

# Um Agente de Comunicação em Linguagem Natural no Contexto de um Sistema Multiagentes Orientado à Tutoria Inteligente na WWW

Eduardo Heredia<sup>1</sup>, Dr. Nizam Omar<sup>1</sup>, Jadir Custódio Mendonça<sup>1</sup>

<sup>1</sup> Universidade Mackenzie - São Paulo - SP - Brasil

eheredia@unipar-sp.com.br

omar@comp.cta.ita.br

jadircmj@yahoo.com.br

**Resumo.** O presente trabalho tem como objetivo apresentar a implementação de um Agente de Comunicação em Linguagem Natural (ACLN) no contexto de um sistema multiagentes orientado à Tutoria Inteligente na WWW. Como base do projeto foi utilizada a arquitetura ITStrategic – um protótipo de Sistema de Tutoria Inteligente. O ACLN está inserido no escopo do Módulo de Percepção desta arquitetura, tendo como principal finalidade a estruturação de interfaces homem-máquina com o acréscimo de características de comunicação em linguagem natural que facilitem e motivem a aprendizagem.

**Palavras Chave:** Sistema de Tutoria Inteligente, ITStrategic, Agentes, Processamento de Linguagem Natural, ACLN.

## 1. Introdução

Um dos maiores problemas na interação homem-máquina é a comunicação. Normalmente as pessoas não entendem a linguagem do computador, pois esta não corresponde às estruturas do pensamento humano.

O objetivo de interfaces em linguagem natural é proporcionar às pessoas uma forma de comunicação natural e simples com as máquinas. Interfaces em linguagem natural permitem aos usuários comunicarem-se com as máquinas em sua língua nativa (Português, Inglês, etc). Tais interfaces geram um impacto profundo no ambiente de trabalho, abrindo novas áreas de aplicação para a Tecnologia da Informação.

O futuro da lingüística computacional será determinado pelo crescimento da demanda por softwares com interfaces mais amigáveis e a criação de produtos de qualidade. A compreensão por parte de uma máquina da linguagem natural está muito longe da ideal, mas grandes progressos já foram feitos e bons resultados já podem ser obtidos.

Neste trabalho, resultado de uma dissertação de mestrado [HER 2001], uma arquitetura multiagentes para comunicação em linguagem natural foi desenvolvido para ser inserida na arquitetura ITStrategic [MAR 2000], a qual constitui um protótipo de Sistema de Tutoria Inteligente (STI) baseado em uma arquitetura de agente cognitivo, que define dinamicamente suas estratégias instrucionais, visando aumentar seu

grau de adaptabilidade ao estado mental do estudante.

O Agente de Comunicação em Linguagem Natural (ACLN) foi desenvolvido como uma sociedade composta por quatro agentes, quais sejam:

**Agente Interface:** responsável pela interação entre o ACLN e os agentes externos.

**Agente Lingüístico:** responsável pela análise sintática e semântica de uma “string” recebida do agente externo. Seu desenvolvimento foi baseado na teoria das redes de transição aumentadas.

**Agente Léxico:** responsável pela manipulação da base léxica (vocabulário) do ACLN. A ele é delegada a função de identificar e qualificar as palavras que constituem a “string” recebida do agente externo. Este agente também tem a função de inferir qual a entrada léxica correspondente, se é que existe, a uma palavra malformada.

**Agente Tradutor:** responsável por traduzir a análise efetuada pelo agente Lingüístico em uma forma adequado ao agente externo. Sua implementação define um método, chamado “Método Tradução”, que deverá ser personalizado para cada agente externo. Esta personalização é necessária uma vez que o agente Tradutor não necessariamente conhecerá a forma de representação da informação utilizada pelo agente externo.

## 2. Agente de Software e o ITStrategic

### 2.1 Definição de Agente

A literatura sobre agentes não apresenta concordância sobre sua definição. Segundo Marietto [MAR 2000], a falta de concordância entre as definições pode ser explicada pelas seguintes razões:

- A interdisciplinaridade e a abrangência do campo de pesquisa e do campo comercial nos quais a teoria de agente pode ser estudada e aplicada;
- Agentes estão presentes no mundo real e, em geral, os conceitos envolvidos com ambientes nos quais os agentes não estão inseridos são imprecisos e incompletos, dificultando a

categorização dos ambientes e, conseqüentemente, dos agentes que o representam.

