



Sobre a aplicação de ontologias para orientar agentes a responder perguntas

On the application of ontologies to guide agents answering questions

Marta Talitha C. F. de Amorim
PPGI - UFES
Av. Fernando Ferrari, 514
29.075-910 Vitória – ES – Brasil
martatcfa@gmail.com

Davidson Cury
PPGI - UFES
Av. Fernando Ferrari, 514
29.075-910 – Vitória – ES – Brasil
dedecury@gmail.com

Crediné S. Menezes
FACED - UFRGS
Av. Paulo Gama, s/n, prédio 12201
90046-900
Porto Alegre – RS – Brasil
credine@gmail.com

Resumo *As tarefas de resolver perguntas ou esclarecer dúvidas são determinadas primeiramente por uma boa análise da pergunta com o fim de identificar o assunto alvo da resposta. Este artigo apresenta um sistema de pergunta-resposta que utiliza ontologias e técnicas de recuperação de informação na análise da pergunta e dessa forma aperfeiçoa a extração da resposta. Sistemas de pergunta-reposta buscam oferecer a mesma facilidade que ocorrem nos diálogos entre as pessoas, onde as dúvidas são respondidas prontamente. Eles vão além da busca mais familiar baseada em palavras chaves, na tentativa de reconhecer o que a pergunta expressa e apresentar uma resposta correta, o que os torna muito vantajosos aos aprendizes. Ao serem incorporados a ambientes colaborativos de aprendizagem torna-se possível fixar ou examinar o conteúdo de um domínio por meio de perguntas e respostas. Um protótipo foi desenvolvido e testado e resultados são apresentados.*

Palavras-Chave: *Sistema pergunta-resposta, ontologias, base de conhecimento AIML, recuperação da informação*

Abstract *The tasks of solving question or clarifying doubts are determined primarily by a good analysis of the question in order to identify the subject target of the response. This article presents a question-answer system that uses ontologies and information retrieval techniques in the question analysis and thus improves the extraction of the response. Question-answer systems seek to provide the same facilities that occur in dialogues among people, where questions are answered promptly. They go beyond the more familiar search engines based on keywords, in an attempt to recognize what the question expresses and submit a correct answer. This feature makes them very favorable to learners. When embedded in collaborative learning environments it is possible to set or examine the content of a domain by means of questions and answers. A prototype was developed and tested and results are presented.*

Keywords: *question-answering system, ontologies, AIML Database knowledge and information retrieval.*

1. Introdução

Vivemos a gestação de uma nova cultura, uma cultura digital ou cibercultura, que provoca um profundo impacto na formação individual e do grupo. A internet é o berço dessa nova cultura. Ela é o espaço social desse novo mundo, um lugar híbrido, “construído na interface entre a experiência direta e a mediada pela comunicação” [1].

Neste cenário, a internet com seus recursos, joga um papel fundamental no que diz respeito às mudanças na educação. Nos últimos anos a internet tem acumulado uma incontável quantidade de documentos, tornando-se uma importante fonte potencial para o esclarecimento de dúvidas, suporte imprescindível à construção de conhecimento. Entretanto, seu uso para esta finalidade tem de lidar com a dificuldade que os aprendizes têm em formular suas questões em linguagem compatível com o paradigma das máquinas de busca. Até mesmo aprendizes mais experientes geralmente não conseguem formular consultas boas quando trabalham com domínios dos quais têm pouca familiaridade [2].

Para lidar com esse problema, uma possível abordagem é utilizar sistemas de pergunta-resposta (QAS) por intermédio do qual um computador responde às perguntas, formuladas em linguagem natural, automaticamente. Os QAS são especialmente úteis em situações nas quais o aprendiz deseja obter informação específica e não dispõe de tempo para ler toda a documentação disponível relacionada ao tópico sob pesquisa, para resolver o problema [3].

De modo geral, o desafio de um QAS consiste em retornar a resposta que mais se aproxima da expectativa do aprendiz para uma pergunta feita em linguagem natural. O processo completo é bastante complicado, pois requer um número de diferentes técnicas, trabalhando em conjunto, a fim de atingir o objetivo. Essas técnicas incluem: a reescrita e formulação da consulta, a classificação da pergunta, a recuperação da informação, a recuperação de passagens textuais, a extração da resposta, a ordenação da resposta e, finalmente, a justificativa [4].

O ato de aprender requer a formulação de perguntas pelo aprendiz e um sistema de apoio facilitador da obtenção de diferentes respostas (com suas fontes) a partir da qual o aprendiz poderá evoluir. Enquanto uma máquina de busca de informação clássica devolve dezenas ou centenas de *links* para assuntos, a maioria das vezes não relacionadas com o domínio de uma consulta formulada por um aprendiz, um QAS devolve respostas concisas e, se inteligente, todas elas aderentes ao contexto da consulta. Podemos, pois afirmar que um QAS cria um

ambiente mais propício a um processo de aprendizagem mais rápido e mais eficiente.

O sistema que aqui apresentamos se encontra no contexto de um projeto maior, o Multiorganizador Flexível de Espaços Virtuais (MORFEU). MORFEU é um esforço multi-institucional de pesquisa de uma concepção inovadora em busca da flexibilidade em espaços virtuais, focando fundamentalmente no suporte à elaboração de arquiteturas pedagógicas, para o qual já foram desenvolvidas várias ferramentas [5, 6, 7, 8]. Portanto, pensamos que seja também interessante pensar nas múltiplas facetas de um QAS como uma ferramenta aliada à educação. Ao ser incorporado em ambientes colaborativos de aprendizagem torna-se possível fixar ou examinar o conteúdo de um domínio por meio de perguntas e respostas.

Este artigo apresenta um sistema de pergunta-resposta que utiliza ontologias, entre outras técnicas, com a finalidade de aperfeiçoar a extração de respostas para qualquer domínio do conhecimento. O artigo está organizado da seguinte forma: Na Seção 2 apresentamos as características que consideramos importantes dos QASs. Na Seção 3 descrevemos alguns trabalhos correlatos importantes. A Seção 4 descreve o problema a ser tratado. A Seção 5 descreve a arquitetura do sistema proposto quanto ao seu funcionamento e seus agentes de software. A Seção 6 apresenta alguns detalhes tecnológicos da implementação de um protótipo bem como de uma ontologia desenvolvida para lhe dar apoio e os resultados obtidos com alguns experimentos concretos. E por final, na Seção 7 citamos trabalhos futuros e tecemos algumas conclusões temporárias sobre o trabalho em realização.

