

Uma Metodologia para Avaliação Computacional do Nível de Compreensão de Textos

Jaime Wojciechowski¹, Bráulio Ávila¹, Fabrício Enembreck¹, Edson Scalabrin¹, João da S. Dias²

Pontifícia Universidade Católica do Paraná – PUCPR

¹Programa de Pós-Graduação em Informática Aplicada – PPGIA

²Programa de Pós-Graduação em Tecnologia em Saúde – PPGTS

Rua Imaculada Conceição, 1155, Prado Velho, 80215-901, Curitiba, PR, Brasil

jaimewo@terra.com.br, {avila, fabricio, scalabrin, jdias}@ppgia.pucpr.br, j.dias@pucpr.br

Resumo

Este artigo apresenta um sistema computacional capaz de avaliar, em um ambiente de aprendizagem colaborativa, o nível de compreensão que um indivíduo possui sobre um dado assunto. Uma questão difícil e essencial é o modo de avaliação, que em geral é realizada por meio de trabalhos coletivos ou em grupo ou ainda por meio de exames individuais. Entretanto, estas técnicas estão sujeitas a fatores subjetivos relacionados ao tutor (nível de experiência, humor, afinidade, etc.) e ao membro da equipe, além de requerer um considerável esforço de leitura por parte do tutor. A realização deste objetivo envolveu, essencialmente, técnicas de processamento de linguagem natural baseadas em conceitos tais como: memória dinâmica, raciocínio baseado em casos e *parser* semântico. Finalmente, uma discussão é mostrada sobre experimentos realizados em um ambiente de aprendizagem colaborativa. Os resultados obtidos foram coerentes de acordo com a opinião dos tutores envolvidos.

Palavras-chave: Memória Dinâmica, *Parser* Semântico, Processamento de Linguagem Natural.

Abstract

This article presents a computational system capable to evaluate, in an environment of collaborative learning, the understanding level that an individual possesses on a given subject. A difficult and essential question is the evaluation way that in general is carried through by means of collective works or in group or still by means of individual examinations. However, these techniques are citizens the related subjective factors to the tutor (experience level, mood, affinity, etc.) and to the member of the team, beyond require a considerable reading effort on the part of the tutor. The accomplishment of this objective involved, essentially, techniques of natural language processing based in concepts such as: dynamic memory, case-based reasoning and semantic parser. Finally, a discussion is shown on experiments carried through in an environment of collaborative learning. The gotten results had been coherent in accordance with the opinion of the tutors in involved.

Key-words: Dynamic memory, Semantic Parser, Natural Language Processing.

1 Introdução

A inteligência artificial tem proposto soluções interessantes para ambientes de trabalho colaborativos e ambientes colaborativos de formação continuada. Dentre elas, tem-se buscado também alternativas para se levar em conta à distribuição natural destes ambientes na tentativa de melhorar as interações neste meio, como por exemplo, a personalização da informação [19] [3] [7]. Técnicas de personalização da informação também são

empregadas para processamento automático de documentos (filtragem, análise e classificação) com o objetivo de reduzir a quantidade de informação manipulada pelos usuários, assim como automatizar determinadas tarefas [4] [5]. Por outro lado, formas de representação de conhecimento são ferramentas poderosas para o desenvolvimento de ontologias que visam normalizar o vocabulário de determinado domínio, reduzindo conflitos de interpretação e facilitando a navegação em hierarquias de conceitos [6] [7].

Menos frequentes são aquelas aplicações que utilizam recursos lingüísticos, de cognição humana e representações

computacionais complexas de eventos e ações para a interpretação de conteúdos em texto livre em ambientes colaborativos. Exemplos de aplicações de processamento de linguagem natural em ambientes colaborativos com essas características podem ser encontrados em [5] [2].

Apesar da complexidade das técnicas envolvidas no desenvolvimento de aplicações de processamento de linguagem natural, acredita-se que elas podem contribuir significativamente para a redução do impacto negativo relacionado à sobrecarga cognitiva provocada pelos sistemas e interfaces computacionais de hoje. Mais particularmente, a interpretação automática de um conteúdo, abre inúmeras oportunidades principalmente em um ambiente colaborativo de formação continuada [10] [20]. Enquanto tais ambientes facilitam a distribuição do conhecimento entre indivíduos e auxiliam na organização da informação, poucos deles oferecem ferramentas automatizadas e/ou semi-automatizadas de avaliação individual. A maior parte desses ambientes reproduz exames e/ou exercícios previamente elaborados por tutores e não asseguram realmente uma avaliação precisa do nível de entendimento que um indivíduo tem sobre determinado assunto.

