposta (PROPOSE) ou recusa (REJECT_PROPOSAL) utilizando as primitivas do protocolo FIPA.

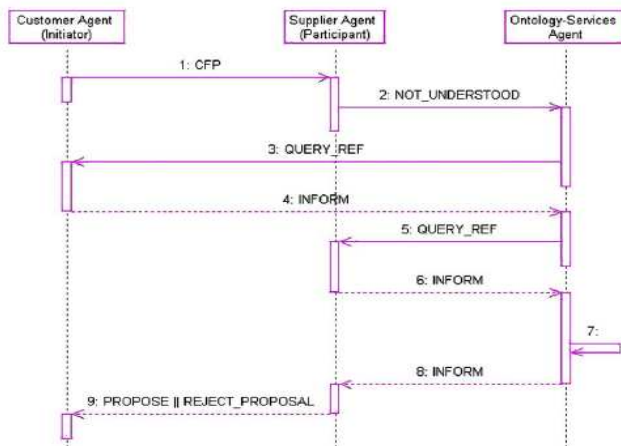


Figura 6: Interação do agente cliente com o agente fornecedor por meio de um agente mediador

Mapeamento de ontologias

O mapeamento de ontologias é o processo de encontrar as semelhanças entre os conceitos de duas ontologias. Se dois conceitos são correspondentes, eles querem dizer a mesma coisa ou coisa intimamente relacionada. O objetivo de Malucelli (21) era encontrar correspondências entre conceitos com base nos nomes dos conceitos, suas características, suas relações e descrições. Com base nisso, os autores (21) utilizaram uma medida de análise léxica chamada N-Grams para buscar similaridade entre *strings*. O algoritmo de N-Grams busca por um subconjunto de uma *string* dentro de outra *string*. Esses subconjuntos são chamados de grama e a quantidade de caracteres em cada grama pode ser definida. A Figura 7 mostra um exemplo da técnica de N-Grams aplicada às ontologias em questão.

| Type | Attributes of motor | Attributes of engine | N-grams | |
|--------|----------------------------------|----------------------|---------|--|
| Ineger | Horsepower | Horsepower | 1.0 | |
| | Horsepower | Nr_of_cylinders | 0.0 | |
| | Horsepower | Torque | 0.0 | |
| | Number_of_cylinders | Horsepower | 0.05 | |
| | Number_of_cylinders | Nr_of_cylinders | 0.7 | |
| | Number_of_cylinders | Torque | 0.0 | |
| | Number_of_gears | Horsepower | 0.0625 | |
| | Number_of_gears | Nr_of_cylinders | 0.3125 | |
| | Number_of_gears | Torque | 0.0 | |
| | Torque | Horsepower | 0.0 | |
| | Torque | Nr_of_cylinders | 0.0 | |
| | Torque | Torque | 1.0 | |
| | Total (%_gram): 3.01250 = 0.7531 | | | |

Figura 7: Exemplo da técnica de N-Grams aplicada ao conceito "motor" e "engine".

VI. CONCLUSÃO

Neste artigo foi apresentada uma visão geral sobre as ontologias aplicadas na área de computação, com base em uma pesquisa bibliográfica realizada. Questões relacionadas às ontologias, como as metodologias, as linguagens e as ferramentas relacionadas foram abordadas. Também foi apresentada uma visão geral da área de ontologias de negociação, que é um campo de pesquisa emergente. Atualmente, as pesquisas na área de ontologia de negociação aplicadas em sistemas multi-agentes visam estimular o comportamento de negociador entre os agentes por meio de tecnologias da *web* semântica. Finalmente, foi apresentado um exemplo de um sistema multi-agente para a negociação de um produto em um ambiente de *e-commerce*.

VII. REFERÊNCIA BIBLIOGRÁFICA

1. Gruber, Thomas R. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*. 1993, pp. 199-220.
2. Xie, Helen and Shen, Weiming. Ontology as a Mechanism for Application Integration and Knowledge Sharing in Collaborative Design: A Review. *Proceedings of the 10th International Conference on Computer Supported Cooperative Work in Design*. 2006, 1-7.
3. McGuinness, D. L. Ontologies come of age. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. 2002, pp. MIT Press,.
4. Uschold, M. and Gruninger, M. Ontologies: principle, methods, and applications. *Knowledge Engineering Review*. 1996, pp. 93-136.
5. Luke, S. and Heflin, J. SHOE 1.01 proposed specification, SHOE project. February 2000.
6. Hendler, J. and McGuinness, D. L. The DARPA agent markup language. *IEEE Intelligent Systems*. 2000, pp. 67-73.
7. Gomez-Perez, A. *Knowledge sharing and reuse, The handbook of applied expert systems*. s.l.: CRC Press LLC, 1997. p. CRC Press LLC.
8. Gruber, T. R. Toward principles for the design of ontologies used for knowledge sharing. *Int. Journal of Human-Computer Studies*. 1995, pp. 907-928.
9. Gomez-Perez, A., Juristo, N. and Pazos, J. Evaluation and assessment of knowledge sharing technology. *In Towards very large knowledge bases*. 1995, pp. 289-296.
10. Apollo. [Online] <http://apollo.open.ac.uk>.
11. LinkFactory. [Online] <http://www.landc.be/>.
12. OilEd. [Online] <http://oiled.man.ac.uk/>.
13. Ontoligua Server. [Online] <http://www-ksl.stanford.edu/>.
14. OntoSaurus. <http://www.isi.edu/isd/ontosaurus.html>.
15. Protégé. [Online] <http://protege.stanford.edu/>.
16. WebODE. [Online] <http://webode.dia.fi.upm.es/>.
17. WebOnto. [Online] <http://kmi.open.ac.uk/projects/webonto/>.
18. Gomez-Perez, A., et al. A survey on ontology tools. *OntoWeb deliverable 1.3*. 2002, p. Universidad Politecnica de Madrid.
19. Dong, Hai, Hussain, Farookh Khadeer and Chang, Elizabeth. State of the Art in Negotiation Ontologies for Enhancing Business Intelligence. *4th International Conference on Next Generation Web Services Practices*. 2008.
20. Bravo, M. C., et al. Ontology Support for Communicating Agents in Negotiation Processes. *the Fifth International Conference on Hybrid Intelligent Systems*. 2005, 482-487.
21. Malucelli, A., Palzer, D. and Oliveira, E. N. Ontology-based services to help solving the heterogeneity problem in e-commerce negotiations. *Electronic Commerce Research and Applications*. 2006, Vol. 5, 29-43.
22. XML e XML Schema.e] <http://www.w3.org/XML/Schema>.
23. RDF. [Online] <http://www.w3c.org/RDF/>.
24. OIL. [Online] <http://www.ontoknowledge.org/oil/>.
25. OWL. [Online] <http://www.w3.org/TR/2002/WD-owl-ref-20020729/>.
26. Borgo, S., Guarino, N. and Masolo, C. Stratified Ontologies: The case of physical objects. *Workshop on ontological engineering, EVAI'96*. 1996, pp. 5-16.