2. Sistemas de Pergunta-Resposta

Um sistema de pergunta-resposta fornece resposta “exata” para perguntas em linguagem natural. A noção de “exata” nesse contexto é uma medida subjetiva que pretende indicar que um QAS distingue-se por tentar fornecer a resposta que contém apenas a informação necessária para responder a pergunta. A resposta exata pode trazer informações adicionais ou complementares, incluindo uma justificativa ou diálogo, explicando o porquê a resposta está correta [9].

Sistemas de pergunta-resposta buscam oferecer a mesma facilidade que ocorrem nos diálogos entre as pessoas, onde as dúvidas são respondidas prontamente. Eles vão além da busca mais familiar baseada em palavras chaves (como no Google, Yahoo, entre outros motores de busca), na tentativa de reconhecer o que a pergunta expressa e apresentar uma resposta correta. Isto representa duas vantagens para os aprendizes. Primeiro, perguntas não se traduzem em uma simples lista de palavras chaves. E, segundo, sistemas pergunta-resposta assumem a responsabilidade de fornecer a resposta ao

invés de uma lista de documentos potencialmente relevantes [10].

Sistemas de pergunta-resposta têm sido abordados a partir de diferentes perspectivas. Abordagens baseadas nas ciências cognitivas estão preocupadas com a simulação humana em responder e perguntar. Problemas de compreensão e geração da linguagem natural vêm à tona em QAS, pois grandes bases de dados de documentos requerem uma análise linguística sofisticada, incluindo entendimento do discurso e sumarização de texto. Os mecanismos de raciocínio para QAS são preocupações dos pesquisadores na representação do conhecimento [11].

Nossa proposta é pesquisar as implicações do uso de inferências em ontologias de domínio para enriquecer a extração da resposta em um QAS. Além de representar conhecimento, a ontologia pode ser útil para [12]:

- Compartilhar entendimento comum da estrutura da informação entre pessoas ou entre agentes. Entender a utilidade para QAS com o que poderíamos, por exemplo, inserir o QAS em ambientes educacionais e, por meio de uma integração semântica, compartilhar um entendimento comum;
- Permitir reutilizar conhecimento do domínio. Reutilizar conhecimento poupa esforços e aumenta a qualidade, uma vez que os esforços anteriores que seriam gastos na estruturação do conhecimento são concentrados em aperfeiçoar o QAS. A reutilização de conhecimento também permite a criação de padrões;
- Fazer hipóteses explícitas do domínio. A ontologia, como uma linguagem lógica e matemática, permite realização de inferências. Com isso novos conhecimentos podem ser incluídos, agregando coesão ao domínio;
- Separar conhecimento do domínio do conhecimento operacional, facilitando o reuso e a organização;

Analisar o conhecimento do domínio. Tendo em vista que para analisar o conhecimento do domínio é usada uma metodologia, então podemos pensar em etapas organizadas, em atividades com o fim de examinar o conhecimento, por exemplo: atividade de definição do conceitos de um domínio, atividade para organizar os conceitos em hierarquias, atividade para definir quais atributos e propriedades (slots) as classes devem ter etc.

Balduccini et al. [13] classificaram os QASs que incorporam representação do conhecimento e raciocínio, baseados em três abordagens: 1) em lógica formal, 2) em extração de informação e 3) usando lógica formal na extração da informação.

Todas as três abordagens em algum momento utilizam linguagens lógicas para extrair algum novo conhecimento inferido. Sistemas que utilizam ontologia para raciocinar e representar conhecimento são conhecidos por utilizar modelagem qualitativa (ou raciocínio qualitativo). Um dos objetivos da modelagem qualitativa é tornar o conhecimento tácito em explícito, fornecendo formalismos que podem ser usados para automatizar (total ou parcialmente, dependendo da tarefa) o processo de raciocínio [14].

Muitos trabalhos na literatura [15, 18, 19] têm utilizado ontologias em seus diversos aspectos e por meio dos experimentos têm apresentado ganhos. Na próxima seção detalharemos alguns desses trabalhos.

3. Trabalhos Correlatos

Apresentamos aqui os resultados de uma pesquisa bibliográfica sobre o uso das ontologias em sistemas pergunta-resposta. Para cada sistema encontrado, damos uma explanação dos seus aspectos que consideramos mais relevantes, do ponto de vista da nossa investigação.

O FREyA [15] traduz uma pergunta em linguagem natural ou palavras chaves em uma consulta SPARQL, e retorna a resposta para o aprendiz depois de executar uma consulta na ontologia. A dinâmica do sistema pode ser resumida nos seguintes passos: 1) Identifica e verifica os conceitos da ontologia, 2) gera a consulta SPARQL, 3) identifica o tipo da resposta e 4) apresenta o resultado para o aprendiz.

O algoritmo para traduzir uma pergunta em linguagem natural em um conjunto de conceitos da ontologia combina análise sintática com raciocínio na ontologia. Nos casos em que o algoritmo não infere conclusões automaticamente, sugestões são geradas para o aprendiz. Ao envolver o aprendiz em um diálogo, têm-se melhores chances de identificar as informações que são consideradas ambíguas. Na fase de identificação dos conceitos, é utilizado conhecimento disponível na ontologia para reconhecer e anotar na pergunta os termos da ontologia. Se existir anotações ambíguas na consulta, é realizado um diálogo com o aprendiz. Os testes obtiveram um recall de 92.4% sobre um total de 250 questões.

O PowerAqua [19] é uma evolução de outro sistema chamado Aqualog, um sistema baseado em ontologia. Na arquitetura do PowerAqua, o componente para a análise da consulta do aprendiz utiliza um componente linguístico para processá-la. A saída desse componente é um conjunto de triplas linguísticas (< sujeito, predicado, objeto>) que é mapeado para a consulta do aprendiz. Assim é possível realizar buscas das respostas em bases

OWL. Os resultados obtidos nos testes apresentaram 48 (69,5%) questões respondidas das 69 questões totais.

