Alinhamento sentencial de textos paralelos português-inglês

Helena de Medeiros Caseli

Alinhamento sentencial de textos paralelos português-inglês

Helena de Medeiros Caseli

Orientador: Prof. Dr. Maria das Graças Volpe Nunes

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências da Computação.

USP – São Carlos
Fevereiro/2003

Agradecimentos

Aos meus pais pelo grande amor que têm por mim e por me fazerem feliz desde meu primeiro dia de vida.

Ao meu irmão, minha outra metade, pela amizade nos anos de faculdade e pelo exemplo de vida.

Ao Fer, meu melhor amigo e meu grande amor, pelo apoio nos momentos mais difíceis e por frases do tipo “se você não conseguir não sei quem conseguirá”.

Às minhas grandes amigas que mesmo à distância continuam sempre presentes: Paula, Cadô, Patrícia, Lecy, Aninha e Karina.

À Graça por me oferecer mais do que sua orientação e por me cobrar e acalmar na medida certa.

Aos amigos do NILC presentes e distantes em especial à Marcela pela prontidão com que sempre atendeu aos meus pedidos de revisão de artigos; à Mônica por me ajudar no desenvolvimento deste trabalho; e ao André por sempre resolver os problemas do meu micro.

Aos amigos do mestrado e dos laboratórios Labes, Labic e Intermídia. Em especial aos intermídia *boys* e às amigas de academia.

Às amizades que cativei e consolidei em Porto de Galinhas onde pude me apaixonar mais ainda pelo meu trabalho de pesquisadora. Em especial à Deinha, ao Thiago e aos amigos das madrugadas e nascer do sol Marcos Geromini, Daniel e Marcelo.

Às professoras Carolina, Lúcia Rino, Sandra e Solange, pela atenção dispensada e pelos momentos de descontração. Aos professores da graduação Sérgio Schneider e Márcia Fernandes, grandes mestres e amigos.

A CAPES pelo apoio financeiro, ao NILC e à USP pelas instalações.

Às secretárias, aos porteiros, às faxineiras e aos amigos da cantina pela atenção e descontração do dia a dia.

Enfim, a cada pessoa que nestes dois anos cruzou o meu caminho me apoiando, me incentivando ou simplesmente me ouvindo ...

muito obrigada!

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Quadros

Resumo

Esta dissertação relata o primeiro trabalho de pesquisa em alinhamento automático de textos paralelos envolvendo o português brasileiro (PB). Neste trabalho foram implementados cinco métodos de alinhamento sentencial automático bastante referenciados na literatura, incluindo métodos empíricos, lingüísticos e híbridos, avaliados com textos paralelos PB-inglês. Os resultados mostraram-se compatíveis com os relatados para outros pares de línguas, sendo que as maiores precisões (acima de 94%) foram obtidas em corpora sem ruídos (sem erros gramaticais e de tradução), conforme era esperado. Além disso, os resultados apontam muita semelhança no desempenho de todos os métodos, o que impossibilita a eleição de um deles como o melhor. Além da implementação dos métodos de alinhamento sentencial e dos corpora paralelos construídos para avaliá-los, outros recursos lingüísticos e computacionais de grande valor para as pesquisas em PLN foram gerados durante este trabalho.

Abstract

As the first attempt at automatic parallel text alignment involving Brazilian Portuguese, in this research we implemented five well-known automatic sentence alignment methods, including empirical, linguistic and hybrid techniques, and evaluated them as applied to Brazilian Portuguese-English parallel texts. The results are in accordance with those reported for other pairs of languages, even in that highest precisions (above 94%) were obtained for corpora without noise (i.e. grammatical or translation errors), as expected. Furthermore, the results point to a virtual tie between the methods, it being impossible to elect one as the best. In addition to the implementations of those methods and the parallel corpora built to evaluate them, other linguistic and computational resources were built during this work which are of great value to PLN research.

