

O uso de Linguagem Natural Controlada para Representação de Conhecimento por Usuários Finais

Sérgio Roberto P. da Silva, Josiane Mechiori Pinheiro

Departamento de Informática – Universidade Estadual de Maringá (UEM)
Av. Colombo, 5790, zona 07 – 87020-900 – Maringá – PR – Brasil

srsilva@din.uem.br, jmpinheiro@din.uem.br

Abstract: *This paper proposes the use of controlled natural language as a way to bring domain experts to the first place in the process of acquisition and refinement of knowledge. The proposed language gives value to communicative aspects of language, using linguistics mechanisms such as quantifiers, pronouns, anaphors and ellipses to strengthen the textual cohesion and, therefore, to make the knowledge expression by the domain experts easier.*

Resumo: Este artigo propõe a utilização de linguagem natural controlada como uma forma de trazer os *experts* em um dado domínio para o primeiro plano dos processos de aquisição e refinamento de conhecimento. A linguagem proposta valoriza aspectos comunicativos das linguagens, usando mecanismos lingüísticos como quantificadores, pronomes, anáforas e elipses para aumentar a coesão textual e, assim, facilitar a expressão do conhecimento pelos *experts* no domínio.

1. Motivação

Atualmente as fases de aquisição e refinamento de conhecimento que compõem o processo de construção de uma base de conhecimento ocorrem de forma indireta, sendo, geralmente, realizadas de duas formas. A primeira, e mais comum, envolve um grupo de *experts* no domínio e um grupo de engenheiros de conhecimento que conhecem a linguagem formal de representação de conhecimento (LRC) empregada no sistema que gerenciará a base [RUSSELL,2003]. A segunda, e mais recente, envolve a utilização de algoritmos de aprendizado de máquina que extraem o conhecimento de uma massa de dados provenientes do domínio. Estes processos, em geral, são muito demorados e custosos e, além disso, apresentam um grande risco de geração de inconsistências na base. Estas inconsistências são mais freqüentes na primeira forma citada e são inerentes ao fato de que em um processo de comunicação, como o que ocorre entre os *experts* no domínio e os engenheiros de conhecimento, a interpretação dada para um mesmo elemento do domínio pelos diferentes integrantes do processo é sempre diferente [ECO,1990]. Na segunda forma, o maior problema está na escolha de um algoritmo adequado para o domínio em questão e na adequação do conjunto de dados às limitações impostas por este algoritmo.

Uma maneira de evitar o problema de interpretação é possibilitar aos *experts* no domínio fazerem a aquisição de conhecimento de forma direta. Para tal, é necessário que a LRC utilizada nesta tarefa seja de fácil compreensão e uso ser entendida e usada por estes usuários. No entanto, as LRC atuais mais comumente usadas, geralmente, são muito complexas em sua estrutura sintática e semântica para que estes *experts* a utilizem

de forma direta e natural. De fato, a maioria destas linguagens foi criada tendo basicamente eficiência computacional como prioridade e acabaram deixando de lado **aspectos comunicativos** que qualquer linguagem de representação deve apresentar.

A hipótese central deste artigo é que o uso de uma linguagem que valorize os aspectos comunicativos pode permitir trazer os *experts* no domínio para o primeiro plano nas fases de aquisição e refinamento do conhecimento, reduzindo, assim, seus custo e riscos. Mais especificamente, propomos o uso de uma linguagem natural controlada (LNC)—um subconjunto da linguagem natural cujo léxico, sintaxe e semântica são restringidos para um domínio específico [ALTWARG,2000]—, como um novo nível de interação com a base. Recentemente as LNCs tem sido propostas e usadas com relativo sucesso em várias tarefas [CLAW,2003; CLAW,2000], mostrando que, dentro de um domínio específico, esta pode ser uma boa alternativa para possibilitar que os *experts* no domínio se comuniquem diretamente com o sistema de gerenciamento da base de conhecimento. É importante salientar que não propomos o uso de uma LNC como a linguagem a ser empregada na representação interna do conhecimento. A linguagem aqui proposta deverá agregar mecanismos comunicativos ao sistema de gerenciamento da base, sendo necessária sua a tradução para a sua LRC interna.

