

Etiquetagem e rotulagem de frases no ambiente de apoio a disciplinas on-line “Assistente de Reuniões”

Carlos Alexandre Siqueira da Silva^{1,2}, Crediné Silva de Menezes¹

¹Departamento de Informática – Universidade Federal do Espírito Santo (UFES)
Campus de Goiabeiras – Vitória – ES – Brasil

²Núcleo de Informática – Instituto Federal do Espírito Santo (IFES)
Campus de Alegre – Alegre – ES – Brasil

cassilva@ifes.edu.br, credine@gmail.com

Abstract. *The intelligent layer called Meeting Assistant aims to support the authoring and assist in information retrieval and automating tasks in online meetings of distance learning disciplines, allowing participants to focus on the ongoing discussion. To achieve its goals, the Meeting Assistant is composed of a community of independent agents, working on various tasks. This paper presents the operation of the two main agents, the Tagger agent and the Labeler agent, and describes its configuration by the user to obtain the necessary flexibility to online meeting.*

Resumo. *A camada inteligente denominada Assistente de Reuniões tem o objetivo de apoiar a autoria e auxiliar na recuperação de informações e na automatização de tarefas em reuniões on-line de disciplinas de EAD, permitindo que os participantes se concentrem na discussão em curso. Para atingir seus objetivos, o Assistente de Reuniões é composto por uma comunidade de agentes independentes, atuando em diversas tarefas. Este artigo apresenta o funcionamento dos dois principais agentes, o agente Etiquetador e o agente Rotulador, e descreve a sua configuração pelo usuário para obter a flexibilidade necessária à reunião on-line.*

1. Introdução

A arquitetura apresentada por [Silva, 2012] subdivide-se em um VCom (Veículo de Comunicação) denominado Reunião On-Line, e uma camada inteligente denominada Assistente de Reuniões.

Esta arquitetura tem como objetivo auxiliar os usuários no processo de autoria, ajudar na recuperação de informações e automatizar tarefas mecânicas em um ambiente de reuniões on-line para fins educacionais. Como exemplos de reuniões on-line de cunho educacional, temos as disciplinas de um curso ministrado na modalidade de EAD, estudos dirigidos on-line, reuniões de orientação de monografias, dissertações e teses, e grupos de estudo em geral.

Para atingir esses objetivos, a arquitetura de software conta com uma comunidade de agentes utilizando recursos de processamento de linguagem natural, através da etiquetagem morfofossintática dos termos de uma frase; e uma interface *web*

onde é possível definir os padrões de mensagens que serão identificadas pelo sistema, as informações que serão armazenadas, e as regras de identificação que serão utilizadas.

Neste trabalho será apresentado o núcleo do Assistente de Reuniões, com foco em dois agentes específicos, e todo o processo de etiquetagem e rotulagem das mensagens enviadas pelos usuários. Será apresentada também a interface de configuração que permite ao usuário expandir o reconhecimento de mensagens do assistente, configurando-o de forma a melhor atender as especificidades da sua reunião on-line.

Nas seções 2 e 3 serão apresentadas a metodologia e o referencial teórico utilizado. A seção 4 trata do problema estudado, enquanto a seção 5 apresenta a solução proposta, descrevendo os agentes Etiquetador e Rotulador. A seção 6 faz uma breve análise dos resultados, e finalmente a seção 7 traz as conclusões e trabalhos futuros.

2. Metodologia

O presente trabalho é baseado em [Silva, 2012], onde são descritos a Reunião On-Line e o Assistente de Reuniões. No presente trabalho é enfatizada a atuação dos dois principais agentes da arquitetura, o agente Etiquetador e o agente Rotulador.

Para o desenvolvimento específico do agente Etiquetador, foi feita uma pesquisa bibliográfica sobre analisadores léxicos e morfossintáticos, e sobre etiquetadores de texto. Já para o desenvolvimento do agente Rotulador, foram pesquisados os temas de lógica de primeira ordem e programação em lógica, para a criação da base de fatos em Prolog; etiquetas sintáticas e semânticas para a separação dos termos; e expressões regulares em Java.

Em seguida foi desenvolvido um protótipo, para verificar o funcionamento do assistente de reuniões e sua comunidade de agentes. Esse protótipo foi configurado para o caso específico de um estudo dirigido on-line, e testado utilizando-se trechos de conversas reais de alguns usuários de reuniões on-line.