Capítulo 1

Introdução

No cenário mundial atual, de rápida expansão nas relações interculturais e na transmissão de conhecimento científico e tecnológico, a língua muitas vezes representa uma barreira para a comunicação.

Apesar da disseminação da língua inglesa como língua franca nos diversos setores da atividade humana, seu aprendizado requer considerável período de tempo e, para um número significativo de indivíduos, é ainda bastante oneroso. Além disso, o número de profissionais qualificados para transpor essa barreira lingüística – tradutores e intérpretes – não cresce na mesma proporção em que cresce a demanda por troca de informações entre línguas diferentes nos variados domínios do conhecimento humano. Sendo assim, faz-se necessária a utilização de ferramentas computacionais que facilitem e acelerem a comunicação escrita e oral entre povos de línguas diversas, a fim de superar essa barreira.

Há tempos essa superação constitui objeto de interesse de uma área da Ciência da Computação, mais especificamente da área de Processamento de Línguas Naturais, na qual se insere o assunto desse trabalho: *o Alinhamento Sentencial de Textos Paralelos*.

Textos paralelos, segundo a terminologia estabelecida pela comunidade de lingüística computacional, são textos acompanhados de sua tradução em uma ou várias línguas. São considerados distintos dos textos sobre um mesmo tópico, escritos em línguas diferentes, mas que não são necessariamente traduções mútuas: os *textos comparáveis*.

Os textos paralelos, também chamados de *bitextos* quando apenas duas línguas estão envolvidas, são fontes ricas de conhecimento lingüístico. Isso porque a tradução de um texto para uma outra língua pode ser entendida como uma anotação detalhada do significado do texto original. Se, além de paralelos, os textos forem alinhados, isto é, possuírem marcas que identifiquem os pontos de correspondência entre o texto original e sua tradução, o conhecimento daí derivado (as equivalências de tradução) assume importância capital em muitas aplicações, como as traduções humana e automática e a recuperação de informação entre línguas diferentes.

Embora o alinhamento de textos paralelos e a própria denominação “textos paralelos” sejam relativamente novos, textos acompanhados de suas traduções não o são. Um dos exemplos mais antigos de que se tem registro é a pedra de Rosetta, encontrada em julho de

1799 por um oficial do exército de Napoleão perto da cidade de Rosetta, no delta do Nilo. A pedra de Rosetta, datada de 196 a.C., é uma estela que contém um decreto de Ptolomeu V em duas línguas (grego e egípcio) e três sistemas de escrita (o egípcio é apresentado em escrita demótica e também em hieróglifos).

A pesquisa sobre textos paralelos alinhados teve início no final da década de 50, com as primeiras tentativas de utilização de textos paralelos na tradução automática. Porém, muitos problemas encontrados nessa época contribuíram para o uso restrito desses recursos, como a capacidade limitada de armazenamento e processamento dos computadores, e a dificuldade de digitalização de grandes quantidades de textos.

Mais recentemente, em 1987, o primeiro método de alinhamento de textos paralelos foi apresentado à comunidade científica por Martin Kay e Martin Röscheisen (Véronis, 2000). Esse método é considerado geral (independente da língua), empírico e de baixo custo, pois alinha dois textos paralelos encontrando os pontos de correspondência entre eles sem levar em consideração informações específicas a respeito das línguas envolvidas. Nesse processo são utilizadas apenas informações sobre a distribuição das palavras nos dois textos e são considerados pontos de correspondência aquelas com distribuições similares a um dado nível de probabilidade limite.

De forma geral, o processo de alinhamento de textos paralelos consiste na identificação de correspondências entre o texto original (texto fonte) e sua tradução (texto alvo). Essas correspondências podem se dar em diferentes níveis: do nível do documento completo ao nível de suas partes componentes (capítulos, seções, parágrafos, sentenças, palavras e, finalmente, caracteres). O nível em que se dá a correspondência é denominado *nível de resolução*.