Na seção 2, descrevemos como o uso de linguagem natural controlada pode ser interessante para a aquisição de conhecimento. Em seguida, na seção 3, apresentamos a estrutura da LNC proposta. Na seção 4, falamos sobre a implementação de um *parser* para esta linguagem. Depois, na seção 5, mostramos um exemplo do uso desta linguagem. Finalmente, na seção 6, trazemos nossas conclusões e sugerimos alguns trabalhos futuros.

2. Linguagem natural controlada e a aquisição de conhecimento

Uma LRC é uma linguagem formal empregada no processo de representação de conhecimento. Em geral, sua estrutura está intimamente ligada ao tipo de conhecimento a se representar e ao tipo de inferência que se deseja realizar sobre este conhecimento [DAVIS,1993]. Dentre os tipos mais comuns de LRCs encontram-se as lógicas, as regras de produção, as redes semântica, os sistemas de *frames*, os grafos conceituais e as redes causais [RICH,1991]. O maior problema com estas linguagens é que elas não fazem parte do dia-a-dia dos usuários finais—os *experts* no domínio—, sendo, portanto, necessário um árduo processo de aprendizado caso estes as queiram utilizar.

Para tornar o processo de aquisição de conhecimento e refinamento de uma base de conhecimento um processo acessível aos *experts* no domínio é necessário disponibilizar uma LRC que seja de fácil aprendizado e uso. Para isto, é necessário que ela contemple **mecanismos comunicativos** que sejam conhecidos destes usuários. Olhando por este aspecto, a candidata mais apropriada seria o uso da linguagem natural (LN), visto os usuários já a conhecerem. Entretanto, além da ambigüidade inerente a tal linguagem, o processamento de sua forma irrestrita ainda não é viável, apesar dos enormes avanços da área de processamento de linguagem natural [COLING,2002]. Contudo, o uso de uma linguagem natural controlada (um subconjunto da LN específico para comunicação dentro de um domínio) tem sido explorado em tarefas como a especificação de requisitos [FUCHS,1996] e a representação de conhecimento [PULMAN,1996] entre outras, com relativo sucesso. A grande vantagem de se usar uma LNC está no fato de que todas as sentenças usadas nesta linguagem estão corretas na LN da qual ela faz parte, mas nem todas as sentenças desta LN são permitidas na

linguagem controlada, o que possibilita a eliminação das sentenças ambíguas. Assim, um *expert* no domínio terá somente que aprender quais os tipos de sentenças que ele pode e quais ele não pode usar na representação do conhecimento, o que se torna uma tarefa bem mais simples do que aprender uma linguagem inteiramente nova.

No entanto, é importante salientar que na definição de uma LNC não basta introduzir ricos mecanismos de comunicação a uma LRC. É necessário que estes mecanismos sejam realmente empregados pelos *experts* no domínio na qual a LNC será usada. Além disso, é extremamente importante que o conjunto de mecanismos comunicativos introduzidos permita manter o processamento computacional da LNC dentro de uma faixa de eficiência razoável, de forma a não inviabilizar sua implementação.

3. O subconjunto da linguagem natural suportada pela LNC proposta

Em [DA SILVA,2001] apresentamos uma análise da linguagem utilizada por usuários finais para expressar planos no dia-a-dia como, por exemplo, receitas e manuais de faça-você-mesmo. Naquele trabalho visamos, dentre outras coisas, obter uma linguagem-tipo—a descrição de uma classe de linguagens—, na forma de uma LNC para programação por usuários finais. Contudo, a função final da linguagem resultante era a de atualizar uma base de conhecimento do design da aplicação. Este artigo procura ampliar o escopo de atuação da linguagem-tipo então definida para o campo da aquisição e refinamento de bases de conhecimento em geral. Deste modo, é interessante observar as características encontradas em tal linguagem em função deste novo objetivo.