O OMeD [17] é um sistema que apoia a tomada de decisões médicas por meio da qual respostas são fornecidas em tempo real para perguntas médicas. Os componentes centrais do OMed são: 1) o componente de interface, que recebe consultas em linguagem natural e realiza busca da informação, 2) o componente de representação do conhecimento, que agrega e traduz informações de um específico cenário para uma representação semântica adequada ao uso do OMeD e 3) o componente de raciocínio semântico, que deriva as respostas para uma consulta

do aprendiz por meio de um raciocínio sobre conhecimento médico relevante.

Em um dos experimentos realizados utilizando cinco conjuntos de dados contendo 1000 registros de pacientes e 20, 30, 40, 50 e 60 de medicamentos (metade foi utilizada como dados de treino) 100% de respostas corretas foram produzidas.

A Tabela 1, a seguir, dá a síntese das principais características dos projetos apresentados.

QA	Análise da Pergunta	Seleção e Extração	Geração das Respostas
FRyEA	Parser, raciocínio na ontologia, algoritmo que mede similaridade entre string, identifica o tipo da questão.	Gera e realiza consulta SPARQL, aprendizagem de máquina semi-supervisionada.	Web/textual
PowerAqua	Análise sintática, expressões regulares, Wordnet, algoritmo que mede similaridade entre string etc.	Consulta SPARQL, algoritmo para classificar a resposta etc.	Web/textual
OMeD	Processamento de linguagem natural, técnicas de representação do conhecimento e inferências.	Inferências e Técnicas de aprendizagem de máquina	-

Tabela 1. Sistemas pergunta-resposta.

3.1 Conclusões da Seção

Nesta seção mostraremos as diferenças conceituais entre o sistema proposto neste artigo e os sistemas referenciados no estado da arte. A seguir elencamos os seguintes pontos:

- De forma geral as ontologias nos sistemas pergunta-resposta são utilizadas como fontes das respostas [15], mas para perguntas que não necessitam respostas singulares, exemplo: O que são sistemas multitarefa? Utilizando apenas as ontologias é difícil responder perguntas que exigem definições ou explicações. A ontologia na arquitetura proposta neste artigo funciona como uma parte do cérebro do sistema. A partir de um conceito presente na pergunta é possível aplicar inferências lógicas para extração de novos conceitos e, dessa forma, enriquecer a consulta a fim de melhorar a extração da resposta.
- Normalmente os QASs iniciam todo processo de resolução da pergunta ao invés de verificar se a pergunta já foi resolvida anteriormente, buscando em alguma base de conhecimento. A classe de sistema que costuma utilizar essas práticas são os *helpdesks* [20], mas poucos utilizam bases próprias para *chatbots*. Na arquitetura proposta neste artigo é utilizada uma base AIML, própria para *chatbot*, que é construída à medida que as perguntas são resolvidas. Dessa forma a base pode ser aproveitada para um *chatbot* in-

teragir com o aprendiz, fornecendo a resposta. Além disso, antes da resposta ser incluída na base AIML especialistas do domínio confirmam a veracidade da resposta. Assim, a base AIML é expandida.

- Muitos sistemas na literatura retornam a resposta sem antes verificar e testar, ou seja, verificar se a resposta é realmente válida para perguntas. Alguns poucos sistemas até realizam essa verificação, mas poucos utilizam a técnica RTE (Seção 6.5). Na arquitetura proposta neste artigo utilizamos a técnica RTE para validação da resposta. Essa técnica é bastante consolidada e apoia-se por mostrar bons resultados [16, 18, 30].

- A arquitetura proposta também utiliza um recurso denominado Word Sense Disambiguation [21] para buscar os sinônimos corretos para o contexto da pergunta. Dentro do domínio de WSD utilizamos a técnica de aprendizagem de máquina supervisionada.

4. Descrição Detalhada do Problema

Observe a seguinte situação hipotética: Realizar uma consulta no Google com a frase “Quais são os sistemas operacionais que são multitarefas?”. Ainda que pareça simples, não encontraremos a resposta ainda que ela possa ser encontrada nos documentos disponíveis na

Web. Mas, os motores de busca (ex: Google) disponíveis podem não localizá-la por não implementarem raciocínio ou inferências. *Linux* é um sistema multitarefa, mas nenhum documento da Web menciona isto. No entanto, existem documentos que mencionam o seguinte: *Linux* é um sistema operacional (1), *Linux* compartilha seus recursos com aplicativos e aprendizes (2), Sistemas operacionais que compartilham recursos são multitarefas (3). Sistemas de recuperação de informação que utilizam raciocínio lógico (exemplo: ontologias) são construídos para responder a estes, e tipos similares, de perguntas [15].

Tomemos outro exemplo: (1) Qual país foi visitado pelo Papa em 1960? As palavras chaves são: “país”, “Papa”, “visitado”, “1960”. Nenhuma dessas palavras denota um país particular (tal como “Reino Unido”, ou “Estados Unidos”), ou “Papa” (chefe da igreja católica, por exemplo), ou a data dentro do intervalo de 10 anos, entre 1960 e 1970. Um conjunto muito mais complexo de palavras chaves é necessário a fim de se aproximar do resultado pretendido. E, para piorar, experiências mostram que pessoas não aprendem facilmente como formular e usar esse conjunto [10].

Os QASs podem melhorar uma consulta, expandindo-a para incluir mais termos do que as entradas inicialmente sugeridas. Dessa forma, eles facilitam a busca de respostas relacionadas. Por exemplo: um sistema pode expandir uma consulta de “carro” para procurar pelo sinônimo “auto” e a especialização “esportivo”. O uso de ontologias pode ajudar a analisar, refinar as consultas, entre outras utilidades [22].

5. A Arquitetura Proposta

A arquitetura proposta [28, 29, 32] difere-se das analisadas pela sua generalidade e também por combinar ontologias com uma família de agentes de software. Com o uso desses agentes pretende-se flexibilizar a arquitetura, favorecendo assim o uso de mecanismos inteligentes específicos, sintonizados com as necessidades de cada ação do sistema. Esperamos com isto obter um *framework* capaz de viabilizar a construção de QASs mais abrangentes e com respostas mais precisas.

