

# Um Helpdesk Inteligente baseado em Ontologias

Marta Talitha C. F. de Amorim<sup>1</sup>, Davidson Cury<sup>1</sup>, Crediné S. Menezes<sup>1</sup>

<sup>1</sup>Departamento de Informática – Universidade Federal do Espírito Santo (UFES)  
Av. Fernando Ferrari, 514, Campus de Goiabeiras – 29.075-910 – Vitória – ES – Brasil

{martatcfa, dedecury, credine}@gmail.com

**Abstract.** *The tasks of solving question or clarify 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-answering system that uses ontologies and information retrieval techniques in the question analysis and thus improves the extraction of the response.*

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

**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. Esse 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.*

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

## 1. Introdução

Nos últimos anos a internet tem acumulado uma incontável quantidade de documentos, tornando-se uma importante fonte potencial para esclarecer dúvidas. Entretanto, seu uso para esta finalidade tem de lidar com a dificuldade que os usuários têm em formular suas questões em linguagem compatível com o paradigma das máquinas de busca. Até mesmo usuários mais experientes geralmente não conseguem formular consultas boas quando trabalham com domínios dos quais têm pouco conhecimento [Agissilaos *et al.* 2005].

Para lidar com esse problema, uma possível abordagem é utilizar sistemas de perguntas-resposta por intermédio do qual um computador responda às perguntas, formuladas em linguagem natural, automaticamente. Sistemas de pergunta-resposta são especialmente úteis em situações nas quais o usuário precisa saber uma parte muito específica de informação e não dispõe de tempo para ler toda a documentação disponível relacionada ao tópico sob pesquisa, para resolver o problema [Vicedo *et al.* 2007].

De modo geral, o desafio de um sistema de pergunta-resposta consiste em retornar a resposta que mais se aproxima da expectativa do usuário para uma pergunta sua, 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 [Akerkar et al. 2009].

Este artigo apresenta um sistema de pergunta e resposta que utiliza ontologias, entre outras técnicas, com a finalidade de aperfeiçoar a extração da resposta para o domínio de sistemas operacionais. O artigo está organizado da seguinte forma: Na Seção 2 mostramos a arquitetura do sistema proposto. A Seção 3 descreve todas as atividades envolvidas no processo de resolução da pergunta. Por fim, na Seção 4 citamos trabalhos futuros e tecemos algumas conclusões temporárias sobre o trabalho realizado.

## 2. Embasamento Teórico-metodológico

Nossa proposta é pesquisar as implicações do uso de inferências em ontologias de domínio para enriquecimento na extração da resposta em um sistema de pergunta-resposta.

Balduccini *et al.* (2008) classificou os sistemas de pergunta-resposta que incorporam representação do conhecimento e raciocínio (ou inferência) baseado em três abordagens: baseado em lógica formal, extração de informação e usando lógica formal na extração da informação. Todas as três abordagens em algum momento utilizam linguagens lógicas para extrair algum novo conhecimento inferido. Sistema que utilizam ontologia para raciocinar e representar conhecimento são conhecidos por utilizar modelagem qualitativa<sup>1</sup> (ou raciocínio qualitativo). Um dos objetivos da modelagem qualitativa é tornar o conhecimento explícito em tácito, fornecendo formalismos [Forbus, 2008].

Além de representar conhecimento, a ontologia pode ser útil para [Noy *et al.*, 2001]: a) Compartilhar entendimento comum da estrutura da informação entre pessoas ou entre agentes; b) Permitir reutilizar conhecimento do domínio; c) Fazer hipóteses explícitas do domínio; d) Separar conhecimento do domínio do conhecimento operacional; e) Analisar conhecimento do domínio.

Muitos trabalhos na literatura têm utilizado ontologias em seus diversos aspectos e por meio dos experimentos tem apresentado ganhos. No próximo capítulo detalharemos alguns desses trabalhos.

## 3. Trabalhos Correlatos

Esta pesquisa bibliográfica faz uma explanação dos sistemas de pergunta-resposta mais atuais que utilizam ontologias.

O *FREyA* [Damijanovic *et al.* 2010] traduz uma pergunta em linguagem natural ou palavras chaves em uma consulta SPARQL<sup>2</sup>, e retorna a resposta para o usuário depois de executar uma consulta na ontologia. A dinâmica do sistema pode ser resumida

---

<sup>1</sup> Modelagem qualitativa (ou raciocínio qualitativo) – Conceito matemático da área de representação do conhecimento.