Em nossa análise anterior, pudemos observar que o grande diferencial entre as linguagens de programação atuais e a linguagem comumente empregada pelos usuários finais para a especificação de processos está no mecanismo de referência a objetos empregado. A linguagem empregada por estes usuários apresenta uma forma muito natural de operar sobre objetos estruturados por meio do uso de conhecimento do senso comum, quantificadores, qualificadores e figuras de linguagem como, por exemplo, anáforas, elipses, metáforas e metonímias. Tais mecanismos ajudam as pessoas a fazer referência a características de objetos complexos uma vez que eles ocultam muitos detalhes sobre a estrutura usada em sua implementação. Esta observação também é válida se compararmos com as LRCs atuais, uma vez que ocorrem os mesmos fenômenos. Percebemos também que a estrutura destes planos permite expressar dois tipos de conhecimento: o declarativo, quando estão introduzindo os recursos a serem usados no processo—que é semelhante à definição das entidades da ontologia de trabalho—, e o procedimental, quando especificam o processo a ser executado sobre os recursos—que é semelhante à definição dos processos de inferências recomendados [DAVIS,1993] para a representação em questão. Além disso, pudemos identificar que de fato esta linguagem de planos pode ser dividida em três, sendo composta por uma sublinguagem para os mecanismos de **referenciação de objetos**, outra para os mecanismos de **metalinguagem**, e uma terceira para os mecanismos de **controle**. Este particionamento é bastante interessante do ponto de vista de representação de conhecimento, demonstrando uma forte relação entre os tipos de conhecimento a serem representados e a estrutura da linguagem usada para a sua especificação.

Em relação à estrutura do texto desta linguagem de planos, é interessante ressaltar que um parágrafo sempre realiza um passo de um plano, seja ele de declaração ou de transformação (controle de inferência no nosso caso), e este passo sempre declara

ou modifica um único objeto. Este objeto é o **foco do discurso** que, em geral, é bem marcado no texto e por isto muitas vezes é usado de forma implícita, como parâmetro no processo de inferência, por exemplo. Além disso, quando se tem que referenciar conjuntos de objetos a referência é feita por meio de quantificadores do tipo “*all*”, “*every*” e “*each*” e pelo uso do plural. Estes elementos referenciam o conjunto implicitamente e, normalmente, usam as relações de todo-parte existentes na ontologia do domínio para facilitar o caminhar sobre os elementos dos conjuntos. Os substantivos, que descrevem objetos no domínio, geralmente são substituídos por **pronomes** nas referências subsequentes a sua introdução no texto, um caso de **anáfora**. Além disso, é comum o uso de **elipses**, confinado às formas básicas, para ocultar objetos das sentenças. Neste caso, os pronomes ocultos sempre se referem a objetos já mencionados no contexto formado pelas sentenças anteriores, podendo ser omitidos sem prejuízo à sua interpretação. Estes tipos de mecanismos comunicativos previnem a necessidade de se escrever e, portanto, interpretar sentenças muito longas, o que é particularmente difícil para as pessoas. Ademais, devido às suas características estas referências são de simples resolução, facilitando em muito sua implementação. Esses são os tipos de mecanismos comunicativos que se espera encontrar em uma LRC para que os *experts* no domínio não tenham dificuldade em usá-la.

É interessante observar que em relação ao conhecimento procedimental, a primeira vista não há a noção de variáveis nestes planos. No entanto, em uma análise mais cuidadosa podemos notar que algumas palavras referem-se a *types*—uma classe inteira de objetos—e, portanto, simbolizam variáveis. Isto é possível devido ao uso de mecanismos anafóricos específicos. Deste modo, os nomes que aparecem na parte de declaração dos recursos—na definição da ontologia no nosso caso—, denotam *types* e são precedidos por quantificadores representados por artigos indefinidos (e.g. “*an e-mail*”). Quando estes mesmo nomes aparecem no corpo do plano—na definição do conhecimento procedimental—, eles geralmente denotarão um objeto único específico—um *token*—, e serão precedidos por um artigo definido (e.g. “*the e-mail*”). Esta é uma diferença crítica do ponto de vista comunicativo, por ocultar a necessidade da introdução de um novo tipo de objeto—as variáveis—, e mostra como a diferenciação *type/token* é sutil e sensível ao contexto neste tipo de linguagem, características estas que seriam interessantes às LNCs, mas que nem sempre são tão simples de serem implementadas.

É importante ressaltar que, apesar do corpus empregado nesta análise referir-se à língua inglesa, os comentários que fizemos também são válidos para outras línguas (com os devidos ajustes), uma vez que analisamos elementos que pertencem aos “mecanismos lingüísticos universais” [LIONS,1981] como, por exemplo, a estrutura frasal das diferentes formas de se referenciar um objeto nas linguagens.