Com o objetivo de reduzir a sobrecarga cognitiva do tutor e indiretamente eliminar fatores subjetivos que podem influenciar a avaliação de indivíduos, este trabalho baseia-se em técnicas automatizadas de compreensão textual que permitem avaliar o nível de conhecimento do autor de determinada dissertação. O sistema utiliza uma hierarquia de conceitos previamente modelada e técnicas de representação de memória dinâmica, raciocínio baseado em casos e processamento semântico de linguagem natural para identificar dentro de um texto o uso correto destes conceitos por meio de métricas relacionadas às definições semânticas, frequência e encadeamento de idéias. Essas métricas permitem que um texto seja avaliado qualitativamente, produzindo uma nota à interpretação do autor sobre o assunto. Nossa aplicação se mostrou bastante consistente em programas de ensino de diferentes domínios como medicina, enfermagem e educação.

Este artigo está organizado da seguinte forma: a seção seguinte introduz alguns conceitos elementares sobre compreensão da linguagem natural e raciocínio baseado em casos. A seção três discute a ferramenta DMAP, um modelo de reconhecimento e indexação de expressões de linguagem natural que foi utilizado neste trabalho. A quarta seção apresenta uma metodologia para avaliação computacional do nível de compreensão de textos usando o DMAP. A quinta seção mostra os experimentos realizados. As demais seções apresentam uma discussão sobre a metodologia e os resultados obtidos, bem como algumas conclusões.

2. Compreensão da Linguagem Natural usando Raciocínio Baseado em Casos

A compreensão da linguagem natural usando o raciocínio baseado em casos visa representar a maneira com que o ser humano compreende uma frase [12] [14] [15]. Nesta abordagem, supõe-se que o ouvinte possui um conhecimento prévio do assunto que está sendo enunciado e que este conhecimento já foi adquirido por meio de suas experiências pessoais. Desta forma, a compreensão de uma frase é um processo de recuperação de conceitos previamente indexados na memória do ouvinte, que na medida em que os termos são ouvidos, não ocorre apenas a identificação dos conceitos, mas também uma previsão do que pode vir na seqüência para que a frase seja compreendida. Todo este processo de recuperação de conceitos segue os princípios do raciocínio baseado em casos, passando pelas etapas elementares, que são: recuperação, adaptação, avaliação, uso e armazenamento dos casos [12]. A etapa de recuperação realiza a busca dos casos em uma memória. As etapas de adaptação e de avaliação garantem que o caso recuperado seja o mais relevante para a situação em questão. As etapas de uso e de armazenamento respondem respectivamente pelo reconhecimento de um conceito e criação de uma nova instância do conceito reconhecido. Tal instância será adicionada na memória.

Por exemplo, se alguém citar a frase:

“Fui ao...”

A pessoa que está ouvindo a frase começa a ter expectativas do que virá na seqüência. Estas expectativas podem passar a idéia de um lugar ou evento. Elas podem dar início a abstração e imaginação de possíveis lugares, eventos ou situações. Finalmente, pode-se ter a expectativa de que a próxima palavra refere-se mesmo a um lugar ou evento. Neste momento, diversos casos já foram recuperados para tentar encaixar com a frase (Fui ao dentista, restaurante, desespero, etc.). Mais algumas palavras são ditas, por exemplo, “fui ao restaurante e comi...”.

Neste ponto, todos os casos que não se relacionam à ida a um restaurante são descartados. Outras expectativas, então, surgem. Porém, o contexto está praticamente definido por se tratar de um evento do nosso cotidiano, onde é possível visualizar sem dificuldade um restaurante baseado-se nos casos recuperados a partir de nossa memória; casos estes que armazenamos por meio de nossas experiências pessoais e vivências.

Ao ser dita a última palavra da frase, por exemplo, “Fui ao restaurante e comi Sushi.”, a frase teria sido compreendida e somente um caso se enquadraria. Se a última palavra dita fosse, por exemplo, “Governo Federal”, isto causaria um desconforto ao processo de entendimento. Não teria sentido ir a um restaurante e comer um órgão institucional do país. A primeira reação seria indagar se Governo Federal é o nome de um prato servido em um

determinado restaurante. Em caso afirmativo, mesmo achando estranho um prato com este nome, seria incluído este nome na relação de pratos passíveis de serem servidos em um restaurante (ou naquele restaurante específico). O conflito se daria se o interlocutor afirmasse de forma categórica que não se trata de um prato e sim do órgão que dirige o país. Neste caso, a frase simplesmente não seria compreendida.

Note que, para um índio que vive isolado no interior de uma selva, sem nunca ter tido contato com a civilização, no início da frase (“Fui ao...”), as expectativas seriam outras, Fui ao rio, Fui ao campo, etc. No momento em que a palavra restaurante fosse apresentada, ele não “compreenderia”, pois não havia recuperado nenhum caso em sua memória com este desconhecido conceito.