O nível de resolução de um método de alinhamento de textos paralelos é utilizado para caracterizá-lo. Essa caracterização, contudo, não impede o uso conjunto de métodos com níveis de resolução diferentes, apenas especifica qual é o objetivo principal. Assim, um método de alinhamento sentencial pode utilizar em seu processo de execução um método de alinhamento de palavras, mas o objetivo principal continuará sendo alinhar sentenças e não palavras.

O processo de alinhamento de textos paralelos pode ser subdividido em várias etapas de acordo com o nível de resolução utilizado. De modo geral, o processo se baseia na determinação dos pontos de correspondência candidatos, ou seja, dos pontos com base nos quais o alinhamento poderá ser feito, e na filtragem desses pontos, isto é, na eliminação

daqueles que apresentam pouca probabilidade de representar no texto alvo a tradução correspondente do texto fonte.

Nesse contexto está inserido o projeto aqui descrito: o PESA (*Portuguese-English Sentence Alignment*) ou alinhamento sentencial português-inglês. O projeto PESA foi desenvolvido no *Núcleo Interinstitucional de Lingüística Computacional* (NILC), um grupo interdisciplinar dedicado à pesquisa e ao desenvolvimento de sistemas de PLN, criado em 1993. Esse grupo de pesquisadores de lingüística e computação tem desenvolvido recursos e aplicativos para o processamento do português brasileiro (PB). Destacam-se o projeto ReGra: revisor gramatical automático do português brasileiro, apoiado pela FAPESP, CNPq, Finep e Itautec-Philco S.A., que originou um produto comercializado pela Itautec e também distribuído com o MS-Office (português) desde 2000; o projeto de um *thesaurus* do português brasileiro e da base de dados lexical Diadorim; o projeto *Universal Networking Language* (UNL), patrocinado pelo Instituto de Estudos Avançados da Universidade das Nações Unidas, para o qual o NILC constrói ferramentas de codificação e decodificação de português-UNL; o projeto de construção de ferramentas de auxílio à escrita técnica em português, apoiado pela FAPESP, entre outros[1].

Os projetos do NILC mais relacionados ao projeto PESA são: o EPT-Web[2], desenvolvido com o apoio do CNPq com o objetivo de implementar um tradutor inglês-português de páginas da Web que utilize a interlíngua UNL para traduzir as primeiras páginas da versão eletrônica do jornal norte-americano "The New York Times"; e o Lacio-Web[3] cujo objetivo é criar recursos lingüísticos e computacionais (como corpora e ferramentas associadas) para aplicações de processamento do PB. O corpus construído para o projeto PESA faz parte de um conjunto de corpora que comporão o ambiente computacional do projeto Lacio-Web.

As próximas seções apresentam a motivação (1.1) e os objetivos (1.2) do projeto PESA. A última Seção (1.3) mostra como este texto está estruturado.

1.1 Motivação

O alinhamento de textos paralelos é uma das áreas de Processamento de Línguas Naturais que mais cresce atualmente devido, principalmente, ao grande número de aplicações para as quais

[1] Outras informações sobre o NILC e seus projetos podem ser obtidas em http://www.nilc.icmc.usp.br.
[2] Em: http://www.nilc.icmc.usp.br/nilc/projects/ept-web.htm (18/02/2003).
[3] Em: http://www.nilc.icmc.usp.br/nilc/projects/lacio-web.htm (18/02/2003).

pode ser útil. Entre essas aplicações, podem-se citar: as memórias de tradução; a recuperação de informações através da troca de dados entre línguas diferentes; a tradução automática propriamente dita; a construção de dicionários bilíngües; a extração de terminologia de textos técnicos; o esclarecimento de ambigüidade; e o aprendizado de idiomas. Cada uma dessas aplicações será brevemente introduzida a seguir.