3.1. As linhas gerais de uma LNC para representação de conhecimento

Nosso objetivo é definir uma LNC que possa funcionar como um nível superior de interação para as tarefas de aquisição e refinamento de conhecimento, atuando como uma camada acima da LRC empregada no gerenciamento da base. O intuito final desta camada é prover mecanismos comunicativos que habilitem a LRC a ser usada por usuários finais. A análise discutida na seção anterior nos mostra que, mantendo um domínio específico, é possível definir-se um conjunto de três linguagens

complementares que permitam a criação de uma LNC que pode expressar tanto o conhecimento declarativo quanto o procedimental.

Nossa análise mostra também que, em geral, o elemento crítico das linguagens de programação e das LRCs atuais é o fraco mecanismo de referência a objetos por elas disponibilizado. Isto ocorre devido à falta de mecanismos lingüísticos que facilitem seu uso como meio de comunicação com os seres humanos, uma vez que estas linguagens foram feitas para “conversar” bem com máquinas. Deste modo, qualquer LNC que venha a ser definida para a aquisição de conhecimento deve permitir tanto o uso de quantificadores, como “*all*”, “*every*” e “*each*”—que facilitam a expressão de *loopings* na definição de conhecimento procedimental—, quanto, principalmente, o de figuras de linguagem como anáforas e elipses, que operam no texto ampliando sua coesão textual e reduzindo o número de elementos a ser escrito e interpretado pelos usuários finais. Assim, nossa proposta divide a LNC a ser usada para representação de conhecimento em três sublinguagens, responsáveis pelos mecanismos de:

- **Metalinguagem**, que permitem a criação da ontologia domínio por meio da definição do conhecimento declarativo;
- **Controle**, que permite a criação a definição do conhecimento procedimental e do controle inferencial em alguns casos; e
- **Referenciação**, que incorpora os mecanismo comunicativos anteriormente citados, atuando de forma ortogonal a outras sublinguagens, para aumentar a coesão textual.

Devido à limitação de espaço, a seguir vamos esboçar somente as sublinguagens para os mecanismos de metalinguagem e referenciação. Uma descrição da sublinguagem para os mecanismos de controle pode ser encontrada em [DA SILVA,2001].

3.1.1. A Sublinguagem de Referência de Objetos

Esta sublinguagem caracteriza o diferencial da LNC proposta em relação a outras LRCs. Ela permite a referenciação a objetos simples e compostos empregando uma gramática restrita para sentenças nominais, semelhante à encontrada nas linguagens naturais. Entre outras coisas, ela admite o uso de determinantes para os objetos e um conjunto restrito de anáforas e elipses. Dentre os determinantes é possível especificar quantificadores (como “*all*”, “*every*”, “*each*” e “*an*”), qualificadores (ADJETIVOS, PRÉ-MODIFICADORES e FRASES PREPOSICIONAIS como, por exemplo, “*red*” *car*) e seletores (ORDINAIS que atuam na escolha de um elemento de um conjunto, tais como “*first*”, “*next*”, etc.). A Tabela 1 apresenta um conjunto de exemplos da classe de referências possíveis.

3.1.2. A Sublinguagem para os Mecanismos de Metalinguagem

Esta sublinguagem é responsável pela descrição do conhecimento declarativo da ontologia domínio. Para tal ela deve dispor de estruturas frasais que permitam a declaração de novas entidades ou o refinamento de entidades já existentes. Um tema relevante a ser considerado, no caso da aquisição de conhecimento, é a possibilidade ou não da criação de entidades sem ligação direta com a ontologia existente. Se isto não for possível, é necessária a definição de um conjunto de entidades básicas que deverão compor a base e que deverão fazer parte da linguagem, como acontece com os tipos primitivos nas linguagens de programação. A hipótese contrária permite ao usuário criar

Tabela 1: Exemplos de referências válidas na LNC proposta. Os elementos marcados com ** são os elementos usados na sublinguagem para os mecanismos de metalinguagem.

Tipo da referência	Exemplo
“noun” (um nome próprio) **	“personal”
[a/an] noun **	[a] message
noun(s) **	Messages
cardinal noun(s) **	7 messages
the/this noun **	the message
a [set list sequence] of noun(s) **	a set of messages
the noun _{ip} of [the] noun	the sender of the message
pronouns (restrito a it e them) **	It belongs to ... / ... and copy them.
the ordinal noun (onde ordinal = next/last/first/previous)	the last message
all [the] noun(s)	all the messages
each/every noun	each message

uma floresta de hierarquias dentro da ontologia, dificultado o processo de inferência. Nosso trabalho adota o princípio do que existirá um conjunto básico de entidades e relações das quais toda a ontologia deriva.