<sup>2</sup> SPARQL é uma linguagem de consulta para RDF. <http://www.w3.org/TR/rdf-sparql-query/>

nos seguintes passos: Identifica e verifica os conceitos da ontologia, gera a consulta SPARQL e identifica o tipo da resposta e apresenta o resultado para o usuário. 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 usuário. Ao envolver o usuário 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 com os termos da ontologia. Se existir anotações ambíguas na consulta, é realizado um diálogo com o usuário. Os testes obtiveram um recall de 92.4% sobre um total de 250 questões.

O *PowerAqua* [Lopez et al., 2011] é uma evolução de outro sistema chamado *Aqualog*, um sistema baseado em ontologia. Na arquitetura do *PowerAqua*, o componente análise da questão utiliza um componente linguístico para processar a consulta. A saída desse componente é um conjunto de triplas linguísticas (< sujeito, predicado, objeto>) que é mapeado para a consulta do usuário. Assim é possível realizar buscas das respostas em bases OWL [OWL]/RDF[RDF]. Os resultados obtidos nos testes apresentaram 48 (69,5%) questões respondidas das 69 questões totais.

O *OMeD* [Doucette et al., 2012] é um sistema que apoia na tomada de decisões médicas por meio da qual respostas são fornecidas em tempo real para perguntas médicas. Os componentes centrais do *OMeD* são: (i) O componente de interface, que recebe consultas em linguagem natural e realiza busca da informação. (ii) O componente de representação do conhecimento, que agrega e traduz informações de um específico cenário para uma representação semântica adequada ao uso do *OMeD*. (iii) O componente de raciocínio semântico, que deriva as respostas para uma consulta do usuário 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) foi produzido 100% de respostas corretas.

#### 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 usuários (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 [Damijanovic et al. 2010].

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 aprenderão como formular e usar esse conjunto [Clark et al. 2010].

Os sistemas de pergunta-resposta podem melhorar uma consulta expandindo-a para incluir mais termos do que as entradas inicialmente sugeridas e, dessa forma, facilitar na 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 [McGuinness, 2004].

## 5. Proposta de Solução

A especialidade da arquitetura proposta (Figura 1) é resolver perguntas, formuladas na língua inglesa, do tipo *WH-Question* (O que, Quem, Quando, Onde, Quais, Quem). Ela é constituída de quatro módulos, a saber: análise da pergunta, seleção e extração da resposta, uma base de conhecimento, implementada em AIML (*Artificial Intelligence Markup Language*), contendo pares perguntas-respostas previamente conhecidas, e um módulo de geração da resposta. A interação funcional dos módulos é realizada por uma agente de software. Cada módulo possui objetivos bem definidos e que detalharemos a seguir.