Em [11] é apresentado um sistema de compreensão de linguagem natural baseado em raciocínio baseado em casos. Este sistema toma a forma de um *parser* baseado em casos e sua espinha dorsal, no processo de compreensão, estrutura-se sobre uma memória de conceitos previamente modelados e indexados. Nesta linha, será apresentada a seguir a teoria básica de um DMAP (**D**irect **M**emory **A**ccess **P**arser) e também como ele atua no tratamento de frases e textos vis-à-vis a compreensão da linguagem natural.

3. DMAP

O DMAP é um sistema voltado a compreensão de frases e textos em linguagem natural [7]. O processo de compreensão consiste em encontrar uma conexão entre um texto de entrada e uma estrutura de memória. A criação do DMAP partiu da tentativa de unir as idéias de compreensão definidas por M. Quilian [18] ao conceito de memória dinâmica de R. Schank [16]. Questões como organização, indexação e busca em memória são os pontos importantes deste sistema. Um problema constante na compreensão da linguagem natural é a ocorrência de ambigüidades, onde uma única palavra pode representar uma grande variedade de significados. Uma possível solução para este problema é considerar o texto (ou as sentenças de entrada) em um contexto bem definido, restringindo, assim os significados e relacionamentos que as palavras podem assumir. Desta forma, é fortalecido o princípio de que para se efetivar a compreensão é necessária à identificação do relacionamento envolvendo enunciados (textos, frases) e conhecimentos previamente já experimentados.

3.1. Organização da Memória

O processo de compreensão proposto no DMAP inclui uma memória de conceitos previamente modelados e indexados. Esta memória é baseada em MOPs (**M**emory **O**rganization **P**ackages). Cada MOP é uma estrutura do tipo *frame* [9] [1]. Ele é usado para representar eventos da comunicação e índices. Cada índice tem a função de representar um critério de ativação de um conceito. A identificação de um conceito é realizada a partir dos índices. Inicialmente, o índice é percorrido e as possíveis entradas são marcadas. À medida que a frase vai sendo apresentada, dois mecanismos básicos entram em funcionamento. O primeiro busca um índice (se existe um critério) para a entrada apresentada e o segundo verifica se alguma previsão foi identificada (ou marcada). Nos dois casos, a compreensão ocorre quando uma previsão de entrada satisfaz o critério que estava sendo esperado. Ao final do processamento da frase, apresentam-se todos os conceitos que foram ativados, mostrando assim que a frase foi compreendida para aquela memória de conceitos. A Figura 1 apresenta um exemplo de memória extraído de [7].

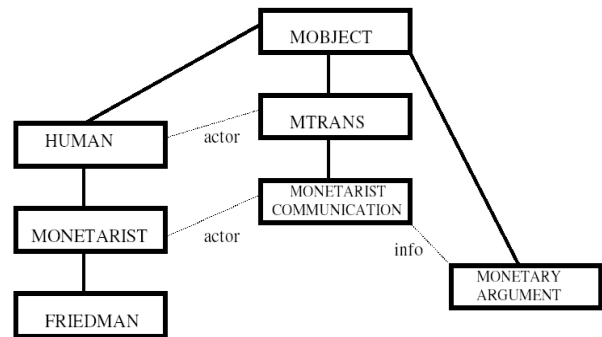


Figura 1 - Exemplo de memória de conceitos.

Nesta rede de conceitos, MTRANS significa “Mental Transfer” simbolizando uma transferência mental de uma idéia por parte de um *actor*. MOBEJCT é o conceito principal do qual derivam todos os outros. Para esta memória, teremos o seguinte índice:

Mtrans-1: {(actor) says (info)}

Friedman-1: {Milton Friedman}

Monetary-Argument-1: {increase tax}

O critério para que o conceito FRIEDMAN seja ativado é a entrada das palavras “Milton Friedman” (nesta ordem), para o conceito MONETARY-ARGUMENT é a entrada das palavras “increase tax” e para o critério MTRANS é a entrada de um *actor*. Finalmente, as palavras “says” e “info”.

3.2. Mecanismo de Busca

A seguir serão mostrados os passos do mecanismo de busca nos índices da memória a partir da entrada “Milton Friedman says increase tax”.

O sistema começa colocando um marcador de previsão no início de cada um dos índices, ou seja, para Mtrans-1 tem-se um marcador em (actor) que é um ponteiro para o conceito HUMAN. Este marcador de previsão foi colocado neste conceito para indicar que a ativação de Mtrans-1 só será satisfeita se ela começar com um HUMAN ou uma informação mais específica do mesmo. Na seqüência, para Friedman-1 a previsão é colocada em “Milton” e para Monetary-Argument-1 a previsão é colocada em “increase”.

A primeira palavra da entrada (“Milton”) é apresentada. O sistema percorre o índice e verifica que Friedman-1 tem uma previsão para esta palavra. O primeiro item do índice é satisfeito e o marcador de previsão neste índice é deslocado para o próximo item (“Friedman”).