A especialidade da arquitetura proposta (Figura 1) é resolver perguntas formuladas na língua inglesa do tipo WH-Question (O que, Quem, Quando, Onde, Quais). Ela é constituída de agentes que possuem perfis aderentes aos seus objetivos, ou seja: perfil para análise da pergunta, perfil para a seleção e extração da resposta, perfil para a construção do conhecimento e perfil para a geração da resposta. A Figura 1 também nos mostra dois tipos

gráficos de agente, apenas com a intenção de demarcar duas áreas funcionalmente diferentes. Os óculos representam agentes construtores do conhecimento (expert, ontology builder etc.). Os agentes sem óculos são os agentes responsáveis por resolver a pergunta (seeker, searcher etc.).

A arquitetura possui uma base de conhecimento, implementada em AIML, contendo pares pergunta-resposta.

5.1 Os Agentes de Software

Os perfis dos agentes serão descritos a seguir.

1. *Perfil de Análise da pergunta*: É responsável por examinar a pergunta em linguagem natural e gerar uma consulta (pergunta analisada) que possibilite a seleção dos documentos candidatos a responder a pergunta. As principais atividades realizadas pelo agente neste módulo são: extrair as palavras chaves, lematizar as palavras, remover as palavras irrelevantes, enriquecer a pergunta com sinônimos da *Wordnet*, realizar inferências nos conceitos da ontologia, resolver o tipo semântico da pergunta e obter a pergunta resolvida do banco AIML. Esse perfil é característico do agente “Recuperador”.
2. *Perfil de Seleção e extração das respostas*: O agente é responsável por selecionar os documentos candidatos da *Web* e extrair as respostas. É realizada uma verificação na resposta candidata a fim de torná-la uma resposta factível de retorno (implicação textual). Esse perfil é característico do agente “Buscador”.
3. *Perfil de Construtor do Conhecimento*: O agente é responsável por organizar e gerenciar consultas à ontologia, à base AIML e à *Web*. Além disso, o agente possibilita uma maior dinamicidade no sentido em que cada base de conhecimento é alimentada sempre que novas informações ou conceitos são encontrados. Adiciona-se a isso uma forma independente de trabalho, ou seja, enquanto uma parte dos agentes está resolvendo a pergunta, outra parte atualiza as bases de conhecimento. Esse perfil é característico dos agentes “Apoio Aprendizagem”, “Aprendizagem” e “Especialista”.
4. *Perfil de Geração das respostas*: Gera a resposta em um formato textual adequado ao entendimento do aprendiz. Para trabalhos futuros pretendemos ampliar a interoperabilidade com outros sistemas, ou seja, permitir a troca de perguntas e respostas com outros sistemas por meio de serviços semânticos. Esse perfil é característico do agente “Respondedor”.

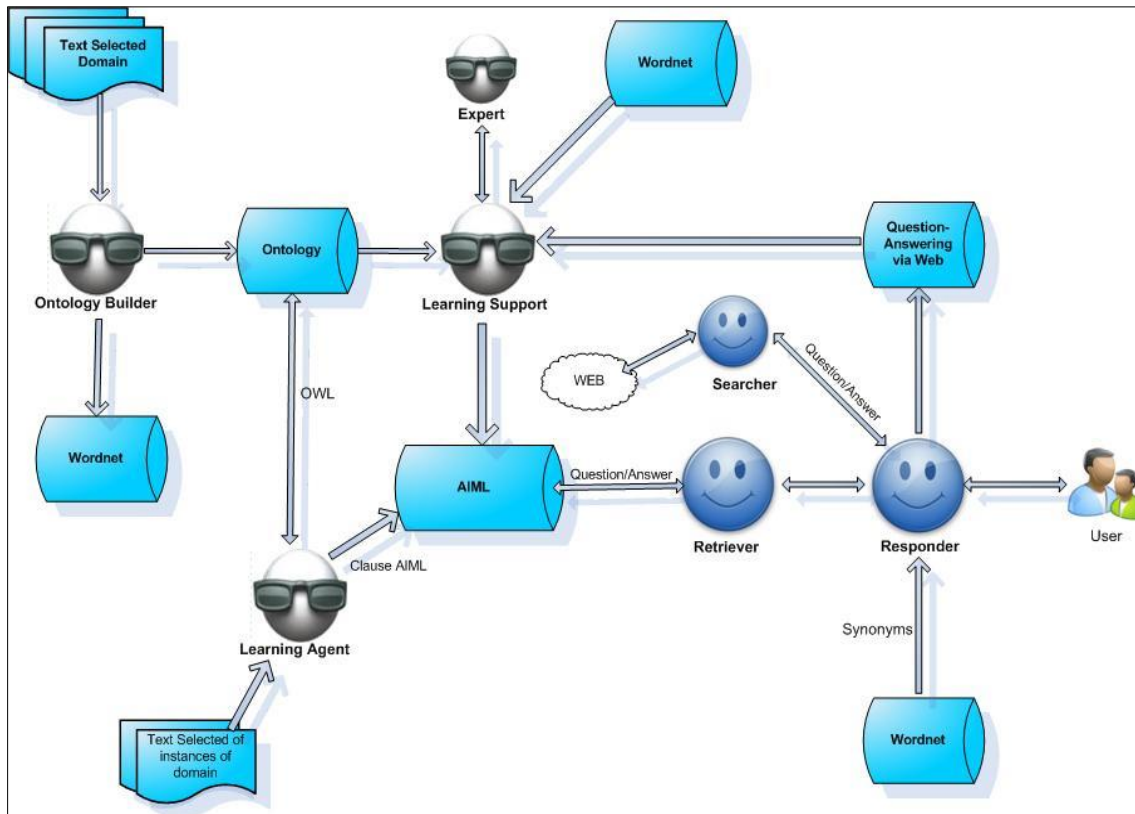


Figura 1. A arquitetura conceitual do sistema de pergunta-resposta.

6. O Protótipo

Para validar a arquitetura foi implementada uma versão preliminar que utiliza apenas um único agente de software. Nosso intuito, num primeiro momento, é apenas verificar a validade do uso das ontologias em sistema de pergunta-resposta. As subseções seguintes mostrarão a ontologia desenvolvida para sistemas operacionais, aspectos tecnológicos do protótipo e alguns experimentos realizados com seus resultados.

6.1 O Funcionamento

A seguir, descreveremos todas as atividades e passos envolvidos no funcionamento do protótipo desenvolvido neste trabalho.