Por fim, foi feita uma análise sobre os agentes Etiquetador e Rotulador, com o objetivo de melhor documentar a sua atuação, detalhar o seu fluxo de informações, e identificar pontos de melhoria.

3. Referencial Teórico

Nesta seção são apresentados alguns conceitos que deram base ao presente trabalho – SMAs, processamento de linguagem natural, programação em lógica e o projeto VISL.

3.1. Sistemas Multiagente (SMA)

Segundo [Wooldridge, 2002], Sistemas Multiagente (SMA) são um grupo de agentes que interagem uns com os outros, tipicamente através da troca de mensagens através de uma infraestrutura de rede.

Sistemas multiagente podem ser vistos então como sociedades de agentes com autonomia para agir em busca de seus objetivos; que podem interagir e se comunicar uns com os outros, estabelecendo relações de dependência ou cooperação; e podem interagir com o ambiente individualmente ou em grupos.

O presente trabalho trata da arquitetura multiagente do Assistente de Reuniões, com sua comunidade de agentes independentes.

3.2. Processamento de Linguagem Natural (PLN)

O Processamento de Linguagem Natural (PLN) é um campo de pesquisa que surgiu da junção da inteligência artificial e da lingüística. O PLN estuda mecanismos para fazer com que o computador se comunique em linguagem humana, em seus diversos níveis. Segundo [Gonzalez & Lima, 2003], estes níveis são:

- 1) Fonético e fonológico: do relacionamento das palavras com os sons que produzem;
- 2) Morfológico: da construção das palavras a partir de unidades de significado primitivas e de como classificá-las em categorias morfológicas;
- 3) Sintático: Do relacionamento das palavras entre si, cada uma assumindo seu papel estrutural nas frases, e de como as frases podem ser partes de outras, constituindo sentenças;
- 4) Semântico: do relacionamento das palavras com seus significados e de como eles são combinados para formar os significados das sentenças;
- 5) Pragmático: Do uso de frases e sentenças em diferentes contextos, afetando seu significado.

Uma mensagem qualquer, enviada por um usuário humano em uma reunião online, precisa passar por diversas etapas de processamento, até que se obtenha a sua mensagem em uma forma lógica, que identifica o sentido (e os relacionamentos semânticos) das palavras e frases [Gonzalez & Lima, 2003]. Mais detalhes sobre a conceituação das formas lógicas pode ser encontrado em [Franconi, 1994].

3.3. Programação em Lógica e Prolog

Segundo [Palazzo, 1997], na programação em lógica, um algoritmo é constituído por dois elementos disjuntos: a lógica e o controle. O programador se preocupa com a lógica, definindo o que deve ser solucionado, enquanto a ferramenta se encarrega do controle, ou seja, como a solução pode ser obtida.

Prolog é uma linguagem de programação que segue o paradigma da programação em lógica. Seu nome é um acrônimo de “Programmation en Logique”, nome esse dado ao primeiro interpretador que utilizava a lógica como linguagem de programação, e que foi desenvolvido em 1972 por Alain Colmerauer e sua equipe.

3.4. O projeto VISL e o analisador “PALAVRAS”

[Bick, 2000] apresenta o “Palavras”, um analisador sintático para o idioma português que faz parte do projeto VISL – Visual Interactive Syntax Learning (VISL1, 2011) – iniciado em 1996 na Odense University (Dinamarca) e que tem como objetivo prover ferramentas visuais para o aprendizado sintático, segundo [Bick, 2001].

Ainda de acordo com [Bick, 2001], o VISL é pautado em quatro princípios básicos, que são: Flexibilidade, Interatividade, Naturalidade e Tutoria.

Neste trabalho é descrito o funcionamento do agente Etiquetador, que utiliza a interface “Complex Parsing” (VISL2, 2011) do ambiente VISL para realizar a etiquetagem morfofssintática das mensagens recebidas dos usuários.

4. Descrição do problema

O Assistente de Reuniões [Silva, 2012] possui uma comunidade de agentes independentes, cada um com seus objetivos e desempenhando suas atividades.

Dentre os agentes, existem dois cujas atividades norteiam o comportamento do Assistente de Reuniões como um todo. Esses agentes são responsáveis por tratar as mensagens que chegam ao sistema, e identificar os desejos dos usuários que as escreveram. Esses dois agentes são: o agente Etiquetador e o agente Rotulador.