As memórias de tradução são ferramentas computacionais que tentam evitar a tradução desnecessária de segmentos de texto previamente traduzidos. Isso é feito através da consulta e recuperação automática das informações contidas nesses segmentos em um banco de dados que, muitas vezes, é construído com base em textos paralelos alinhados. Como exemplos de memórias de tradução comercialmente disponíveis atualmente têm-se: Trados[4], Transit, Déja Vu[5], XL8, Eurolang, Catalyst e SDLX, desenvolvidas por diversas empresas de tradução americanas e européias (Melby, 2000).

Recentemente, o sistema líder de mercado, Trados, começou a ser ameaçado por um concorrente distribuído gratuitamente, o Word Fast[6]. Esse sistema, desenvolvido pelo francês Yves Champollion, une as tecnologias de memória de tradução e de tradução automática em um processo de duas etapas. Primeiro, o Word Fast faz uma busca pela expressão ou palavra na memória e, caso não a encontre, o tradutor automático oferece uma sugestão de tradução.

A recuperação de informações através da troca de dados entre línguas diferentes é um mecanismo que ganhou muita importância com a disseminação do uso da Internet. É muito utilizado em buscas na *web* quando a consulta é feita em uma determinada língua e o resultado é apresentado em outra ou diversas outras.

A tradução automática se beneficia do alinhamento de textos paralelos no que diz respeito à aquisição automática de conhecimento: dicionário, padrões e regras de tradução.

A construção de dicionários bilíngües, bem como a extração de terminologia de textos técnicos, são aplicações bastante favorecidas pelo alinhamento de textos paralelos. Ambas são consideradas tarefas difíceis e que demandam muito tempo para serem concluídas, mas podem se tornar menos onerosas com o auxílio de técnicas de alinhamento. Além disso, o tipo de conhecimento que geram pode ser utilizado por todas as outras aplicações já citadas.

O esclarecimento de ambigüidade se refere a um tipo comum de ruído gerado na tradução que, na maioria dos casos, está presente em apenas uma das línguas envolvidas na

[4] Em: http://www.trados.com/ (17/02/2003).
[5] Em: http://www.atril.com/ (17/02/2003).
[6] Em http://www.champollion.net (17/02/2003).

comunicação. Utilizando-se o alinhamento de textos paralelos, é possível recuperar informações sobre a língua que esclareçam a ambigüidade resultante da tradução.

O aprendizado de idiomas é outra aplicação que se beneficia do alinhamento de textos paralelos, pois os bancos de dados gerados a partir dos resultados desse alinhamento são valiosos para o aprendiz de uma língua estrangeira.

Além da motivação decorrente da grande utilidade dos métodos de alinhamento de textos paralelos e de seus resultados (textos paralelos alinhados, léxicos bilíngües, etc.), há ainda uma motivação especial em relação às línguas envolvidas na construção do corpus: não há conhecimento de publicações na área de alinhamento de textos paralelos, em qualquer nível de resolução, envolvendo o português brasileiro.

Os trabalhos publicados fazem referência apenas ao português europeu, sem levar em consideração o brasileiro. Por exemplo, dentro do projeto "Processamento Computacional do português", desenvolvido pelo SINTEF[7], um corpus de textos inglês-português europeu foi utilizado para testar e avaliar um método híbrido de alinhamento sentencial, o *Translation Corpus Aligner* (TCA) (Santos & Oksefjell, 2000).

Os trabalhos de Ribeiro et al (2000a, 2000b) também utilizam o português europeu. Neles, um método empírico de alinhamento de segmentos delimitados por dois pontos de correspondência é avaliado em relação a um corpus composto por textos legislativos paralelos selecionados aleatoriamente do Jornal Oficial das Comunidades Européias e do Tribunal de Justiça das Comunidades Européias, redigidos nas onze línguas oficiais da União Européia, entre elas o português europeu.