1. O sistema recebe a pergunta em linguagem natural e inicia análise a fim de formar uma consulta para o módulo de seleção e extração das respostas. A análise da pergunta está intimamente relacionada ao campo de pesquisa denominado Expansão Automática da Consulta (AQE – *Automatic Query Expansion*) [23]. No módulo de análise da pergunta são realizadas as seguintes ações: a) As *stopwords* são eliminadas da pergunta, uma vez que são insignificantes para a recuperação de informação; b) Os conceitos da ontologia são identificados na pergunta por meio de um reconhecedor de entidades. Esse reconhecedor é um dicionário preenchido automaticamente com conceitos da ontologia; c) As palavras simples (que não são conceitos da ontologia) são lematizadas.
2. Por meio do agente “Recuperador”, a pergunta analisada é consultada na base AIML.
3. Se a resposta for encontrada na base AIML, então a resposta é retornada ao aprendiz e o sistema finaliza. Se não for encontrada é iniciado o processo de resolução da pergunta. A base AIML se pré-configura como uma memória, ou seja, armazena as perguntas que foram resolvidas para utilizá-las novamente quando for necessário.
4. O primeiro passo na resolução da pergunta é etiquetar as palavras simples (que não são conceitos da ontologia) com a classe gramatical correspondente. A etiqueta será útil para seleção dos substantivos nos passos consecutivos do sistema.
5. Os conceitos são enviados para a ontologia, a fim de raciocinar e retornar outros conceitos subentendidos. Tomemos como exemplo o domínio de sistemas

operacionais: se o conceito “multitarefa” for enviado para a ontologia, um dos conceitos produzidos pela inferência (raciocínio) é “Linux”. Isso ocorre porque existem as seguintes sentenças lógicas que permitem a inferência: (1) Linux é um sistema operacional, (2) Linux compartilha seus recursos com vários aplicativos, (3) Sistemas Operacionais que compartilham recursos são multitarefa. Esses conceitos subentendidos permitem a expansão da consulta e uma melhora na extração da resposta. A consulta é então formada com: (1) conceitos alvos do domínio, (2) palavras significativas, mas que não são conceitos, (3) conceitos da ontologia, (4) sinônimos.

6. Outra forma de expandir a consulta é por meio da seleção dos sinônimos, os quais serão utilizados para melhorar a recuperação de informação. São selecionados os substantivos, pois são as palavras com maior peso caracterizador sobre as coisas do mundo real. Nessa fase é estabelecida a interpretação singular do sentido da palavra, ou seja, buscamos os sinônimos coerentes com o contexto da palavra. Por exemplo: o substantivo “função” possui inúmeros significados na *Wordnet*, mas para a pergunta “Quais são as funções básicas de um sistema operacional?”, a palavra “função” possui entradas bem específicas.
7. O próximo passo é resolver o tipo semântico da pergunta. Isto é realizado por meio de padrões encontrados na pergunta. O tipo semântico será útil para formatar a resposta. Por exemplo: Para a pergunta: Quem inventou o Linux? O tipo semântico é uma Pessoa. Então a resposta deve ter um formato compatível com o tipo semântico.
8. A pergunta analisada, ou consulta expandida, é enviada para a *Web* com o fim de selecionar as páginas candidatas.
9. As páginas candidatas são transformadas em documentos passíveis de extração, ou seja, toda formatação inútil é extraída, tais como: estilo CSS, HTML.
10. Os trechos da resposta são extraídos e classificados com uma pontuação (grau de relevância). Neste momento a resposta não é a final e sim uma resposta candidata.
11. As respostas candidatas passam por um processo de confirmação para verificar se são respostas factíveis. Esse processo de confirmação é realizado por meio de uma técnica de implicação textual (RTE - *Recognizing Textual Entailment*). O RTE é definido

como uma tarefa de determinação se um dado pedaço de texto T implica em outro pedaço de texto H, chamado Hipótese [16]. O RTE avalia cada resposta candidata em relação à pergunta analisada.

12. As respostas são enviadas para o agente “Respondedor” que as formata de acordo com o tipo semântico.
13. A resposta é retornada para o aprendiz.
14. Os pares perguntas-respostas, não resolvidas e resolvidas, são enviados para especialistas humanos. Para as perguntas resolvidas os especialistas deve avaliar o grau de qualidade da resposta. Para as perguntas não resolvidas os especialistas devem responder e solicitar que o sistema classifique e armazene na base AIML. Se uma pergunta receber várias respostas, o sistema deve identificar as diferenças léxicas e semânticas, e a partir disso armazenar as respostas no banco com um percentual de importância para cada uma delas.

6.2 A Ontologia

A ontologia de conceitos básicos de sistemas operacionais foi desenvolvida utilizando a ferramenta Protégé e a linguagem OWL. A ontologia que usamos foi desenvolvida para um domínio de conhecimento específico para servir como prova de conceito do modelo conceitual proposto. Escolhemos o domínio de conceitos básicos de sistemas operacionais por se tratar de um sistema já previamente desenvolvido e contendo uma base AIML inicial [24].

A linguagem OWL é baseada em conceitos matemáticos da lógica descritiva e por isso é possível realizar inferências. A Figura 2 ilustra um exemplo de inferência consistente gerada pela ontologia. É importante entender que as classes em negritos foram criadas manualmente no Protégé. E as classes em amarelo foram geradas pela máquina de inferência. A inferência na classe “Multitasking” gerou uma nova superclasse “typeoperatingsystem”.

Na Seção 6.3 veremos as evidências do benefício de utilizar inferências na ontologia em sistema de pergunta-resposta.