Neste trabalho são descritos o funcionamento interno e as características desses dois agentes, bem como a sua configuração através da reunião on-line, para permitir que o sistema seja mais eficiente em suas tarefa de auxiliar os participantes de uma reunião educacional.

5. Os agentes Etiquetador e Rotulador

A Figura 1 mostra a arquitetura completa do sistema proposto por [Silva, 2012], com os agentes, bases de conhecimento, comunicações, integração com o VISL e interfaces entre o Assistente de Reuniões e o VCom Reunião On-line.

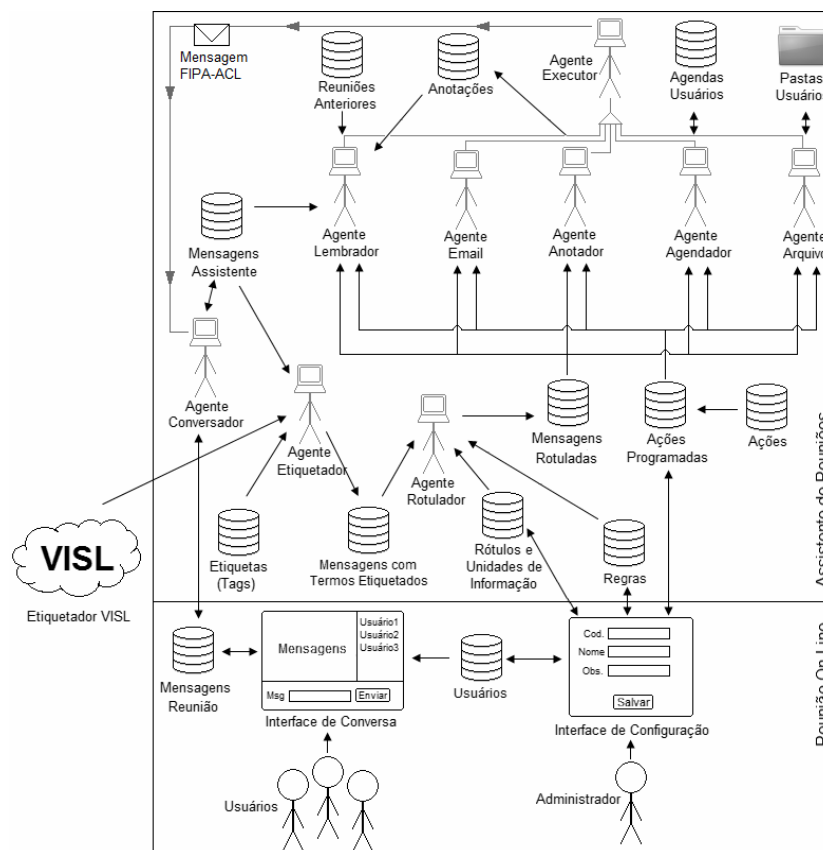


Figura 1 - Arquitetura completa do sistema proposto por [Silva, 2012]

Os agentes foram desenvolvidos na linguagem de programação Java, utilizando o framework para sistemas multiagente JADE [JADE, 2012] e banco de dados MySQL [MYSQL, 2012]. Já a interface web de configuração dos agentes foi desenvolvida na linguagem PHP [PHP, 2012], com servidor web Apache [APACHE, 2012] e banco de dados MySQL [MYSQL, 2012].

Na base de Mensagens do Assistente, temos as mensagens postadas pelos usuários no ambiente Reunião On-Line. Estas mensagens serão carregadas pelo agente Etiquetador, e enviadas para o etiquetador VISL. Esse envio ocorre por uma chamada remota ao site do VISL, passando a frase como parâmetro.

O agente Etiquetador recebe como resposta do VISL a mensagem original separadas em termos, e cada termo associado a uma ou mais etiquetas sintáticas e semânticas. A Figura 2 ilustra a comunicação do agente Etiquetador com o VISL, a resposta recebida, e como esses termos são armazenados no banco de dados. Na figura é possível notar que cada termo da frase recebeu classificações morfossintáticas do VISL ou do próprio agente Etiquetador (como é o caso do ponto final, que recebeu a etiqueta de “Pontuação”). Uma lista completa das etiquetas existentes pode ser vista em [VISL3, 2012].