Um ponto importante a considerar em relação ao refinamento de conhecimento é a possibilidade ou não de se revogar conhecimento já existente na base, ou seja, a base terá crescimento monotônico ou não. Em geral, as bases de conhecimento não são monotônicas, possibilitando o apagamento de conhecimento caso este não seja mais válido. Apesar de importante, esta questão está mais relacionada ao processo de criação e manutenção da base do que com a LRC empregada no seu gerenciamento. Deste modo, a princípio vamos adotar a política de crescimento monotônico da base. Assim, o refinamento da base estará restrito à adição de atributos e relações às entidades. Esta restrição deverá ser retirada em trabalhos futuros de nosso grupo de pesquisa.

Do ponto de vista ontológico, nossa LRC aceitará que os usuários definam e manipulem os seguintes elementos: **entidades**—que representam as categorias existentes no domínio; **atributos**—que representam características destas entidades; **ações**—que representam as operações que uma entidade pode realizar no domínio; **partes**—que representam entidades que compõem (fazem parte ou estão agregadas a) uma outra entidade do domínio; e **relações**—que representam as inter-relações entre as entidades do domínio. Apesar de a LNC proposta aceitar qualquer tipo de relação, as relações de classificação (“*is a*”) e partonomia (“*part of / has*”) são predefinidas, por serem úteis na resolução de anáforas e elipses. Uma parte da gramática da classe de sentenças aceita pela sublinguagem para os mecanismos de metalinguagem é apresentada na Tabela 2.

4. Uma implementação para LNC proposta

Um *parser* para a LNC proposta foi implementado usando a técnica de *top-down* empregando o mecanismo de *DCG* presente na linguagem *Prolog*. Ele está dividido em

níveis de modo a facilitar a sua manutenção e o seu entendimento, conforme mostra a Figura 1. Sua entrada é composta de um arquivo texto, contendo as atualizações para a

Tabela 2: Gramática para a classe de sentenças aceitas pela sublinguagem para os mecanismo de metalinguagem.

Object_Reference_1 'is' Object_Reference_2 '.'
Object_Reference_1 'is' Object_Reference_2 'that has' Object_Reference_3 '.'
Object_Reference_1 ('is part of' 'are parts of') Object_Reference_2 '.'
Object_Reference_1 'has' Object_Reference_2 '.'
Object_Reference_1 'has' Object_Reference_2 ',' 'with possible' ('value' 'values') Object_Reference_3 '.'
Object_Reference_1 'has' Object_Reference_2 ',' 'which is' Object_Reference_3 '.'
Object_Reference_1 'has default value' Object_Reference_2 '.'

base descritas na LNC proposta. Este arquivo passa pelos analisadores léxico e sintático, gerando uma árvore sintática decorada com informações relevante à resolução de referências. Esta árvore é entregue ao analisador semântico que trabalhará ligado diretamente à base de conhecimento, verificando o correto uso de referências à objetos da base e validando as definições do texto. Para realizar estas validações este analisador trabalhará em conjunto com um algoritmo de resolução de anáforas. Termina esta fase a base será atualizada.

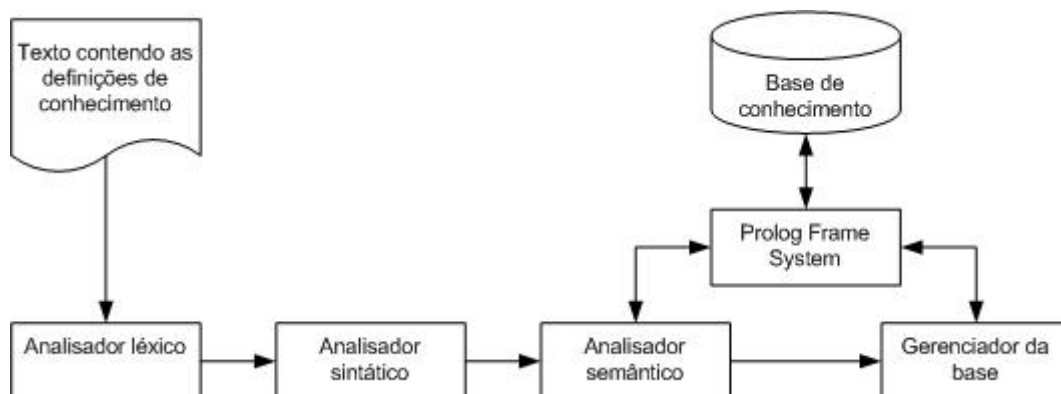


Figura 1: A estrutura do parser para a LNC proposta.