Uma definição contextual que apresenta uma abordagem geral e didática pode ser encontrada no trabalho de Russel e Norvig [RUS 1995]. Russel e Norvig definem um agente como “... qualquer coisa que pode ser vista como percebendo seu ambiente através de sensores e atuando sobre o ambiente através de efetadores”. Outras definições de agente e uma comparação entre elas podem ser encontradas no trabalho de Franklin e Graesser [FRA 1996].

Wooldridg e Jennings [WOO 1995], em uma tentativa de padronização, apresentaram uma proposta de adoção das noções forte e fraca de agência. A noção fraca considera um conjunto de propriedades/atributos que um software ou hardware deve apresentar para ser considerado um agente. As propriedades consideradas são as seguintes:

**Autonomia:** agentes operam sem a intervenção direta de seres humanos ou outros sistemas, e têm algum tipo de controle sobre suas ações e estados internos;

**Habilidade Social:** agentes interagem com outros agentes (e possivelmente com outros seres humanos) através de algum tipo de linguagem de comunicação;

**Reatividade:** agentes percebem seu ambiente (que pode ser o mundo físico, um usuário através de uma GUI, outros agentes, a Internet, e talvez todos eles combinados), e respondem a mudanças que ocorrem neste ambiente;

**Iniciativa:** agentes não agem simplesmente em resposta a seus ambientes, mas tomam iniciativa e por isto são capazes de exibir comportamento orientado a objetivos.

A noção forte de agência considera que, além das propriedades apresentadas na definição de noção fraca, outras propriedades devem ser adicionadas ao software ou hardware. Estas propriedades estão relacionadas à características humanas, como por exemplo noções mentais (conhecimento, crença, intenção, obrigação) e emoções.

Neste trabalho, para a definição de agente, optou-se pela proposta de Durfee, Lesser e Corkill [DUR 1989], que considera o agente como:

“... resolvedores de problemas que podem trabalhar em conjunto para resolver problemas que estão além das suas capacidades individuais”.

### Agentes Reativos

Os agentes reativos não usam raciocínio simbólico complexo, estruturas de memória e uma representação interna explícita do conhecimento. Desta forma não possuem um histórico de suas ações passadas, nem podem fazer previsões de atos futuros. Um agente reativo apenas pode perceber o ambiente externo e, baseado nos estímulos recebidos, reagir de forma pré-determinada. Em função de sua arquitetura ser baseada em estímulo –resposta, agentes reativos não precisam de sofisticados mecanismos de coordenação e comunicação. O comportamento inteligente advém da interação dos comportamentos básicos de cada agente [MAR 2000].

### Agentes Cognitivos

Um agente possui cognição quando possui uma representação simbólica do mundo e toma suas decisões considerando esta representação. Desta forma, agentes cognitivos ou deliberativos são capazes de raciocinar a respeito de suas intenções e conhecimentos, criar planos de ação e executá-los. Possuem modelos explícitos do mundo externo, estruturas de memória que permitam manter um histórico de ações passadas e fazer previsões de ações futuras, e um sistema desenvolvido de cooperação e comunicação. Possuem controle deliberativo na medida em que conseguem deliberar qual ação será executada, e posteriormente executá-la [MAR 2000].

## 2.2. ITStrategic

O ITStrategic [MAR 2000] é um protótipo de STI baseado em uma arquitetura de agente cognitivo, cujo desenvolvimento embasou-se em pesquisas nas áreas de Inteligência Artificial Distribuída (IAD), Sistema de Tutoria Inteligente e Projeto Instrucional. Representa uma arquitetura de STI que define dinamicamente suas estratégias instrucionais, visando aumentar seu grau de adaptabilidade ao estado mental do estudante.

O ITStrategic é composto por cinco módulos e três interfaces, quais sejam:

- Módulo de Percepção;

- Módulo do Estudante;
- Módulo do Domínio;
- Módulo de Tutoria;
- Módulo de Entrega;
- GUI do Estudante;
- GUI do Educador;
- GUI do Administrador.

**Módulo de Percepção:** formado pelos agentes Receptores que personalizam o diálogo entre usuários e o sistema. O estudante interage com o sistema através do agente ReceptorEstudante, o qual faz parte dos Módulos de Percepção e Entrega.