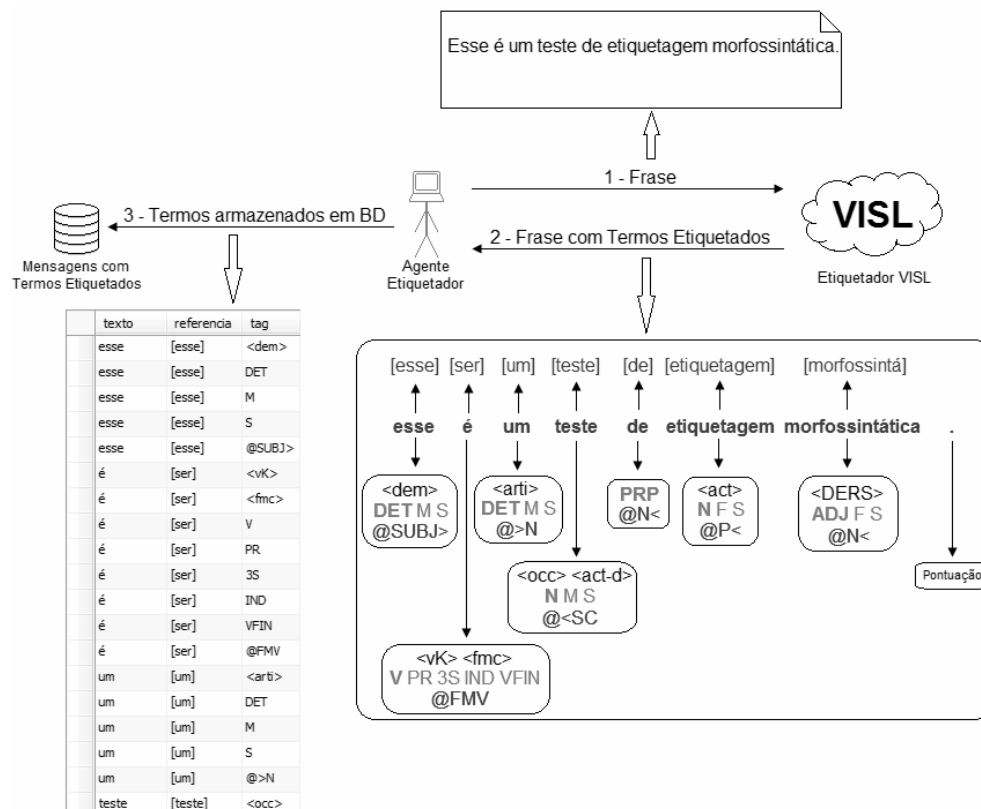


Figura 2 - Fluxo de informações do agente Etiquetador

A etiquetagem morfossintática é o primeiro passo para o tratamento das mensagens pelo Assistente de Reuniões. Após o agente Etiquetador tratar a resposta do VISL e separar os termos e etiquetas de cada frase, o agente Rotulador inicia seu processo de identificação das mensagens.

O agente Rotulador recebe as mensagens tratadas pelo agente Etiquetador (separadas em termos etiquetados) e busca um ou mais rótulos no sistema que se encaixem nessa mensagem. Cada rótulo possui unidades de informação (UIs) e regras. As UIs são partes da informação contida no rótulo, e as regras são expressões em Prolog que identificam o casamento de um rótulo com uma mensagem. Na Figura 3 é possível observar as interfaces da reunião on-line onde é possível consultar e excluir os rótulos e as regras do assistente de reuniões. Já a figura 4 exibe as telas de inclusão e alteração de rótulos e regras no sistema. A figura 5 mostra a tela de inclusão de unidades de informação de um rótulo.