Na continuação da frase de entrada, a palavra “Friedman” é apresentada e é encontrada uma previsão para esta palavra no índice Friedman-1. Ao tentar passar o marcador de previsão para o próximo item do índice, verifica-se que o índice em questão já foi reconhecido por completo e em consequência deste reconhecimento o conceito foi marcador como ativado. O sistema busca na hierarquia de Friedman-1 se existe alguma previsão para este índice e a encontra no conceito HUMAN. O índice Mtrans-1 tinha a previsão inicial para um *actor* e esta previsão foi satisfeita, passando assim para o próximo item deste índice (“says”).

A palavra “says” é apresentada na entrada e é encontrada uma previsão para esta palavra no índice Mtrans-1. Neste índice, a previsão é deslocada para o próximo item (info). As próximas duas palavras (“increase tax”), de maneira similar, irão satisfazer a previsão no índice Monetary-Argument-1 e como existe uma previsão para este índice em Mtrans-1, todas as previsões estão satisfeitas.

Os conceitos FRIEDMAN, MONETARY ARGUMENT e MONETARYCOMUNICATION foram reconhecidos na memória, acarretando, assim, a compreensão da frase de entrada.

Dois arquivos servem de repositório dos dados que serão carregados em memória. O primeiro arquivo contém as informações da hierarquia de conceitos. O segundo arquivo contém os índices e informações de empacotamento de conceitos.

Na Tabela 1, a coluna *Conceito* representa cada conceito semântico e a informação que possibilita navegar na hierarquia entre os mesmos. Cada elemento desta tabela é instância de uma classe, chamada *Conceito*, implementada na linguagem de programa Java™. A Tabela 2 define os índices para conceitos apresentados em Tabela 1.

A Tabela 3 mostra uma relação de conceitos e um ponteiro para o primeiro índice de cada um deles. Tal

tabela tem como função facilitar a busca dos índices na memória.

Tabela 1 - Exemplo de arquivo de conceitos

Conceito	Conceito Pai
MTRANS	MOBJECT
HUMAN	MOBJECT
MONETARY ARGUMENT	MOBJECT
MONETARIST	HUMAN
MILTON FRIEDMAN	MONETARIST
MONETARIST COMMUNICATION	MTRANS

Tabela 2 - Exemplo de arquivo de índices

Nome do Índice	Conceito	Índice
Mtrans-1	MTRANS	(actor) says (info)
Friedman-1	MILTON FRIEDMAN	Milton Friedman
Monetary-Argument-1	MONETARY ARGUMENT	increase tax

Tabela 3 - Conceitos e ponteiros.

Conceito	Próximo	PrimÍndice	AtualÍndice
MTRANS	MILTON FRIEDMAN	(actor)	(actor)
MILTON FRIEDMAN	MONETARY ARGUMENT	Milton	Milton
MONETARY ARGUMENT	Null	Increase	Increase

Finalmente, a Tabela 4 apresenta os índices dos conceitos, bem como a informação de previsão que será usada quando um empacotamento ocupar o lugar do texto no índice.

Tabela 4 - Exemplo de índices

Texto	Previsão	Próximo
(actor)	False	Says
Says	False	(info)
(info)	False	Null
Milton	False	Friedman
Friedman	False	Null
Increase	False	Tax
Tax	False	Null

Na Tabela 3 e 4 os atributos rotulados *próximo* servem para a realização da navegação no espaço de estados da memória. I.e., cada atributo *próximo* contém o endereço de memória do dado seguinte.

3.3. Parser Semântico

O algoritmo de *parser* semântico usado no DMAP envolve diversas etapas. Ele inicia percorrendo todos os índices de todos os conceitos. Nos índices que aparece a informação de empacotamento é colocado um *marcador de previsão*; a variável *previsão* recebe o valor *true*. Após este processo de

montagem do conjunto de dados de entrada para o sistema, os próximos passos a serem executados são:

Passo 1: para cada palavra do conjunto, busca-se nos índices uma palavra igual à entrada. Para os índices que satisfazem esta condição, o ponteiro *próximo* passa a apontar para o próximo índice;

Passo 2: caso o item do índice seja um empacotamento, um *marcador de previsão* deve ser colocado no conceito;

Passo 3: caso o índice seja percorrido por inteiro, isto significa que o conceito foi reconhecido. Neste momento, o conceito é armazenado no conjunto de conceitos *ativados*. Este conjunto será apresentado como saída do sistema;

Passo 4: no momento que um conceito foi reconhecido, o sistema deve verificar se existe algum índice com um *marcador de previsão* para este conceito (ou para um conceito mais abstrato);