A resolução de anáforas é um fator crítico em uma LNC, uma vez que um dos seus principais objetivos é reduzir a quantidade de texto a ser escrita e interpretada pelos usuários. A LNC proposta aceita uma classe restrita de anáforas intra e inter-sentenciais. A principal restrição imposta está relacionada à classe de pronomes que podem ser empregados na criação destas anáforas. Em particular, como os domínios de interesse não denotam gênero, restringimos o uso de pronomes ao pronome neutro (“it”) e a terceira pessoa do plural (“them”), reduzindo, assim, a complexidade do algoritmo a ser utilizado. Inicialmente empregamos o algoritmo de *History List*. No entanto, um estudo mais cuidadoso nos mostrou que no caso de sentenças com cópula (“is a”)—por exemplo, “A car is a vehicle”—este tipo de abordagem não funcionaria. Isto acontece porque, ao aparecer no discurso, este tipo de sentença traz o foco do discurso para a entidade que está sendo definida (“car”). Como solução, incrementamos a abordagem do *history list* com características do algoritmo de *Centering* [ALLEN,1995]. Deste modo, ao terminar de analisar uma sentença (i.e., ao encontrar um ponto final) as

entidades do discurso serão inseridas no *history list* respeitando uma ordem que reflete as preferências estruturais lingüísticas, ou seja, primeiro o sujeito, em seguida o objeto direto, depois o objeto indireto, e então as outras entidades do discurso presentes na sentença. Esta modificação é suficiente para tratar as anáforas presentes no conjunto de sentenças aceitas na LRC proposta e também tem uma implementação eficiente.

Até o presente momento, foi realizada a implementação somente para as sublinguagens de referência de objetos e para os mecanismos de metalinguagem [PINHEIRO,2003]. A base de conhecimento utilizada está baseada em um sistema de *frames* e também foi implementada na linguagem *Prolog*. A escolha por este tipo de LRC interna se deve ao fato de os sistemas de *frames* permitirem a representação tanto do conhecimento declarativo quanto do conhecimento procedimental. Na base de conhecimento empregada em nossa implementação, uma entidade é representada como um *frame*, tendo pelo menos duas relações que determinam as hierarquias de classificação e composição. Uma relação também é representada por um *frame* individual, assim, como os atributos da entidade. Porém, neste último, o nome do *frame* é composto pelo nome da entidade a que o atributo pertence mais o nome do atributo. Maiores detalhes da LRC empregada podem ser encontrados em [DA SILVA,2001].

5. Um exemplo de uso da LNC proposta

Nesta seção apresentaremos um exemplo concreto do uso de LNC proposta. Neste caso estaremos tratando do problema de refinamento da ontologia de um sistema de *e-mail* apresentada na Figura 2 abaixo. A ontologia inicial conta com a entidade *e-mail* somente (não serão apresentadas aqui as entidades que compõem a base da ontologia) e será acrescida das entidades *personal-info*, *contact* e *address-book*. O texto que será usado pelo *expert* no domínio para atualizar a base de conhecimento é apresentado na Figura 3.

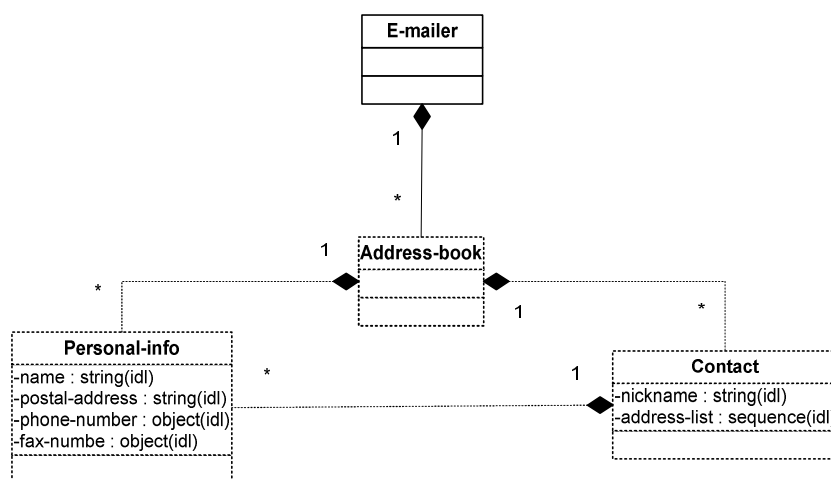


Figura 2: Um descrição do refinamento de uma ontologia para um sistema de e-mail. As entidades tracejadas foram adicionadas pelo refinamento.