1. Análise da pergunta: Este módulo 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 [AIML].

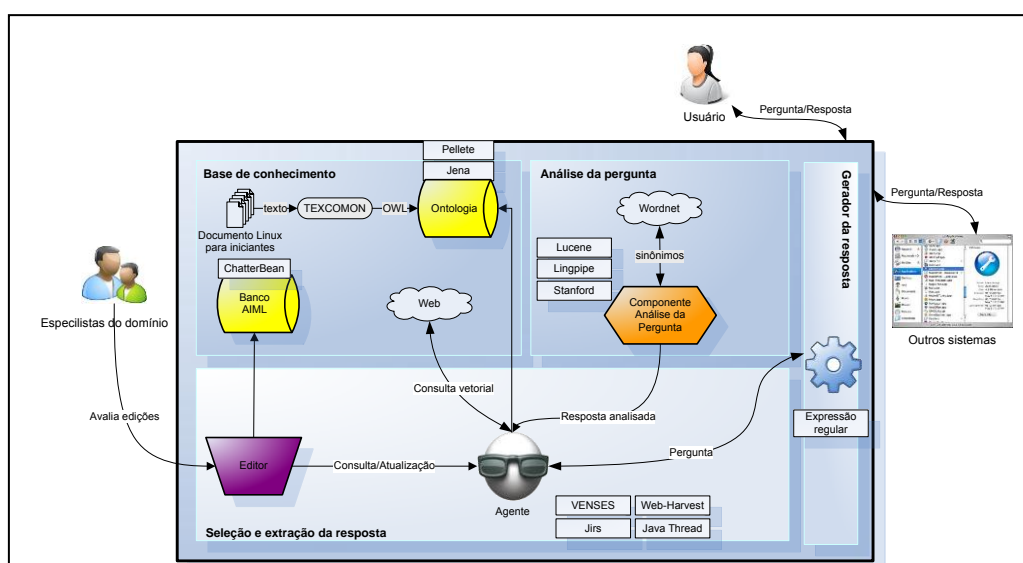


Figure 1. A arquitetura do sistema de pergunta-resposta

2. Seleção e extração das respostas: Neste módulo o agente é responsável por selecionar os documentos candidatos da *Web* e extrair as respostas, tarefas que se baseiam na pergunta analisada. É realizada uma verificação na resposta candidata a fim de torná-la uma resposta factível de retorno (implicação textual). O agente nesta arquitetura se comporta como um gestor das atividades executadas no módulo de seleção e extração da resposta. Assim que uma pergunta é enviada, o agente percebe que é necessário coordenar as atividades de resolução da resposta.
3. Base de conhecimento: Aqui o agente é responsável por organizar e gerenciar consultas à ontologia, à base AIML e à *Web*. A base AIML foi estendida do trabalho desenvolvido por Teixeira (2005) com o fim de aprimorar o tempo de retorno da resposta.
4. Geração das respostas: Este módulo gera a resposta em um formato textual adequado ao entendimento do usuário. Para trabalhos futuros pretendemos tornar esse módulo interoperável com outros sistemas, ou seja, permitir a troca de perguntas e respostas com outros sistemas por meio de serviços semânticos.

A seguir, descreveremos todas as atividades e passos envolvidos no funcionamento da arquitetura proposta 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*) [Carpineto *et al.* 2012]. 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 de um agente, a pergunta analisada é consultada no banco AIML.
3. Se a resposta for encontrada na base AIML, então a resposta é retornada ao usuário e o sistema finaliza. Se não for encontrada é iniciado o processo de resolução da pergunta. A base AIML se preconfigura 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. Por exemplo: 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 formada com: (1) conceitos alvos do domínio, (2)

palavras significativas, mas que não são conceitos, (3) conceitos subtendidos, (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 [Iftene, 2009] O RTE avalia cada resposta candidata em relação à pergunta analisada.
12. As respostas são enviadas para o módulo de geração da resposta que as formata de acordo com o tipo semântico.
13. A resposta é retornada para o usuário.
14. A resposta é enviada para um especialista do domínio para avaliação. De acordo com a pontuação da avaliação o sistema grava a resposta na base AIML por meio do editor. O editor é um componente que gerencia os acessos externos a base AIML. A atividade de avaliação do especialista é um trabalho futuro pretendido por este trabalho.

## 6. Prova de Conceito

Para validar a arquitetura foi implementada uma versão preliminar, usando a linguagem Java, apoiada por vários outros componentes, tais como: *Chatterbean*, *Lingpipe*, *Jena*, *Lucene*, entre outros. Desenvolvemos uma ontologia particular utilizando o *Protégé* [Protégé] e a linguagem OWL.

Uma síntese dos componentes tecnológicos da arquitetura pode ser encontrada na Tabela 1.

**Tabela 1. Componentes Tecnológicos da Arquitetura**

<b>Componente</b>	<b>Objetivo</b>	<b>Módulo</b>
Wordnet	É uma base de conhecimentos linguísticos para a língua inglesa. O objetivo é consultar os sinônimos dos substantivos.	Análise da pergunta
Lucene	É uma biblioteca Java para recuperação da informação. O objetivo é remover as <i>stopwords</i> .	Análise da pergunta
LingPipe	É uma biblioteca Java para processamento de textos usando linguística computacional. Os objetivos para a utilização deste componente são: <ul style="list-style-type: none"> <li>• Reconhecer os conceitos da ontologia (também chamados de reconhecimento de entidades)</li> <li>• Escolher quais das entradas (sinônimos) encontradas na <i>Wordnet</i>, para um determinado substantivo, pertence ao contexto da pergunta.</li> <li>• Etiquetagem sintática.</li> </ul>	Análise da pergunta
Stanford	É um conjunto de ferramentas para análise da linguagem natural. O objetivo é lematizar as palavras.	Análise da pergunta
Venses	É um serviço que realiza implicação textual (RTE - <i>Recognising Textual Entailment</i> ), 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.	Seleção e extração da resposta
Jirs	É uma biblioteca Java específica para recuperação de passagens. O objetivo é extrair as respostas prováveis a pergunta.	Seleção e extração da resposta
Web-Harvest	É uma biblioteca Java que localiza páginas na web e extrai o conteúdo.	Seleção e extração da resposta
ChatterBean	É um interpretador da linguagem AIML. O objetivo é gerenciar a base de conhecimento AIML.	Base de conhecimento
Pellet	É um raciocinador ( <i>reasoner</i> ) 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.	Base de conhecimento
Jena	É um conjunto de bibliotecas Java que permite manipular e salvar OWL. O objetivo é manipular OWL com consultas SPARQL.	Base de conhecimento
AIML	O AIML ou, <i>Artificial Intelligence Markup Language</i> é uma linguagem que permite descrever o conhecimento que será manipulado por <i>chaterbots</i> . O objetivo é armazenar na base de conhecimento o par pergunta/resposta.	Base de conhecimento
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.	Gerador da resposta

## 6.1. Experimentos

O experimento foi realizado com 60 perguntas selecionadas de livros educacionais de sistemas operacionais (Machado *et al.*, 2008) (Tanenbaum *et al.*, 2006). Os resultados dos experimentos são registrados na Tabela 2.