Passo 5: este passo é o de maior complexidade do algoritmo. O processo de busca recebe o conceito que acaba de ser reconhecido e percorre todos os índices para verificar quem tem uma previsão para ele. Para o caso do índice estar apontando para um conceito com uma previsão, deve-se verificar se o conceito recém ativado é filho do conceito do índice que continha a previsão. Em caso afirmativo, significa que aquela previsão foi satisfeita e o item do índice foi reconhecido. Com isto, como no reconhecimento de um item do índice pela entrada de uma palavra, avança-se para o próximo item. O processo se repete, visto que, neste ponto, o empacotamento pode ser o último índice do conceito, ocorrendo assim o reconhecimento do mesmo. O conceito reconhecido é adicionado no conjunto de conceitos *ativados* e uma chamada recursiva é feita para este mesmo processo para verificar se existe outro conceito com uma outra previsão para o conceito que acaba de ser reconhecido.

A verificação do *marcador de previsão* deve ocorrer a partir do conceito reconhecido, subindo pela hierarquia por meio de suas abstrações.

Passo 6: no final deste processo são mostrados todos os conceitos *ativados*.

O que foi apresentado até o momento sobre o DMAP já permite propor uma metodologia para avaliação computacional do nível de compreensão de textos em linguagem natural.

4. A Metodologia

A validação da metodologia para avaliação computacional do nível de compreensão de textos foi realizada a partir de uma demanda na área educacional.

De uma maneira geral, os estudantes adquirem uma grande parte de seus conhecimentos por meio da leitura. Um grande desafio na área educacional é a verificação do conhecimento adquirido por meio da leitura. Muitas vezes o aluno lê um determinado texto, mas não se tem certeza do grau de compreensão obtido e também quais foram pontos essenciais assimilados. Os métodos tradicionais de avaliação como provas, resumos, etc., têm seu valor histórico e cultural, ou seja, acostumou-se a se atribuir uma nota representando o nível de entendimento de um determinado assunto por meio destes métodos. Com o avanço das pesquisas em computação na área da educação, têm-se desenvolvido softwares educacionais com o objetivo de trabalhar esta questão. Na sua maioria, os softwares são capazes de ensinar (por meio de tutoriais) e realizar a verificação da aprendizagem por meio de provas nos mais diversos formatos. É uma tarefa extremamente simples para estes softwares, conferir a quantidade de acertos e erros destas provas, avaliando, assim, o grau de entendimento do assunto abordado. Os avanços no processamento da linguagem natural estão contribuindo de forma substancial às questões tais como: a verificação de ortografia, a tradução de línguas e a confecção de sumários entre outros. A proposta é a aplicação de técnicas de compreensão da linguagem natural objetivando a verificação automática do nível de compreensão de um texto.

O experimento consiste em submeter resumos de um determinado texto ao sistema DMAP. De posse dos conceitos reconhecidos o sistema atribui um grau de compreensão ao resumo. Para isto, algumas métricas são definidas a fim de medir a abrangência do assunto e o encadeamento das idéias [13]. Estas métricas são definidas pelos seguintes parâmetros:

Grau de importância do conceito: valor numérico que determina o quanto à utilização daquele conceito é importante para um resumo. Varia de 0 (irrelevante) a 10 (essencial).

Grupo do conceito: usado para classificar os conceitos que se relacionam.

Número de seqüência do conceito no grupo: usado para medir o encadeamento das idéias dentro de um grupo de conceitos.

Grau de importância da seqüência dos conceitos no grupo: determina o quanto é importante que os conceitos apareçam em seqüência dentro do grupo.

Número de seqüência do grupo: usado para medir o encadeamento das idéias.

Grau de importância da seqüência dos grupos no resumo: determina o quanto é importante que os grupos apareçam em seqüência dentro do resumo.

No final será atribuída uma nota ao resumo utilizando-se a seguinte fórmula:

$$Nota = \left(\sum_{j=1}^{10} \left(\sum_{i=1}^k grauConceito_{ij} \right) + grauGrupo_x + (percSeqConceito_x * grauSeqConceito_x) \right) + (percSeqGrupo * grauSeqGrupo) \quad (1)$$

As figuras 2 e 3 mostram uma estrutura de memória contendo conceitos da área de enfermagem. Tal estrutura foi definida com a participação do tutor para este domínio. Da mesma forma, o tutor participou da definição dos valores para os parâmetros de avaliação dos resumos, atribuindo os graus de importância para os parâmetros de avaliação.