Como foi dito na seção 4, o *parser* desenvolvido para o processamento da LNC proposta está apto a processar este diálogo e resolver as referências anafóricas interfrasais nele contidas (como o uso do pronome *it*) e também o uso restrito do determinante *the*. Deste processo resulta uma árvore intermediária que será processada para realizar as alterações na base de conhecimento.


```

%%% code for the definition of a PERSONAL INFO
A personal-info is a concept. It has a name, which is a
string. It has a postal-address, with possible values "work"
and "home". It has a phone-number and a fax-number, which are
integers. The phone-number has default value 0.

%%% code for the definition of a CONTACT
A contact is a concept that has a personal-info. It has
nickname and an address-list, which are strings.

%%% code for the definition of an ADDRESS BOOK
An address-book is a concept. It has owner-info, which is
personal-info. It has a set of contacts.

```

Figura 3: Texto contendo o refinamento de uma ontologia para um sistema de e-mail.

É relevante observar que neste exemplo limitamo-nos a representar o uso da LNC proposta para representação do conhecimento declarativo, por ser esta a parte que se encontra implementada atualmente. A estrutura resultante a ser inserida na base de conhecimento irá variar com o tipo de LRC empregada. Em nosso caso, a estrutura de *frames* resultante do processamento das duas primeiras sentenças será representada conforme mostra a Figura 4.

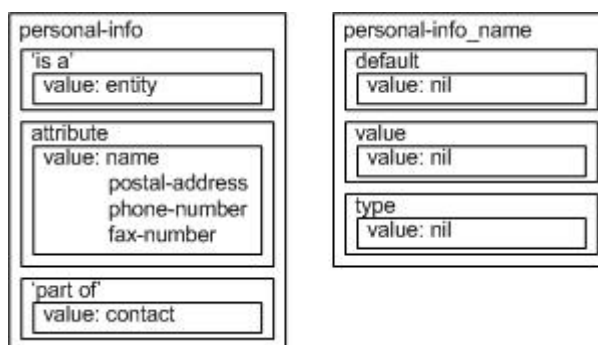


Figura 4: Representação das estruturas criadas na base de conhecimento pelo processamento das duas primeiras sentenças do exemplo da Figura 3.

6. Discussão final

A proposta deste trabalho é reduzir os custos e riscos dos processos de aquisição de conhecimento e refinamento de bases de conhecimento trazendo os *experts* no domínio para o primeiro plano destas tarefas. Para tal, propomos o uso de uma linguagem natural controlada como uma nova camada de interação com a linguagem de representação de conhecimento da base. A linguagem proposta é uma instância da linguagem-tipo proposta por Da Silva [DA SILVA,2001], composta por um subconjunto da linguagem usada pelas pessoas leigas para expressar planos no seu dia-a-dia, como a linguagem usada nas receitas de bolo e nos manuais de faça-você-mesmo.

A vantagem direta de se usar uma LNC na aquisição de conhecimento advém do fato do usuário não ter que aprender uma nova linguagem. Em vez disso, ele terá apenas que se acostumar com os tipos de sentenças que são permitidas na LNC proposta. Uma outra vantagem relevante é o fato de se permitir que o usuário use recursos naturais de comunicação da LN, como as figuras de linguagem. Estes mecanismos aumentam a coesão textual reduzindo a quantidade de texto a escrever e interpretar, tornando a tarefa

de aquisição e refinamento de conhecimento um processo muito mais natural para os *experts* no domínio.

Até o presente momento, foi implementado um *parser* para uma parte da LNC proposta. Este *parser* cobre as classes de sentenças descritas na seção 3. Tais sentenças permitem o uso de anáforas simples, que são resolvidas por um algoritmo de resolução de anáforas que contém características tanto do algoritmo de *History List* quanto do algoritmo de *Centering*. Além disso, o *parser* também suporta um conjunto restrito de elipses, como a omissão de verbos e sujeitos dentro do conjunto de sentenças válidas.