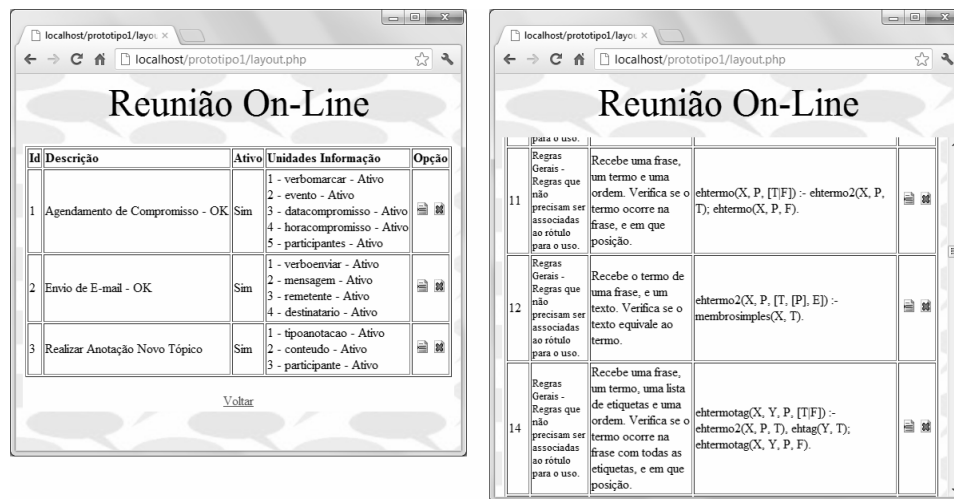


Figura 3 - Tela de consulta de Rótulos e Tela de consulta de Regras.



Figura 4 - Tela de Cadastro de Rótulos e Tela de cadastro de Regras.

Figura 5 - Tela de cadastro de Unidades de Informação.

O agente Rotulador possui internamente uma máquina Prolog que carrega e avalia todas as regras associadas a um rótulo, para verificar o seu casamento com a mensagem.

Para efetuar a avaliação das regras, a frase que está sendo analisada é transformada em uma série de fatos Prolog, e esses fatos são carregados nessa máquina interna. Em seguida, as regras do rótulo que será verificado são também carregadas nessa máquina Prolog, e finalmente são realizadas consultas à base de conhecimento. A figura 4 mostra um exemplo de avaliação de mensagens feita pelo agente Rotulador, onde foi carregado o rótulo “Agendar Compromisso”, com suas UIs e regras. As regras são avaliadas uma a uma, na ordem em que foram carregadas. Todo rótulo possui uma regra no formato **rotulo(UI)**, que é a consulta Prolog que traz os valores das UIs. E todo rótulo possui pelo menos uma regra que gera os valores das UIs. Na figura 5, esta regra está no formato da sentença **rotulo([UI1, UI2, UI3, UI4])**, simbolizando as 4 unidades de informação presentes na figura.

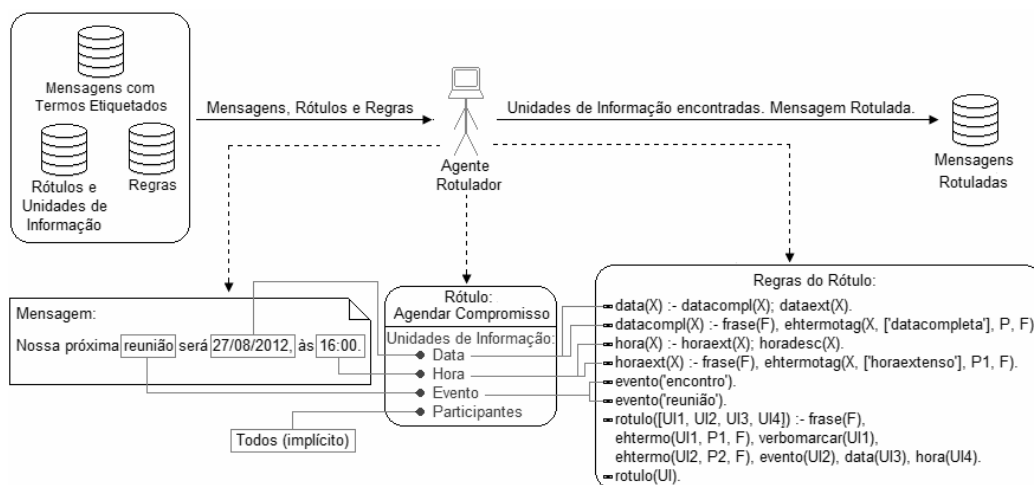


Figura 6 - Fluxo de informações do agente Rotulador - Agendar Compromisso

É possível observar que existem regras que buscam palavras específicas (“encontro” ou “reunião”, por exemplo); regras que buscam por etiquetas de palavras; e regras que buscam expressões compostas. Essa flexibilidade permite que os usuários do sistema configurem o conjunto de regras da forma como desejarem, incluindo, alterando ou excluindo as regras existentes.