**Módulo do Estudante:** formado pelo agente Guia, responsável pela ativação ou criação de um Modelo do Estudante personalizado. Este agente também atualiza o Modelo do Estudante, bem como disponibiliza informações sobre o mesmo à sociedade.

**Módulo de Domínio:** armazena a estrutura o conhecimento do domínio. O acesso a este banco de dados é realizado através do agente Informação.

**Módulo de Tutoria:** responsável pela escolha da arquitetura instrucional e posterior direcionamento das megaestratégias instrucionais. Seu funcionamento é regido pelo Agente Tutor. O comportamento próprio do Agente Tutor é direcionado por um Planejador e um Executador. O Planejador, tendo como base o atual estado mental do estudante elabora um plano instrucional linear. O Executador adequa uma metaestratégia instrucional para cada passo do plano gerado pelo Planejador. Cada passo de um plano é enviado ao Módulo de Entrega, para a posterior apresentação ao estudante do conceito correspondente.

**Módulo de Entrega:** aloca as macro e microestratégias instrucionais. Este módulo operacionaliza a tutoria de acordo com as linhas gerais definidas no Módulo de Tutoria através da Sociedade Didática dos Agentes de Arquitetura Instrucional (SDAAI) e Sociedade dos Agentes de Eventos Instrucionais (SAEI). Este é o módulo responsável pelo diálogo entre o sistema e o estudante.

### 3. Agente de Comunicação em Linguagem Natural

A Linguagem Natural, enquanto objeto de estudo, tem recebido inúmeros enfoques no decorrer do tempo. Não existem conclusões definitivas quanto as suas características, modelagem e funcionamento no âmbito computacional.

O problema da compreensão da linguagem humana não se reduz a meramente analisar uma sentença procurando o significado das palavras em um dicionário. Inferências são requeridas sobre as metas e perspectivas daquele que está falando, assim como o domínio no qual está inserido o discurso, o que agrega uma complexidade muito grande ao problema.

Na arquitetura ITStrategic, o Módulo de Percepção é responsável por intermediar o diálogo entre o sistema e o usuário. Na concepção original, o Módulo de Percepção é formado pelo agente ReceptorEstudante, abreviadamente chamado de agente Receptor. No trabalho de Mendonça [MEN 2001] desenvolveu-se o Agente Receptor, definindo uma interface gráfica com a qual o usuário pode interagir. Esta interface provê acesso ao conteúdo instrucional do ITStrategic através de uma linha de menus ou através da digitação de uma sentença em uma linha de comando. Tais sentenças são formuladas em um subconjunto restrito da linguagem natural, sendo sua interpretação realizada pelo agente de Comunicação em Linguagem Natural, ACLN. O ACLN complementa o módulo de Percepção agregando características de comunicação em linguagem próxima à natural ao Agente Receptor.

Baseado no conhecimento que o agente possui sobre “strings” em linguagem natural, o ACLN analisará a solicitação do usuário quanto a sua concordância com as regras gramaticais e vocabulário previamente definidos. Estando sintaticamente bem estruturada, a solicitação do usuário passará por uma avaliação semântica, através da qual o ACLN, à luz do conhecimento que tem sobre o Agente Receptor, tentará convertê-la em um comando válido suportado por este agente. Ao final de seu processamento, caso o ACLN não logre êxito em sua análise, enviará uma mensagem ao Agente Receptor, informando que não foi possível identificar as intenções do usuário. Caso obtenha êxito, uma mensagem será enviada ao Agente Receptor cujo

conteúdo contém um dos comandos suportados por este agente.

#### 3.1. Sociedade de Comunicação em Linguagem Natural

O ACLN é visto, a partir de sua interface, como sendo constituído por um único agente cujo objetivo é analisar sintática e semanticamente uma “string” digitada pelo usuário e, a partir do conhecimento que tem sobre o Agente Receptor, tentar inferir um comando disponibilizado por este agente e que corresponda de alguma forma àquilo que o usuário quis expressar quando fez a solicitação.

Em seu desenvolvimento, entretanto, o ACLN foi concebido para ser o “agente equivalente” ao resultado de uma sociedade de quatro agentes que se comunicam de forma a atingir os objetivos propostos para o ACLN. São eles:

- Agente Interface
- Agente Lingüístico
- Agente Léxico
- Agente Tradutor.