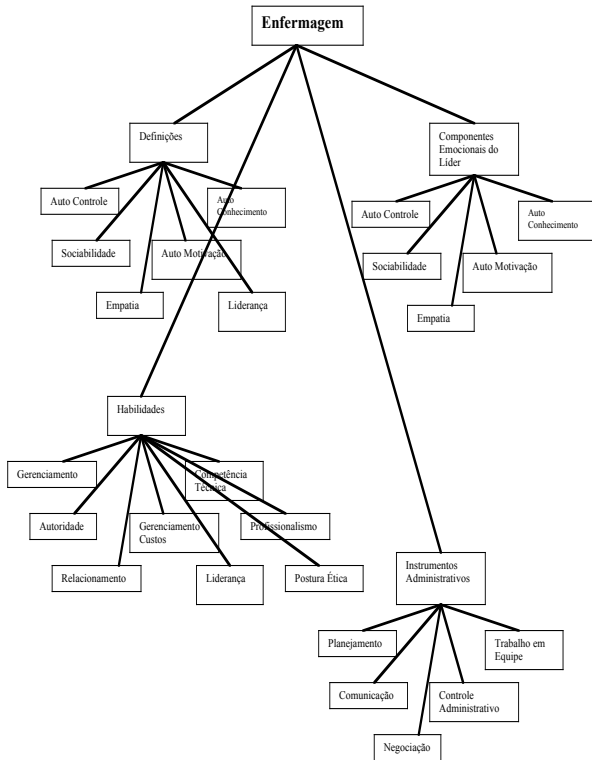


Figura 2 - Memória gerada com o auxílio do tutor – Parte I.

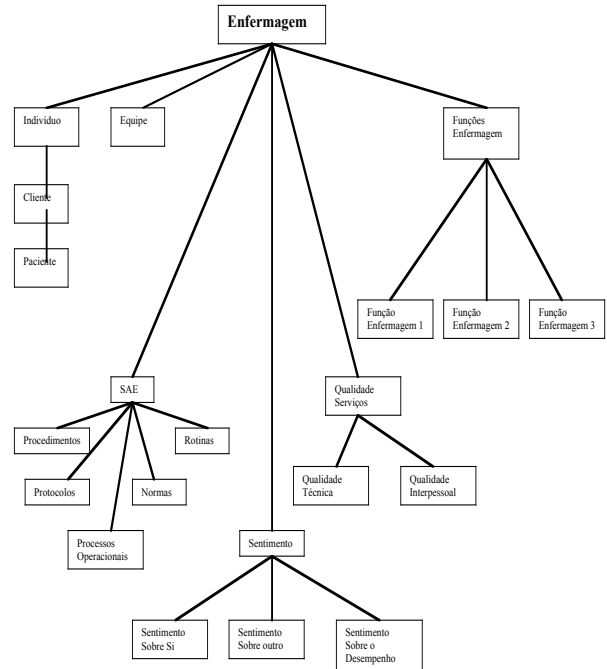


Figura 3 - Memória gerada com o auxílio do tutor – Parte II.

5. Experimento na Formação de Enfermagem

Este experimento teve como objetivo abordar um assunto técnico, longo e com uma grande quantidade de conceitos envolvidos. Desta forma, o tutor elaborou um texto longo (3129 palavras) e submeteu este texto aos seus 42 alunos do 8º período do curso de Enfermagem da Pontifícia Universidade Católica do Paraná (PUCPR). Cada aluno leu o texto e produziu um resumo com as idéias que ele considerou importantes. Cada resumo foi avaliado de forma independente pelo tutor e pelo sistema. O gráfico da Figura 4 mostra os graus atribuídos tanto pelo tutor como pelo sistema.

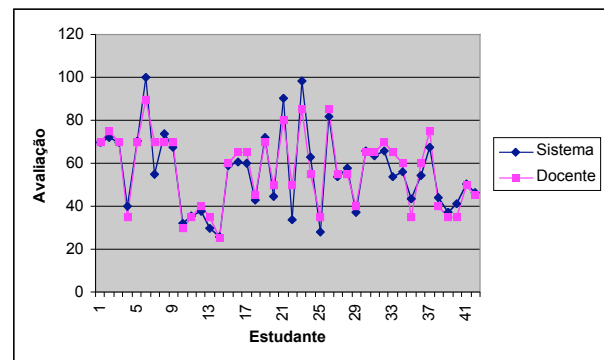


Figura 4 - Comparação entre os graus atribuídos pelo sistema e pelo docente.

A Figura 5 mostra a diferença por estudante, entre os graus atribuídos pelo sistema e pelo docente. A maior diferença foi observada para o estudante 22 (16,3%) e, se for considerado aceitável uma diferença de 10% entre o grau atribuído para cada resumo pelo sistema e pelo docente, tem-se, das 42 avaliações, apenas 5 delas (11,9%) tiveram uma diferença (para cima ou para baixo) acima deste valor.