6.3 A Implementação

Na implementação do protótipo utilizamos a linguagem Java e vários outros componentes, tais como: *Chatterbean*, *Lingpipe*, *Jena*, *Lucene*, entre outros. A

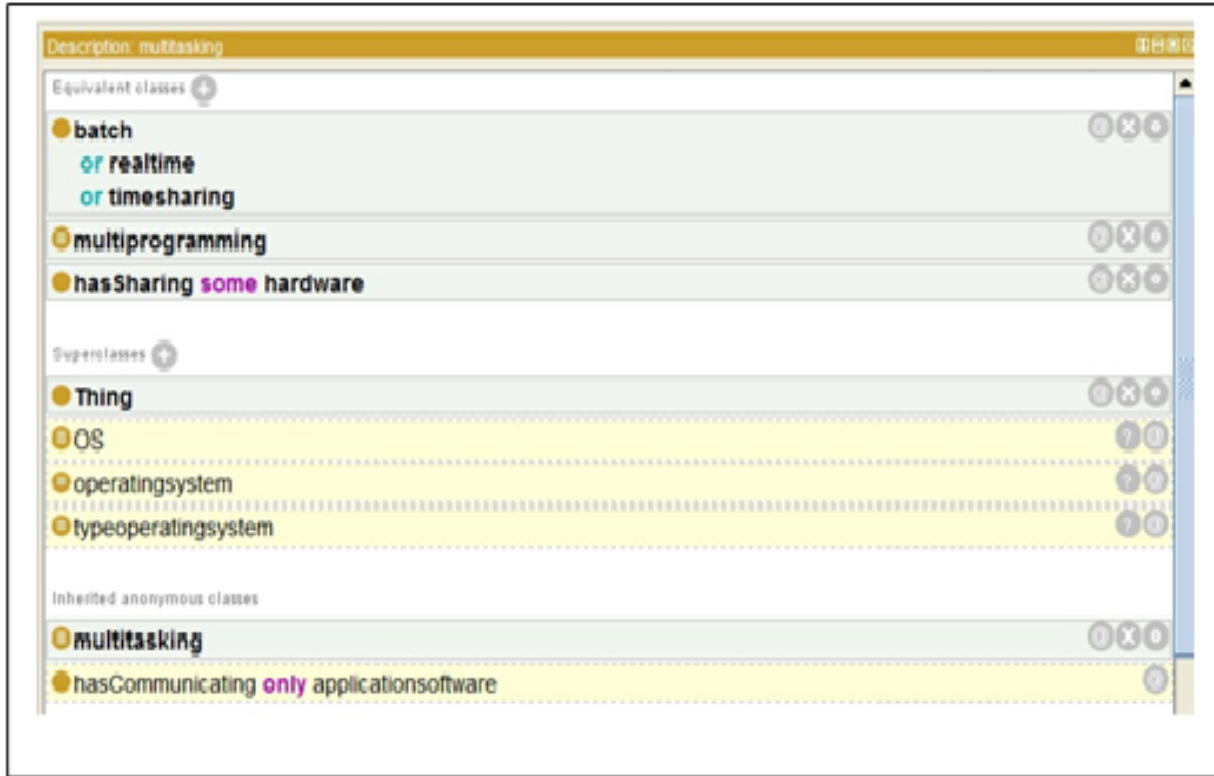


Figura 2. Inferência na ontologia.

Tabela 2 mostra uma síntese dos componentes tecnológicos da arquitetura com seus respectivos agentes. As explicações sumárias de cada um dos componentes podem ser vistas na Seção 6.5.

Agentes Responsável	Componente
“Recuperador”	Wordnet, Lucene, LingPipe, Stanford
“Buscador”	Venses, Jirs, Web-Harvest
“Recuperador” e “Aprendizagem”	ChatterBean, Pellet, Jena, AIML,
“Respondedor”	Expressão regular

Tabela 2 - Componentes Tecnológicos utilizados no protótipo corrente.

6.4 Os Experimentos

O experimento foi realizado com 60 perguntas selecionadas de livros educacionais de sistemas operacionais [25, 26] cerca de 5 perguntas de cada capítulo. Então a dispersão do conteúdo das perguntas se estende a mais ou menos 12 capítulos diferentes do contexto de sistema operacional. A estratégia foi simular as dúvidas comuns de estudantes da matéria de sistema operacional.

Nesse cenário foram obtidas perguntas sugeridas pelos livros texto usados nos programas dos cursos superiores em informática. Os resultados dos experimentos são registrados na Tabela 3.

Usamos as métricas comumente utilizadas nos QAS que são típicas em sistemas de recuperação de informação. As métricas são:

- *Precision* [27] - o número de questões respondidas corretamente, dividido pelo número de questões respondidas, e
- *Recall* [31] - o número de questões respondidas pelo sistema, dividido pelo número total de questões.

O termo “corretamente” está ligado à medida de recall/precision. Podemos ter perguntas respondidas, porém suas respostas estão ou parcialmente corretas ou não corretas. Entretanto, foram respondidas. Então, aquelas perguntas em que as respostas atendem totalmente a intenção do domínio ou do aprendiz são as corretamente respondidas. Os resultados são mostrados na Tabela 4.

Descrição	Quantidade	Percentual
Quantidade total de perguntas	60	100%
Quantidade de perguntas respondidas com Ontologia e RTE	45	75%
Quantidade de perguntas respondidas sem RTE	49	81,67%
Quantidade de perguntas respondidas sem ontologia	57	95%
Quantidade de perguntas respondidas corretamente com Ontologia e RTE	43	71,67%
Quantidade de perguntas respondidas corretamente sem RTE	35	58,33%
Quantidade de perguntas respondidas corretamente sem Ontologia	26	41,67%

Tabela 3 - Quantidade de perguntas testadas.

Descrição	Precision/Recall
<i>Precision</i> com Ontologia e RTE	0,955
<i>Recall</i> com Ontologia e RTE	0,75
<i>Precision</i> sem Ontologia	0,438
<i>Recall</i> sem Ontologia	0,95
<i>Precision</i> sem RTE	0,714
<i>Recall</i> sem RTE	0,816

Tabela 4 – *Precision e Recall*.

6.5 Descrição das Tecnologias Utilizadas

A seguir apresentamos as ferramentas de software livre, recuperadas da internet, utilizadas na construção do protótipo, com seus respectivos objetivos.

RTE - Recognizing Textual Entailment – grosso modo, é um resolvidor de ambiguidade textual. O RTE é definido como uma tarefa de determinação se um dado pedaço de texto T implica em outro pedaço de texto H, chamado Hipótese [16].

Wordnet – É uma base de conhecimentos linguísticos para a língua inglesa. O objetivo é consultar os sinônimos dos substantivos.

Lucene – É uma biblioteca Java para recuperação da informação. É uma biblioteca de pesquisa de texto completo em Java.

LingPipe – É uma biblioteca Java para processamento de textos usando linguística computacional. Os objetivos para a utilização deste componente são:

- Reconhecer os conceitos da ontologia (também chamados de reconhecimento de entidades).
- Escolher quais das entradas (sinônimos) encontradas na *Wordnet*, para um determinado substantivo, pertence ao contexto da pergunta.
- Etiquetagem sintática.