A esta sociedade foi atribuído o nome de Sociedade de Comunicação em Linguagem Natural.

#### 3.2. Os Agentes da Sociedade

Em conformidade com a arquitetura ITStrategic, os agentes da Sociedade de Comunicação em Linguagem Natural respeitam os seguintes princípios gerais de comportamento:

- Princípio da Benevolência
- Princípio da Sinceridade
- Princípio do Auto-Conhecimento
- Princípio da Comunicação-Assíncrona
- Princípio da Sociedade aberta

##### 3.2.1 Agente Interface

O Agente Interface é responsável por intermediar a comunicação entre o ACLN e o mundo exterior. Sua base de dados é composta por

quatro tabelas em uma base de dados relacional, quais sejam:

**Tabela BaseLingüística:** mantém uma relação das bases lingüística que o ACLN pode tratar. Por base lingüística entende-se uma linguagem natural sobre a qual os agentes possuem conhecimento. Para cada base lingüística relacionada nesta tabela deve haver pelo menos um agente lingüístico em correspondência.

**Tabela AgenteExterno:** mantém uma relação dos agentes externos conhecidos pelo ACLN e com os quais pode trocar mensagens. Adicionalmente, o registro de um agente externo nesta tabela associa a este agente uma base lingüística, subsidiando o Agente Interface com a informação necessária à seleção de um agente lingüístico apropriado.

**Tabela AgenteLingüístico:** relaciona os agentes Lingüísticos conhecidos pelo agente interface. Ao receber uma solicitação de um agente externo, o Agente Interface seleciona, através desta tabela, um dentre todos os agentes associados à base lingüística utilizada pelo agente externo.

**Tabela AgenteTradutor:** contém a relação dos agentes Tradutores conhecidos pelo Agente Interface. Para cada agente externo deve haver um agente Tradutor correspondente.

### 3.2.2. Agente Lingüístico

O Agente Lingüístico é responsável pela análise sintática e semântica de uma "string" através de técnicas advindas da teoria de rede de transição aumentada. A gramática reconhecida pelo agente é armazenada em tabelas de banco de dados relacional e representa o conhecimento que o agente possui da linguagem natural. As tabelas são:

**Tabela SímboloGramatical:** responsável por manter uma relação dos símbolos gramaticais. Por exemplo, *frase*, *SN*, *SV* e *Adjetivo* comporiam registros desta tabela.

**Tabela AtributoSímboloGramatical:** contém a lista de atributos relacionados a cada símbolo gramatical. Por exemplo, um *Verbo* pode possuir os atributos *pessoa*, *tempo* e *modo*.

**Tabela Gramática:** contém a gramática reconhecida pelo agente.

**Tabela Procedimento:** responsável por armazenar os procedimentos que devem ser executados para validar uma transição. Deve conter procedimentos escritos segundo a linguagem ACLN.

**Linguagem ACLN:** linguagem desenvolvida para a definição do código que deve ser executado para que uma transição entre estados ocorra em uma rede de transição. Compõe-se por um subconjunto da linguagem Pascal.

### 3.2.3. Agente Léxico

O Agente Léxico é responsável por identificar e qualificar as palavras que constituem o vocabulário da base lingüística em análise. Sua base de dados é construída a partir da gramática especificada no Agente Lingüístico com o acréscimo de tabelas para verificação de entradas malformadas.

### 3.2.4. Agente Tradutor

O Agente Tradutor é responsável por realizar o mapeamento entre uma árvore sintática/semântica gerada pelo Agente Lingüístico e transformá-la em uma "string" compreensível ao agente externo que solicitou o trabalho de tradução.

Para realizar seu trabalho, o agente Tradutor precisa conhecer a representação utilizada pelo agente externo. Realizará então um mapeamento muitos-para-um.

## 3.3. Protocolo de Comunicação entre Agentes

No protótipo desenvolvido a comunicação entre agentes ocorre através da passagem de mensagens KQML através do AMR JATLite. As *performatives* KQML utilizadas entre os agentes são:

**ask-one:** para um agente solicitar uma resposta de outro agente

**tell:** para um agente responder a outro agente

**sorry:** para um agente informar que não pode responder a uma questão prévia.

## 4. Resultados Obtidos

O ACLN foi totalmente codificado. Os quatro agentes que compõem sua arquitetura foram implementados. Um interpretador para a Linguagem ACLN foi desenvolvido e incorporado ao agente Lingüístico.