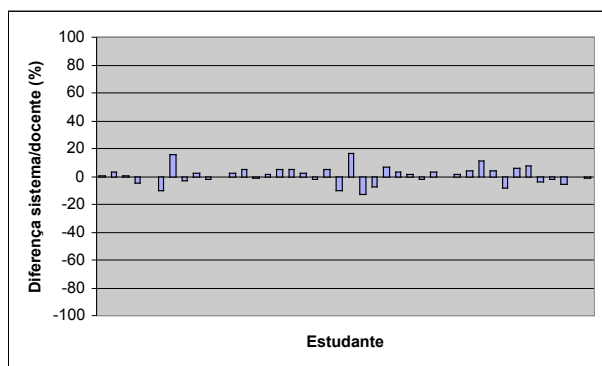


Figura 5 - Diferenças entre graus atribuídos pelo sistema e pelo docente.

Ao final da realização deste experimento, os comentários do docente envolvido foram os seguintes:

- **Quanto à forma de elaboração dos parâmetros para avaliação:** “A elaboração de parâmetros para a avaliação, num primeiro momento, é trabalhosa, porém facilita na análise face aos indicadores elaborados”.
- **Quanto aos graus atribuídos pelo sistema e pelo docente:** “Verificou-se que os resultados foram bastante similares. Tenho a observar que em algumas situações a análise do conteúdo da avaliação permitiu interpretar algumas questões no desenvolvimento do texto. Na elaboração de parâmetros para análise via sistema não se permite ou tem-se que acrescentar outros critérios”.
- **Avaliação final:** “Considero importante à alternativa de avaliação, visto que o apoio computacional facilita as atividades do docente”.

6. Discussão

A avaliação computacional de resumos de textos permite que muitos problemas sejam reduzidos no tocante a tarefa de medir o quanto foi entendido do texto. Deve-se destacar também a enorme rapidez para se avaliar um resumo por meio de um artefato computacional, onde o tempo de execução do procedimento de atribuição do grau de compreensão de um texto é praticamente desprezível para um resumo de uma página.

Algumas questões pertinentes à avaliação e comparação de resultados são resolvidas com esta metodologia. A primeira delas se refere à padronização das avaliações. Existe um grande problema na correção de grandes quantidades de textos produzidos, por exemplo, em redações de vestibular onde se faz uso de um comitê com vários professores para realizar a tarefa. A mesma redação não é avaliada por todos os professores do comitê e todos os participantes do comitê não avaliam todas as redações. Deve-se salientar que apesar dos critérios serem explícitos, pode haver uma divergência dependendo da visão de cada avaliador possui sobre tema. Esta metodologia faria com que todas as redações passassem por um processo único de avaliação. É claro que muitas questões não seriam contempladas pelo método, pois a interpretação do ser humano sempre trará mais requintes do que um artefato computacional, por mais aberto à parametrização que ele seja. Porém, este artefato poderia servir como um filtro inicial para descartar redações sem um mínimo de qualidade para participar da classificação final.

Outro ponto importante vislumbrado para o método de avaliação é a exploração de um ramo do ensino que está em franca ascendência, o Ensino Colaborativo à Distância. Com a possibilidade de o próprio sistema fazer a avaliação do estudo realizado na forma de leitura de textos e produção de resumos, repassando de forma rápida e quantitativa os resultados aos alunos. Graças à organização dos parâmetros da avaliação, o aluno pode ser informado de forma automática sobre o que ele deixou de considerar no seu resumo, por exemplo, quanto a encadeamento de idéias e abrangência do assunto.

O sistema fornece também informações que podem auxiliar o tutor a adequar o andamento do programa de aprendizagem. Por exemplo, o sistema pode identificar conceitos pouco citados pelos alunos em seus resumos, mas que possuem um alto grau de importância atribuído pelo tutor. É o caso do conceito “Norma” que foi citado apenas por 9,5% dos estudantes, cujo grau de importância é máximo neste caso. Desta forma, o tutor pode identificar claramente os *gaps* que precisam ser mais bem trabalhados. Este artefato pode ser utilizado para apoiar os estudantes de medicina na elaboração de diagnósticos e /ou processos terapêuticos. Assim, após avaliar um caso pré-apresentado pelo computador, o aluno realiza a anamnese computadorizada da mesma e ao final emite o diagnóstico. O computador terá condições de avaliar e alertar o aluno que determinado conceito não foi abordado na anamnese.

7. Conclusões

Este trabalho utilizou um paradigma para o tratamento de textos em linguagem natural. O DMAP é uma poderosa ferramenta para a compreensão de textos que se enquadrem

na sua condição de se ter uma memória modelada de conceitos, e que usa as definições de compreensão que tentam reproduzir a maneira como nós, seres humanos, realizamos tal tarefa. Pelas características deste método, é possível a manipulação desta memória de conceitos para diversas atividades, dentre elas a atribuição automática de um grau de compreensão.