1.2 Objetivos

O principal objetivo deste trabalho inclui o estudo, a implementação e a avaliação das principais técnicas de alinhamento sentencial de textos paralelos e, se fosse possível, a eleição de uma delas como a melhor. Dessa forma, tem-se como submeta deste trabalho a construção de um alinhador automático de textos bilíngües PB-inglês cuja precisão esteja de acordo com o estado da arte.

A metodologia utilizada para alcançar o objetivo citado inclui diversas tarefas como: a) o estudo de técnicas e metodologias de alinhamento sentencial de textos paralelos; b) a implementação de uma ou mais técnicas pertencentes a cada tipo de metodologia existente; c)

[7] Em http://www.portugues.mct.pt (18/02/2002).

a construção dos corpora de teste e de referência, além de outros recursos lingüísticos e d) a avaliação das técnicas implementadas segundo critérios de avaliação reportados na literatura.

1.3 Organização do Texto

O presente texto está organizado conforme o que se segue.

No Capítulo 2 é apresentado um breve histórico sobre a área de alinhamento de textos paralelos e, mais especificamente, o alinhamento sentencial, foco principal do projeto PESA.

O Capítulo 3 descreve o processo de construção dos recursos lingüísticos utilizados no projeto PESA e especifica a função de cada um deles com relação aos métodos implementados.

Os próximos três capítulos apresentam em detalhes os métodos implementados com ênfase no processo de alinhamento e nas características de cada um. O Capítulo 4 descreve os métodos empíricos, o Capítulo 5, o método lingüístico e o Capítulo 6, os métodos híbridos.

O Capítulo 7 descreve a metodologia utilizada na avaliação dos métodos de alinhamento sentencial bem como os resultados dessa avaliação, organizados de acordo com a classe a qual pertencem: métodos empíricos (Seção 7.1), método lingüístico (Seção 7.2) e métodos híbridos (Seção 7.3), nesta ordem.

Finalmente, o Capítulo 8 traz algumas considerações finais sobre esse trabalho e o Apêndice A apresenta os resultados da avaliação dos métodos de alinhamento sentencial implementados no projeto PESA com outro corpus, o corpusALCA.

Capítulo 2

Alinhamento Sentencial de Textos Paralelos

A pesquisa sobre textos paralelos alinhados teve início no final da década de 50, mas só ganhou força no final da década de 80, devido, principalmente, aos avanços tecnológicos e suas conseqüências nas funcionalidades computacionais, como uma maior capacidade de armazenamento e processamento. Outro fato importante que estimulou a pesquisa nessa área foi o crescimento dramático da disponibilidade de textos acompanhados de suas respectivas traduções com o advento da Internet. Alguns exemplos de sites com textos paralelos são os sites oficiais da Área de Livre Comércio das Américas (ALCA)[8] e do Mercado Comum do Sul (MERCOSUL)[9], com versões de seus documentos oficiais em PB e em inglês, por exemplo.

Como resultados desse esforço de pesquisa surgiram muitos métodos para o alinhamento automático de textos paralelos baseados em critérios de alinhamento variados, como a similaridade de tamanho ou carga semântica das sentenças a serem alinhadas[10].

Esses critérios são utilizados para classificar os métodos de alinhamento sentencial de textos paralelos em: métodos empíricos, métodos lingüísticos ou métodos híbridos. Os métodos empíricos são aqueles que não utilizam qualquer tipo de informação lingüística em seu processo de alinhamento, apenas informações estatísticas, como a freqüência de ocorrência de palavras e/ou sua distribuição no texto e técnicas de reconhecimento de padrão para determinar se duas palavras com grafias similares podem ser consideradas traduções mútuas. Os métodos lingüísticos, por sua vez, utilizam recursos lingüísticos específicos para as línguas envolvidas, como léxicos, listas de palavras âncoras[11] e glossários. Por fim, os métodos híbridos englobam as duas abordagens anteriores, utilizando os recursos dos métodos empíricos e lingüísticos conjuntamente.