### 4.1. Ambiente de Desenvolvimento

#### Linguagem de Programação

A linguagem de programação escolhida para desenvolvimento do ACLN foi a linguagem Java. O fato de ser uma linguagem orientada a objetos, possuir inúmeras APIs em sincronia com as últimas tecnologias e ser portátil entre plataformas foram determinantes em sua escolha como ferramenta de desenvolvimento. Foi utilizada a distribuição Java 2 SDK versão 1.3 da Sun Microsystems.

A modelagem do sistema foi realizada em conformidade com a UML, sendo utilizada a ferramenta Together versão 4.2 como ambiente de modelagem e desenvolvimento.

#### Outros Softwares

O banco de dados escolhido para o desenvolvimento do sistema foi o InstantDB, um banco de dados desenvolvido totalmente em Java, em conformidade como SQL padrão e de livre distribuição. A versão utilizada pelo ACLN foi a versão 3.6. Para prover suporte à rede, o InstantDB necessita da utilização do driver RmiJdbc, que também proporciona o modelo de acesso em três camadas ao banco de dados.

O JATLite foi utilizado como “facilitador” do sistema federado. Seu Agent Message Router (AMR) foi utilizado especialmente para a troca de mensagens entre agentes.

#### Hardware

O protótipo foi desenvolvido em uma estação de trabalho Pentium III 455 MHz, com 256 Mb de memória RAM, 10 Gb de disco em plataforma Windows 98 e rede local através de protocolo TCP/IP. O roteador JATLite foi instalado em um servidor SCO Unix, plataforma Intel. O InstantDB foi instalado em um servidor Window NT 4.0, juntamente com o RmiJdbc.

### 4.2. Agente Receptor

O Agente Receptor [MEN 2001] integra o Módulo de Percepção do ITStrategic apresentando-se como uma interface gráfica com a qual o usuário pode interagir. Esta interface suporta interação com o usuário através de uma barra de menus ou de uma linha de texto na qual o usuário pode digitar um comando em linguagem próxima à linguagem natural.

No trabalho de Mendonça [MEN 2001] define-se um protocolo para troca de mensagens entre o ACLN e o Agente Receptor. Este protocolo limita as interpretações que o ACLN pode inferir sobre uma “string” digitada pelo usuário. Toda solicitação do usuário deve ser, em conformidade com o protocolo definido, mapeada pelo ACLN em um comando aceito pelo Agente Receptor, por exemplo:

UL: última lição; ML: mostrar lição; ME: mostrar exemplo, etc.

### 4.3. Desenvolvimento da Gramática

Visando abranger o maior número de frases que possam se traduzir em um comando válido para o Agente Receptor, a seguinte gramática foi implementada:

<Frases> -> <SN><SV>

<Frases>-> <SN>

<Frases>-><SV>

<SN>-><Pronome>

<SN>-><Artigo><Nome>

<SN>-><Nome>

<Nome>-><Substantivo>

<Nome>-><Adjetivo>

<Nome>-><Substantivo><Adjetivo>

<Nome>-><Adjetivo><Substantivo>

<SV>-><Verbo>

<SV>-><Verbo><SN>

<SV>-><Verbo><Preposição><SN>

A gramática desenvolvida foi inserida na base de dados do Agente Lingüístico.

#### 4.4. Agente Tradutor

O Agente Tradutor teve seu método MétodoTradução personalizado para utilização com o Agente Receptor. O algoritmo utilizado para mapear uma árvore sintática/semântica baseou-se em uma busca de padrões. A cada vocábulo do dicionário do Agente Léxico, criou-se um campo chamado Semântica. Uma vez que o Agente Receptor só utiliza o ACLN para mapeamento de ações da interface, cada sentença em linguagem natural deve ser expressa como uma ação para a interface. Desta forma o conteúdo semântico de todo verbo utilizado é uma ação.