Sabe-se que a pesquisa da compreensão da linguagem natural já vem de longos anos e muitos trabalhos já foram escritos tentando reproduzir a maneira com que compreendemos uma linguagem. O uso do DMAP como manipulador da memória modelada se aproxima da maneira como realizamos esta tarefa. Espera-se, então, com este trabalho, trazer uma contribuição para área de formação colaborativa analisando textos produzidos por estudantes, trazendo uma melhora na avaliação automática do conhecimento adquirido.

Referências

- [1] ÁVILA, B. C. Representação do Conhecimento utilizando Frames. Dissertação de Mestrado, Universidade de São Paulo, São Paulo, 1991.
- [2] CHENG, J., KUMAR, B., LAW, K. L. A question answering system for project management applications, *Advanced Engineering Informatics, Elsevier*, v. 16, pp. 277-289, 2002.
- [3] ENEMBRECK, F., BARTHÈS, J-P. Improving CSCW with Personnel Assistant Agents, In: *Journal of Integrated Design & Process Science*, (B. Kramer, J.P. Tsai, eds.), IOS Press, vol 7, n. 2, pp. 3-19, 2003, ISSN: 1092-0617.
- [4] ENEMBRECK, F., BARTHÈS, J-P. Dialog with a Personnel Assistant, XII Brazilian Symposium on Artificial Intelligence – SBIA'04, *Lecture Notes in Computer Science*, 2004.
- [5] ENEMBRECK, F., BARTHÈS, J-P. ELA – A new approach for Learning Agents, In: *Journal of Autonomous Agents and Multi-Agent Systems - JAAMAS*, Kluwer, vol 10, n. 3, pp 214-248, 2005. ISSN: 1387-2532
- [6] FREITAS, F. L. G., BITTENCOURT, G. An Ontology-based Architecture for Cooperative Information Agents, In: *Proceedings of IJCAI'03, Morgan Kaufmann Publishers*, pp. 37-42, Georg Gottlob and Toby Walsh (eds.), 2003. ISBN 0-127-05661-0
- [7] LIU, H., TANG, M., FRAZER, J. H. A Knowledge Based Collaborative Design Environment, In: *Agents in Design, Key Centre of Design Computing and Cognition*, University of Sydney, Australia, J. S. Gero and F. M. Brazier (eds.), 2002. ISBN 1 864 87 1202.
- [8] MARTIN, C. E. Direct Memory Access Parsing. *University of Chicago. Technical Report*, 1993.
- [9] MINSKY, M. The Society of Mind. Touchstone Book, New York, 1985.
- [10] RAMOND, B. Collaborative Design in Education: the Taxia Project, IDMMÉ'98, pp 1261-1268, Compiègne, 27-29 May 1998.
- [11] RIESBECK, C. K. "From Conceptual Analyzer to Direct Memory Access Parser: An Overview". *Excerpt from Advances in Cognitive Science*, 1986, pp. 236-258.
- [12] RIESBECK, C. K., SCHANK, R. C. Inside Case-Based Reasoning. *Lawrence Erlbaum Associates Publishers*, Hillsdale, New Jersey, 1989.
- [13] SÁNCHEZ MIGUEL, E. Compreensão e Redação de Textos. *Artmed*, Porto Alegre, 2002.
- [14] SCHANK, R. C. Scripts, Plans, Goals and Understanding – An Inquiry into human knowledge Structures. *Lawrence Erlbaum Associates Publishers*, Hillsdale, New Jersey, 1977.
- [15] SCHANK, R. C. Riesbeck, C. K. Inside Computer Understanding. *Lawrence Erlbaum Associates Publishers*, Hillsdale, New Jersey, 1981.
- [16] SCHANK, R. C. Dynamic Memory Revisited. *Cambridge University Press*, Cambridge, UK, 1999.
- [17] SHEN, W., NORRIE, D. H.; BARTHÈS, J-P. Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing, *Taylor & Francis*, UK, 2001. ISBN 0-7484-0882-7
- [18] QUILIAN R. M. Word concepts: a theory and simulation fo some basic sematic capabilities. *Behavioral Science*, 12:410--30, 1967.
- [19] TACLA C. A. BARTHÈS J-P. A., Multi-agent Architecture for Knowledge Acquisition, AAI Spring Symposium, March 24-26, *Stanford University - Technical Report SS-03-01*, L. van Elst, V. Dignum, A. Abecker (eds), AAI Press, Menlo Park, ISBN 1-57735-178-9 SS-03-01, p. 159-166, 2003.
- [20] THOUVENIN, I., ABEL, M-H., RAMOND, B., QAMHIYAH, A. Environment improvements for a better cooperation in multi-culture collaborative mechanical design, In: *Journal of Integrated Design & Process Science*, (B. Kramer, J.P. Tsai, eds.), IOS Press, vol 7, n. 2, pp. 131-142, 2003, ISSN: 1092-0617.