---

[8] Em: http://www.ftaa-alca.org/alca_e.asp (17/02/2003).
[9] Em: http://www.mercosur.org.uy (17/02/2003).
[10] A carga semântica de uma sentença, nesse caso, pode ser entendida como a quantidade de substantivos, verbos, adjetivos e advérbios presentes nessa sentença.
[11] Listas bilíngües nas quais uma palavra na língua fonte é acompanhada de uma ou mais traduções para a língua alvo.

Outra característica dos métodos de alinhamento sentencial está relacionada às categorias de alinhamentos que eles consideram. A categoria mais comum é a 1-1 na qual uma sentença fonte corresponde a exatamente uma sentença alvo; porém outras categorias são possíveis como: 0-1, 1-0, 1-2, 2-1, 2-2, etc. A maioria dos métodos de alinhamento sentencial considera seis casos: remoção (1-0), inserção (0-1), substituição (1-1), expansão (1-2), contração (2-1) e união (n-m, com n,m >1).

Entre as características desejáveis em todos os métodos de alinhamento automático de textos paralelos pode ser citada a de serem computacionalmente tratáveis e eficientes, mantendo a complexidade linear em relação ao tamanho dos textos. Os dados não-textuais, como tabelas e figuras, que podem representar obstáculos no processo de alinhamento devem ser previamente tratados a fim de garantir simplificação e melhoria de desempenho. Além dessas, outras duas características desejáveis são a escalabilidade e a confiabilidade. A última é óbvia, ao passo que, por escalabilidade entendem-se as arquiteturas de alinhamento modulares e extensíveis, permitindo, assim, acomodar melhorias e necessidades futuras.

Quanto à metodologia de alinhamento sentencial, dois grupos de estudos iniciais se destacam: o grupo de Kay e Röscheisen (1988, 1993) e o grupo de Gale e Church (1991, 1993) e Brown et al. (1991), todos citados por Jean Véronis (Véronis, 2000).

O primeiro grupo levantou a hipótese de que, se duas sentenças são correspondentes dentro de textos paralelos, então, necessariamente, as palavras que as formam também são correspondentes. Assim, as palavras presentes nos dois textos são alinhadas entre si e, em seguida, tomando os pares de palavras resultantes como pontos de correspondência, o alinhamento sentencial é produzido. Embora a tarefa de alinhar palavras seja considerada mais difícil do que a de alinhar sentenças, tal hipótese tem como base o fato de que um alinhamento imperfeito de palavras pode resultar em um alinhamento sentencial satisfatório.

O segundo grupo baseou-se na suposição de que as sentenças do texto fonte e suas correspondentes traduções têm tamanhos semelhantes, assim, sentenças pequenas teriam traduções pequenas e sentenças grandes, traduções também grandes. O tamanho das sentenças, nesse caso, pode ser calculado em relação ao número de caracteres ou de palavras presentes nelas.

Esses dois grupos serviram e até hoje servem de base para a maioria dos métodos nessa área. Um exemplo é o método híbrido apresentado recentemente à comunidade científica por Moore (2002). Tal método alinha as sentenças baseando-se nas correspondências de palavras presentes nelas e no método empírico baseado em tamanho proposto em (Brown et al., 1991).

Alguns métodos de alinhamento propostos recentemente também apresentam uma característica nova: a tentativa de lidar com o problema de re-otimização de parâmetros para cada novo par de línguas. Nesses métodos o processo de re-otimização é feito simultaneamente ao processo de alinhamento. Chuang et al., por exemplo, propôs um alinhamento em várias etapas no qual os parágrafos são alinhados e a partir desse alinhamento é feita uma estimativa para o alinhamento sentencial (Chuang et al., 2002).

Apesar da diversidade de critérios dos métodos de alinhamento sentencial de textos paralelos, o processo de alinhamento é feito em basicamente duas etapas: determinação de pontos de correspondência candidatos nas sentenças fonte e alvo e filtragem desses pontos. Os pontos de correspondência são determinados de acordo com o critério de alinhamento empregado pelo método. A filtragem seleciona os pontos de correspondência candidatos que mais se aproximam da tradução real de acordo com parâmetros específicos de cada método.