#### 4.5. Exemplos de Frases Reconhecidas

- Eu quero ver a próxima lição
- Rever exemplo
- Ajuda
- Ver dica
- Mostrar próxima lição
- Rever dica
- Mostrar último exemplo
- Mostrar lição anterior
- Rever lição
- 

#### 4.6. Conclusões

Uma implementação de gramática utilizando a estrutura ACLN foi criada para validação da arquitetura. Esta gramática foi desenvolvida para ser utilizada na interação como Agente Receptor. Em sua concepção, o Agente Receptor, por ser um protótipo, apresenta um número pequeno de comandos que aceita via linha de comando. A gramática implementada e a personalização do MétodoTradutor do Agente Tradutor, apesar de simples, procuraram abranger as requisições mais frequentes através da linha de comandos.

Como trabalho futuro, uma gramática mais robusta para interação com outro agente com uma definição maior do número de comandos oferecidos será implementada.

#### 5. Conclusões

Este Trabalho apresentou uma arquitetura para um agente de comunicação em linguagem natural a ser inserido no contexto de um sistema STI particular, o ITStrategic.

A arquitetura proposta baseia-se na interação entre quatro agentes, quais sejam:

- Agente Interface
- Agente Lingüístico
- Agente Léxico
- Agente Tradutor

A gramática reconhecida por essa arquitetura é personalizável. Através da base de conhecimento do agente Léxico pode-se estabelecer o vocabulário a ser utilizado. Através da base de conhecimento do agente Lingüístico se estabelece a gramática a ser reconhecida. O agente Interface atua apenas como um intermediador entre o ACLN e agentes externos. O agente Tradutor possui um método, “Método Tradução”, que deve ser personalizado para atender as necessidades de cada agente externo conectado ao sistema.

Para sua implementação, o ACLN baseou-se na teoria das redes de transição aumentadas. Uma linguagem, chamada Linguagem ACLN, foi criada para dar suporte à utilização de procedimentos que validam as transições através da rede. Para esta linguagem um interpretador foi desenvolvido.

Para sua validação, esta arquitetura foi totalmente implementada em código Java e interconectada com o agente Receptor desenvolvido por Mendonça [MEN 2001], o qual apresenta uma interface cuja funcionalidade pode ser acessada através de uma linha de comandos em linguagem natural. O ACLN interage com o agente Receptor para traduzir a linha de comando digitada pelo usuário em um comando da interface.

Por ser totalmente personalizável e, apesar das limitações inerentes à tecnologia adotada, o agente ACLN é um agente de comunicação cuja base lingüística pode ser personalizada tornando-o genérico em relação aos agentes externos com o quais pode interagir.

## 6. Referências

- [DUR 1989] DURFEE, E.H.; LESSER, V.R.; CORKILL, D.D. Trends in cooperative distributed problem solving. IEEE Transaction Knowledge Data Eng, v.11,n.1,p.63-83,1989.
- [FRA 1996] FRANKLIN, S.; GRAESSER, A. Is it an agent, or just a program? A taxonomy for autonomous agents. Springer-Verlag: Proceeding of the Third International Workshop on Agent Theories, Architectures and Languages, 1996. URL: <http://www.msci.mephis.edu/franklin/AgentProg.html>, 1996.
- [HER 2001] HEREDIA, E. Um agente de comunicação em linguagem natural no contexto de um sistema multiagentes orientado à tutoria inteligente na WWW. São Paulo: Universidade Presbiteriana Mackenzie, 2001. Dissertação de Mestrado.
- [MAR 2000] MARIETTO, M.d.G.B. Definição dinâmica de estratégias instrucionais em sistemas de tutoria inteligente – uma abordagem multiagentes na WWW. São José dos Campos: Instituto Tecnológico de Aeronáutica, 2000. Tese de doutorado.
- [MEN 2001] MENDONÇA, J.C. Definição de um sistema de tutoria inteligente baseado em agentes do ITStrategic. São Paulo: Universidade Presbiteriana Mackenzie, 2001. Dissertação de Mestrado.
- [RUS 1995] RUSSEL, S.J.; NORVIG, P. Artificial intelligence – A modern approach. USA: Prentice Hall, 1995.
- [WOO 1995] WOOLDRIDGE, M.; JENNINGS, N. Intelligent agents – Theory and practice. Knowledge Engineering Review, v.10, n.2, 1995.