Por ter sido definida com o propósito de aquisição de conhecimento, a LNC aqui posposta também pode ser utilizada na etapa de elicitação de requisitos do processo de desenvolvimento de software, facilitando sua especificação diretamente pelos usuários e evitando, assim, problemas de interpretação. Outra aplicação já estudada é seu uso como linguagem de programação por usuários finais em aplicações extensíveis [DA SILVA,2001].

É importante salientar que ainda é necessária a realização de testes de usabilidade com a LNC proposta para que possamos ter certeza de que ela é realmente de fácil aprendizagem para os *experts* no domínio. Além disso, é necessário dizer que a simples definição de uma LNC não resolve o problema de aquisição de conhecimento. É preciso também que a interface do ambiente de atualização auxilie o usuário nesta tarefa, orientando seus passos durante o processo, como se o usuário estivesse seguindo um roteiro. Um ambiente com esta estrutura está sendo desenvolvido no âmbito do projeto APEX—Aplicações Extensíveis por Usuários Finais [DA SILVA,2003]—, dentro do qual este trabalho se enquadra.

Como trabalhos futuros pretendemos, refazer a implementação do *parser* da LNC proposta usando o mecanismo de *Discourse Representation Theory* [BLACKBURN,1999]. Esta mudança na linguagem intermediária do *parser* nos dará condições de traduzir o texto para a linguagem Prolog e também ampliar seu escopo de resolução de anáforas e elipses.

Referências Bibliográficas

- [ALLEN,1995] ALLEN,J., *Natural Language Understanding*, 2nd Edition. The Benjamim/Cummings Publish Company, Inc., 1995.
- [ALTWARG,2000] ALTWARG,R., *Controlled Languages: An Introduction*, <http://www.mri.mq.edu.au/ltg/slp803D/class/Altwarg/index.html>, último acesso 09/06/2003 às 13:55.
- [BLACKBURN,1999] BLACKBURN, P. AND BOS, J., *Representation and Inference for Natural Language. Vol. II – Working with Discourse Representation Structure*, Universität des Saarlandes, September 1999.
- [CLAW,2003] CLAW 2003, *Proceedings of the Fourth International Workshop on Controlled Language Applications*, 2003.
- [CLAW,2000] CLAW 2000, *Proceedings of the Third International Workshop on Controlled Language Applications*, 2000.
- [COLING,2002] COLING, *The 19th International Conference on Computational Linguistics*. Taipe, 2002.

- [DA SILVA,2001] DA SILVA, S.R.P., *Um Modelo Semiótico para Programação por Usuários Finais*, Tese de Doutorado, Departamento de Informática PUC-Rio, Rio de Janeiro, Maio 2001.
- [DA SILVA,2003] DA SILVA, S.R.P., APEX — *Aplicações Extensíveis por Usuários Finais*, Projeto aprovado pelo Fundação Araucária, 2003.
- [DAVIS,1993] RANDALL, D., What is knowledge Representation?, *AI Magazine*, 14(1):17-33, 1993
- [FUCHS,1999] FUCHS, N.E., SCHWITTER, R., AND SCHWERTEL, U., *Attempto Controlled English (ACE) – Language Manual*, Institut für Informatik der Universität of Zürich, August 1999.
- [LYONS,1981] LYONS, J., *Language and Linguistics*. Cambridge University Press. London, UK, 1981.
- [PINHEIRO,2003] PINHEIRO, J.M., *Atualização de Bases de Conhecimento pelo Usuário Final* Trabalho de Graduação, TG-18-02, UEM, Maringá, Paraná, 2003.
- [PULMAN,1996] PULMAN, S.G., Controlled Language and Knowledge Representation. In *Proceedings of the First International Workshop on Controlled Language Applications*, Katholieke Universiteit Leuven, Belgium, March 1996, pg. 233-242.
- [RICH,1991] RICH,E. AND KNIGHT, K., *Artificial Intelligence*, 3rd Edition. McGraw-Hill, New York, NY, 1991.
- [RUSSELL,2003] RUSSELL, S. AND NORVIG, P., *Artificial Intelligence – A Modern Approach*, 2nd Edition. Prentice -Hall, Inc., Upper Saddle River, NJ, 2003.