Tabela 2. Quantidade de perguntas testadas

Descrição	Quantidade	Percentual
Quantidade total de perguntas	60	100%
Quantidade de perguntas respondidas <b>corretamente</b> com Ontologia e RTE.	43	71,67%
Quantidade de perguntas respondidas <b>corretamente</b> sem Ontologia	26	41,67%

Para que ontologias aperfeiçoem sistemas de pergunta-resposta, é necessário estruturar os conceitos adequadamente. 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 “*multiprogramming*”. 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.

## 7. Conclusões e Trabalhos Futuros

Este trabalho desenvolveu 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. Os sistemas atuais de pergunta-resposta utilizam muitas dessas técnicas, mas poucos utilizam todas em conjunto.

Como percebido nos resultados do experimento da Tabela 2, o emprego das ontologias para expansão da consulta aprimora a recuperação da informação. Também encontramos na literatura [Ferrucci *et al.*, 2012; Unger *et al.*, 2011] outras pesquisas que utilizam da ontologia como meio para melhorar a recuperação da informação. 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. Por exemplo, a pergunta: “*what are type 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 “*multiprogramming*”. Existem muitas formas para expandir a consulta utilizando ontologia, por meio de classes (conceitos) equivalentes, super/subclasses, negação classes, mereologia (parte/todo).



Para os trabalhos futuros do sistema proposto, citamos:

1. Realizar experimentos reais, ou seja, criar uma base com perguntas e permitir que o sistema interaja com grupos de usuários.
2. O sistema deve enviar para os especialistas as perguntas não resolvidas e resolvidas. Para as perguntas resolvidas o especialista 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.
3. 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.
4. 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. Por exemplo: ambientes de aprendizagem virtuais poderiam utilizar o método de pergunta-resposta para testar o conhecimento do aluno.
5. Dar ao usuário a opção de emitir opinião sobre a qualidade da resposta.

## 8. Referências

- Agissilaos, A. (2005). Ontologies and Query expansion. Master of Science School of Informatic, University of Edinburgh, 2005.
- Akerkar, R. A.; Sajja P. S. (2010). Knowledge-Based system, capítulo Natural Language Interface: Question Answering System, pag. 323-330. Jones and Barlett Publishers.
- Balduccini, M.; Baral, Baral, C.; Lierler, Y. (2008). Knowledge representation and question answering, Handbook of Knowledge Representation, Chapter 20, Elsevier.
- 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.
- Clark, A.; Fox, C.; Lappin S. (2010). The: Handbook of computacional linguistics and natural language processing, capítulo Question and Answering, pag. 630-654. Wiley-Blackwell.
- 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.
- 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 2<sup>nd</sup> ACM SIGHIT International Health Informatics Symposium, pag. 703-708, New York, USA.
- 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.

Forbus, K. D. (2008). Qualitative Modeling, Handbook of Knowledge Representation, Chapter 9, Elsevier.

Iftene, A. (2009). Textual Entailment, Romania: Phd Thesis, Computer Science, University of Iasi.

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>>.

Machado, F. B.; Maia, L. P. (2008). Arquitetura de Sistema Operacionais, Editora LCT, 4 edição.

McGuinness, D. L. (2004). Question answering on the semantic Web, In Journal IEEE Intelligent Systems, Volume 19, Issue 1, pag. 82-85.

OWL, “Web Ontology Language”. Disponível em: <<http://www.w3.org/2004/OWL/>>. Acesso em: 17 de 07 de 2012.

Protege, “Open source ontology editor and knowledge-base framework.”. Disponível em: <<http://protege.stanford.edu>>. Acesso em: 17 de 07 de 2012.

RDF, “Resource Description Framework”. Disponível em: <<http://www.w3.org/RDF/>>. Acesso em: 17 de 07 de 2012.

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.

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.

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.

Wordnet, “A lexical database”. Disponível em: <<http://wordnet.princeton.edu/>>. Acesso em 24 de junho de 2011.