A figura 6 mostra outro exemplo de avaliação de regras pelo agente Rotulador, dessa vez com o rótulo “Lembrar Anotações”. Nessa figura é possível observar que duas mensagens em formatos diferentes são identificadas pelo mesmo rótulo, devido ao conjunto de regras existente e às diversas formas que a linguagem Prolog tem de avaliar uma consulta. Nesse exemplo, está presente a consulta **rotulo(UI)**, e também a sentença **rotulo([UI1, UI2, UI3])**, pois esse rótulo só possui 3 unidades de informação.

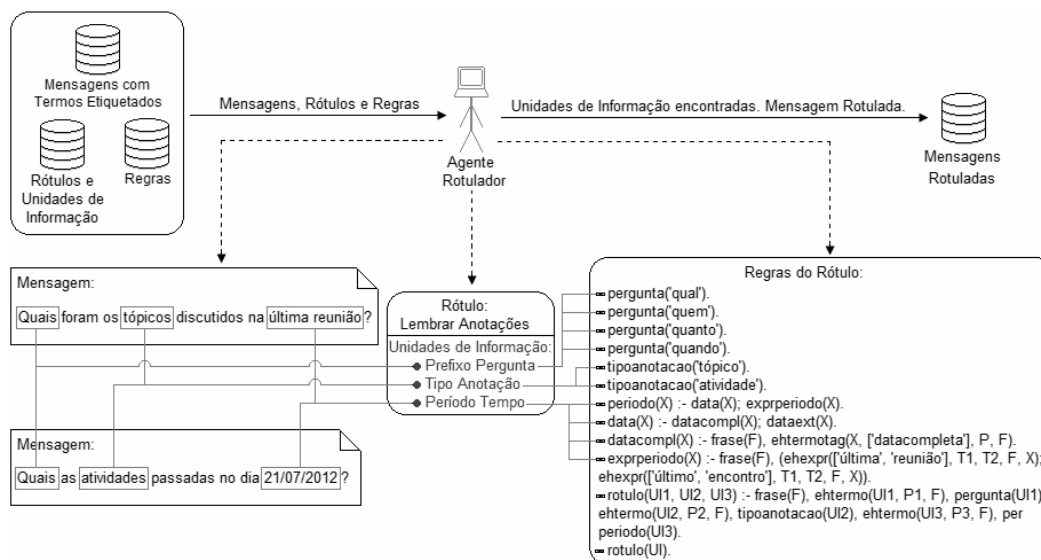


Figura 7 - Fluxo de informações do agente Rotulador - Lembrar Anotação

6. Análise dos Resultados

Nos exemplos apresentados, é possível observar como se dá o mecanismo de rotulagem do agente Rotulador, e como é efetuada a etiquetagem dos termos da mensagem através do agente Etiquetador.

Nos testes realizados com o protótipo desenvolvido, foram criadas regras, rótulos e unidades de informação específicas para a utilização do Assistente de Reuniões em uma disciplina de estudo dirigido on-line sobre o tema “Introdução a Sistemas Multiagente”. Alguns trechos de conversas reais foram inseridos no ambiente, verificando o comportamento dos agentes. Para as tarefas mais simples, o desempenho foi satisfatório: tarefas como o agendamento da próxima reunião ou a busca de um tópico anotado anteriormente passaram a ser realizadas automaticamente, poupando tempo e esforço dos participantes.

Portanto, a configuração detalhada desses dois agentes – através da criação de rótulos e UIs voltados a um domínio de conhecimento específico, e de um conjunto de regras que atenda as especificidades de uma dada reunião on-line – permite ao assistente

de reuniões atingir razoáveis taxas de precisão na obtenção da real intenção do usuário. A identificação dos rótulos, quando o ambiente está adequadamente configurado, consegue elevar o nível de automatização das tarefas mais simples, como mandar e-mails de convite, anotar tópicos discutidos e agendar as próximas reuniões. Isso tem um impacto direto nos participantes da reunião, pois com menos tarefas para se preocupar, eles podem se concentrar na produção intelectual, na reflexão e na disseminação do conhecimento.