Stanford – É um conjunto de ferramentas para análise da linguagem natural. O objetivo é lematizar as palavras.

Venses – É um serviço que realiza implicação textual (RTE - *Recognising Textual Entailment*), ou seja, verifica

se uma resposta provável implica na pergunta analisada. A implicação realizada por esse componente é puramente sintática, ou seja, se baseia nas relações gramaticais para concluir se um pedaço de texto implica em outro texto.

Jirs – É uma biblioteca Java específica para recuperação de passagens. O objetivo é extrair as respostas prováveis a pergunta.

Web-Harvest – É uma biblioteca Java que localiza páginas na web e extrai seus conteúdos.

ChatterBean – É um interpretador da linguagem AIML. O objetivo é gerenciar a base de conhecimento AIML.

Pellet – É um raciocinador (*reasoner*) que checa a consistência e gera inferências nas bases de conhecimento OWL. O objetivo é realizar inferência e extrair novos conceitos importantes na extração da resposta.

Jena – É um conjunto de bibliotecas Java que permite manipular e salvar OWL. O objetivo é manipular OWL com consultas SPARQL.

AIML – O AIML ou, *Artificial Intelligence Markup Language* é uma linguagem que permite descrever o conhecimento que será manipulado por *chaterbots*. O objetivo é armazenar na base de conhecimento o par pergunta/resposta.

Expressão regular – A expressão regular é um padrão bastante geral construído por caracteres que permitem a recuperação de pedaços de textos correspondentes ao padrão. O objetivo é formatar a passagem extraída (provável resposta) de modo que se aproxime da resposta final.

7. Considerações Finais

Neste trabalho apresentamos um sistema de pergunta-resposta objetivando perguntas do tipo *WH-question* da língua inglesa. No desenvolvimento do projeto foram adicionadas ontologias e diversas outras técnicas úteis para aperfeiçoar a recuperação da resposta. Como percebido nos resultados dos experimentos mostrados na Tabela 2, o emprego das ontologias para expansão da consulta aprimora a recuperação da informação.

Encontramos na literatura [9, 26] outras pesquisas que utilizam a ontologia como meio para melhorar a

recuperação da informação bem como muitas das técnicas mencionadas. No entanto, poucos sistemas as utilizam todas em conjunto.

Nesse contexto nos perguntamos como introduzir ontologias para melhorar a recuperação? Para que ontologias aperfeiçoem sistemas de pergunta-resposta, é necessário, primeiro, estruturar os conceitos adequadamente e realizar inferências lógicas. Existem muitas formas para expandir a consulta utilizando ontologia, por exemplo, por meio de classes (conceitos) equivalentes, super/subclasses, negação de classes, mereologia (parte/todo). Por exemplo, a pergunta: “what are the types of the operating system?” por abranger outros domínios poderia gerar as respostas válidas: “32 bits”, “64 bits” ou “mobile”. No entanto, para o domínio de conceitos básicos de sistema operacionais a resposta deveria ser “batch, multitasking, monotasking”.

Para melhorar a extração da resposta, novos conceitos devem ser buscados na ontologia, tal como “multi-programming”. Buscar os conceitos corretos é importante para expandir a pergunta para o domínio pretendido. Por isso que o uso de ontologias melhora a quantidade de perguntas respondidas corretamente.

Para realizar uma comparação com outros trabalhos seria necessária uma base de dados com as mesmas características da nossa (ou pelo menos do mesmo domínio). Isso é bem difícil de conseguir. Chegamos a testar sistemas question-answering que prometiam responder perguntas de qualquer domínio. Quando jogamos algumas perguntas (exemplo de sistema que testamos: openephra) não funcionou perfeitamente. Isso é muito complicado, pois há uma intersecção semântica “estranha” entre domínios que leva a confusão muitos sistemas question-answering. Uma das soluções para isso talvez seja a interação com aprendizes por meio dos agentes do sistema.

A partir deste ponto do projeto temos várias linhas de trabalho a serem exploradas dos quais podemos destacar:

1. Dar prosseguimento à implementação da família de agentes que compõem o sistema, como mostramos na Figura 1.
2. Implantar o sistema para uso em situações reais, ou seja, criar bases de conhecimento em domínios específicos e disponibilizar o sistema para uso.
3. Implementar o envio dos pares perguntas-respostas para os especialistas decidirem o que fazer com eles.
4. É necessário um mecanismo para enriquecer a ontologia com novos conceitos, relações e instâncias do domínio a partir de textos da *Web*.

A inclusão de novos conceitos deve contar com o auxílio dos especialistas.

5. Ampliar a comunicação do sistema de pergunta-resposta com outros sistemas através da troca semântica de dados. Dessa maneira outros sistemas poderiam consumir o serviço de pergunta-resposta sem precisar conhecer o funcionamento interno do sistema.
6. Dar ao aprendiz a opção de emitir opinião sobre a qualidade da resposta.
7. Utilizar corretor ortográfico na interface de entrada da pergunta, possibilitando assim ao aprendiz a correção gramatical da pergunta. E, num momento mais à frente,
8. Desenvolver outra versão do protótipo no qual a base AIML seja gerada automaticamente a partir da ontologia. Em seguida, comparar os resultados obtidos com as duas versões.

Referências

- [1] Castells, M.A sociedade em rede. Tradução Roneide Venancio Majer. 3. ed. São Paulo: Paz e Terra, V.1, 2000.
- [2] Agissilaos, A. (2005). Ontologies and Query expansion. Master of Science School of Informatic, University of Edinburgh.
- [3] Vicedo, J. L.; Mollá, D. (2007). Open-Domain Question-Answering State of the Art and Future Trends OU Question Answering in Restricted Domain: An Overview, Journal Computational Linguistics, Volume 33, Issue 1, MIT Press.
- [4] Akerkar, R. A.; Sajja P. S. (2010). Knowledge-Based system, capítulo Natural Language Interface: Question Answering System, pag. 323-330. Jones and Barlett Publishers.
- [5] Menezes, C.S., Nevado, R.A., Castro, A.N.Jr. e Santos, L.N. (2008). MORFEU – Multi-Organizador Flexível de Espaços Virtuais para Apoiar a Inovação Pedagógica em EAD. Simpósio brasileiro de Informática na Educação. Fortaleza – CE. Anais do XVI SBIE.
- [6] Rangel, V.G; Cury, D.; Menezes C.S.; Beltrame, W.A.R. (2010) “Um Ambiente Para Construção de Veículos de Comunicação: da Modelagem de Sites Interativos ao Apoio às Arquiteturas Pedagógicas”. V WAPSEDI, João Pessoa – PB.
- [7] Menezes, C. S. e Tavares, O. L. (2009) “Veículo de Comunicação - Uma abordagem para o desenvolvimento rápido de Recursos Digitais para