7. Conclusões e Trabalhos Futuros

Neste trabalho foram descritos os comportamentos dos agentes Etiquetador e Rotulador, presentes na camada inteligente do Assistente de Reuniões, bem como a sua configuração por parte do usuário através da Reunião On-Line.

Esses dois agentes são de suma importância para o funcionamento do ambiente. Criar uma base de regras ampla e ao mesmo tempo aderente às especificidades de uma determinada reunião on-line (uma disciplina ou um estudo dirigido on-line, por exemplo) torna os rótulos existentes muito mais descritivos e identificáveis, o que permite que as atuações do assistente de reuniões sejam de fato proveitosas para os participantes da reunião. Da mesma forma, preencher a base de conhecimento do sistema com rótulos e UIs adequados ao tipo de mensagem comum ao grupo de participantes (levando em consideração o jargão técnico adotado e o domínio de conhecimento a que se referem normalmente as reuniões) aumenta a eficácia do sistema em auxiliar os usuários em suas tarefas.

Como trabalhos futuros, existem alguns caminhos que podem ser seguidos, entre eles:

1) A expansão do agente Etiquetador, através de técnicas de PLN, de forma a cobrir melhor as mensagens postadas. O agente Etiquetador é capaz de ele mesmo criar algumas etiquetas padronizadas (para sinais de pontuação, por exemplo). Mas a sua compreensão das mensagens ainda é muito limitada às etiquetas fornecidas pelo VISL.

2) Realizar um estudo mais aprofundado sobre o agente Rotulador, no sentido de simplificar a manutenção das regras de validação. A linguagem Prolog possui uma sintaxe relativamente simples, mas a exigência de se conhecer essa sintaxe para que o usuário possa incluir ou alterar as regras do sistema talvez seja um fator de bloqueio do usuário quanto à configuração adequada do sistema. Talvez a criação de uma linguagem de mais alto nível, que possa ser traduzida automaticamente para o Prolog, seja uma solução para esse problema.

8. Referências

- APACHE (2012) Open Source HTTP Server. <<http://www.apache.org/>>. Agosto.
- Bick, E. (2000) "The Parsing System "Palavras": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework". Tese de Doutorado, Aarhus University. Aarhus: Aarhus University Press.
- Bick, E. (2001) "The VISL system: Research and applicative aspects of IT-based learning". In: Proceedings of NoDaLiDa, 13th Nordic Conference on Computational Linguistics. Uppsala, Sweden: Department of Linguistics, Uppsala University.

- FIPA (2012) The Foundation for Intelligent Physical Agents <<http://www.fipa.org/>>. Agosto.
- Franconi, E. (1994) “Description logics for natural language processing”. In: Working Notes of the 1994 AAAI Fall Symposium on Knowledge Representation for Natural Language Processing in Implemented Systems.
- Gonzalez, M., Lima, V. L. S. (2003) “Recuperação de informação e processamento da linguagem natural”. In: XXIII CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. Anais da III Jornada de Mini-Cursos de Inteligência Artificial, p.347-395. Campinas.
- JADE (2012) Open Source platform for peer-to-peer agent based applications. <<http://jade.tilab.com/>>. Agosto.
- MYSQL (2012) Open Source Database. <<http://www.mysql.com/>>. Agosto.
- Palazzo, L.A. (1997) “Introdução à Programação Prolog”. Editora EDUCAT. Pelotas.
- PHP (2012) Free Web-based Language. <<http://www.php.net/>>. Agosto.
- Silva, C. A. S., (2012) “Um assistente inteligente para apoiar reuniões on-line”. Dissertação de Mestrado, Universidade Federal do Espírito Santo -UFES, Vitória-ES.
- VISL1 (2012) “Visual Interactive Syntax Learning” <<http://beta.visl.sdu.dk/>>. Agosto.
- VISL2 (2012) “Complex Parsing Interface - Visual Interactive Syntax Learning”. <<http://beta.visl.sdu.dk/visl/pt/parsing/automatic/complex.php>>. Agosto.
- VISL3 (2012) “The Constraint Grammar category set of ‘Palavras’ - Visual Interactive Syntax Learning”. <<http://beta.visl.sdu.dk/visl/pt/info/portsymbol.html>>. Agosto.
- Wooldridge, M. (2002) “An Introduction to Multiagent Systems”. John Wiley & Sons Ltd, New York.