- elaboração de Arquiteturas Pedagógicas”. IV WAPSEDI, Florianópolis – SC.
- [8] Monteiro, V. C. P. C., Menezes, C. S., Nevado, R. A., Fagundes, L. C. (2005) “Ferramenta de Autoria e Interação para apoio ao desenvolvimento de Projetos de Aprendizagem”. *Renote Revista Novas Tecnologias na Educação* V3, v. 3, n. 2.
- [9] Ferrucci, D. A.; Kalyanpur, A. A; Murdock, W. J.; Welty, C. A.; Zadrozny, W. W. (2012). Using ontological information in open domain type coercion, Patent Application Publication, United States.
- [10] Clark, A.; Fox, C.; Lappin S. (2010). *The Handbook of computational linguistics and natural language processing*, capítulo Question and Answering, pag. 630-654. Wiley-Blackwell.
- [11] Burhans, D. T. (2002). A question-answering interpretation of resolution refutation, capítulo Introduction: Question Answering, Doctoral Dissertation, pag. 1-2. University at Buffalo, New York, USA.
- [12] Noy, N. F.; McGuinness, D. L. *Ontology Development 101: A Guide to Creating your First Ontology*, Stanford University, 2001.
- [13] Balduccini, M.; Baral, Baral, C.; Lierler, Y. (2008). Knowledge representation and question answering, *Handbook of Knowledge Representation*, Chapter 20, Elsevier.
- [14] Forbus, K. D. (2008). Qualitative Modeling, *Handbook of Knowledge Representation*, Chapter 9, Elsevier.
- [15] Damijanovic, D.; Agatonovic, M.; Cunningham H. (2010). Natural language interfaces to ontologies: combining syntactic analyses and ontology-based lookup through the user interaction, In *Proceedings of the 7th extended semantic web conference*, Crete, Greece, pag. 106-120.
- [16] Iftene, A. (2009). Textual Entailment, Romania: Phd Thesis, Computer Science, University of Iasi.
- [17] Doucette, J. A.; Khan, A.; Coher, R. (2012). A comparative evaluation of an ontological medical decision support system (OMeD) for critical environments, In *Proceedings of th 2nd ACM SIGHIT International Health Informatics Symposium*, pag. 703-708, New York, USA.
- [18] Ferrández, O.; Spurk, C.; Kouylekov, M.; Dornescu, I; Ferrádez, S.; Negri, M.; Izquierdo, R.; Tomás, D.; Orasan, C.; Neumann, G.; Magnini, B.; Vicedo, J. L. The QUALL-ME Framework: A Specifiable-Domain multilingual question answering architecture, capítulo Question Answering via Web Services Composition, pag. 29-37. Springer Berlin Heidelberg, 2011.
- [19] Lopez, V.; Fernández, M; Stieler, N.; Motta, E.(2011). Discovering authorities in question answer communities by using link analysis, In *Journal Web Semantic*, Disponível em: <<http://www.semantic-web-journal.net>>.
- [20] Wang, D.; Li, T.; Zhu, S.; Gong, Y. iHelp: An Intelligent Online Helpdesk System, *IEEE Transactions on systems, man, and cybernetics*, 2011.
- [21] Navigli, R.; Faralli, S.; Soroa, A.; Lacalle, O.; Agirre, E. Two birds with one stone: learning semantic models for text categorization and word sense disambiguation, In *Proceedings of the 20th ACM international conference on Information and knowledge management*, New York, USA, 2011.
- [22] McGuinness, D. L. (2004). Question answering on the semantic Web, In *Journal IEEE Intelligent Systems*, Volume 19, Issue 1, pag. 82-85.
- [23] Carpineto, C.; Romano, G. (2012). A survey of automatic query expansion in information retrieval, *Journal ACM Computing Surveys*, Volume 44, Issue 1, New York, USA.
- [24] Teixeira, S. (2005). Chatterbots – Uma proposta para a construção de bases de conhecimento. Dissertação de Mestrado apresentado a Programa de Pós-Graduação em Informática do Centro Tecnológico, Universidade Federal do Espírito Santo.
- [25] Machado, F. B.; Maia, L. P. (2008). *Arquitetura de Sistema Operacionais*, Editora LCT, 4 edição.
- [26] Tanenbaum, A. S.; Woodhull, A. S. *Operating systems: design and implementation*, Editora Pearson Prentice Hall, 3 edição, Universidade da Califórnia , 2006.
- [27] Unger, C.; Cimiano, P. (2011). Representing and resolving ambiguities in ontology-based question answering, In *Proceedings of the TextInfer 2011 Workshop on Textual Entailment*, pag. 40-49.
- [28] Amorim M.T., Cury D., Menezes C.S., 2011(a). Um sistema inteligente baseado em ontologia para apoio ao esclarecimento de dúvidas. Em SBIE2011, Simposio Brasileiro de Informática na Educação. Aracaju-SE.
- [29] Amorim M.T., Cury D., Menezes C.S., 2011(b). Uma abordagem arquitetônica para um sistema pergunta-resposta. Em Conferencia IADIS Ibero Americana. Rio de Janeiro.



- [30] Dagan, I.; Dolan, B.; Magnini, B.; Roth D. Recognizing textual entailment: Rational evaluation and approaches Metasearch, Natural Language Engineering, Cambridge University Press, 2009.
- [31] Popescu, A-M.; Etzioni, O.; Kautz, H. Toward a theory of natural language interfaces to databases -, In Proceedings of the 8th international conference on intelligent user interfaces, pag. 149-157, New York, USA, 2003
- [32] Amorim M.T., Cury D., Menezes C.S., 2012(c). Um Helpdesk Inteligente baseado em Ontologias. Em SBIE2012, Simpósio Brasileiro de Informática na Educação. Rio de Janeiro-RJ.