Formalmente, um alinhamento sentencial pode ser definido como:

Dado um texto A e sua tradução B, considerados como dois conjuntos de sentenças A = $a_1, a_2, ..., a_m$ e B = $b_1, b_2, ..., b_n$, um alinhamento X_{AB} entre A e B pode ser definido como um subconjunto do produto cartesiano 2^A X 2^B, em que 2^A e 2^B são, respectivamente, o conjunto de todos os subconjuntos de A e de B.

Por exemplo, considere o alinhamento apresentado no Quadro 1 no qual as sentenças do texto fonte (A) aparecem à esquerda e as sentenças do texto alvo (B), à direita.

Quadro 1: Exemplo de alinhamento sentencial.

	A		B
a_1	Tal abordagem parte do documento de requisitos do sistema e propõem a especificação das interações entre o sistema e seus agentes (cenários), e então os cenários são especificados detalhadamente.	**b_1**	This approach starts from the system's requirement document and proposes to specify interactions between the system and its agents (scenarios), and then the scenarios are specified in detail.
a_2	Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.	**b_2** **b_3**	Heuristics to evolve from the requirements model to the analysis are also presented. An example to illustrate the approach is also presented.

A representação formal desse alinhamento é dada por: X_{AB} = {({a_1}, {b_1}), ({a_2}, {b_2, b_3})}.

Porém, essa forma de representação, baseada no alinhamento e não nas sentenças que o formam, não considera alinhamentos parcialmente corretos como o apresentado no Quadro 2, representado formalmente como Y_{AB} = {({a_1}, {b_1}), ({}, {b_2}), ({a_2}, {b_3})}.

Quadro 2: Exemplo de alinhamento sentencial parcialmente correto.

	A		B
a_1	Tal abordagem parte do documento de requisitos do sistema e propõem a especificação das interações entre o sistema e seus agentes (cenários), e então os cenários são especificados detalhadamente.	b_1	This approach starts from the system's requirement document and proposes to specify interactions between the system and its agents (scenarios), and then the scenarios are specified in detail.
		b_2	Heuristics to evolve from the requirements model to the analysis are also presented.
a_2	Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.	b_3	An example to illustrate the approach is also presented.

Desta forma, uma representação baseada em sentenças será adotada no restante deste texto para possibilitar que alinhamentos parcialmente corretos sejam considerados. Nela, assume-se $X_{AB} = \{x_1, x_2, ..., x_m\}$ e $Y_{AB} = \{y_1, y_2, ..., y_n\}$, com $x_i = (x_{a,i}, x_{b,i})$ e $y_j = (y_{a,j}, y_{b,j})$, e têm-se X'_{AB} = união$_i$ $(x_{a,i}$ X $x_{b,i})$ e Y'_{AB} = união$_j$ $(y_{a,j}$ X $y_{b,j})$, onde "união"é a operação de união utilizada em teoria dos conjuntos.

Com a nova representação, os alinhamentos anteriormente descritos podem ser definidos como: $X'_{AB} = \{(a_1, b_1), (a_2, b_2), (a_2, b_3)\}$ e $Y'_{AB} = \{(a_1, b_1), (a_2, b_3)\}$, para X_{AB} e Y_{AB}, respectivamente[12]. Agora sim, o alinhamento parcialmente correto (a_2, b_3) pode ser considerado.

A próxima Seção (2.1) descreve o projeto de mestrado ao qual este texto se refere – o projeto PESA – e suas características.

2.1 O Projeto PESA: Alinhamento Sentencial português-inglês

O projeto descrito neste texto recebeu a denominação de PESA (*Portuguese-English Sentence Alignment*) ou Alinhamento Sentencial português-inglês. Como o próprio nome já diz, trata-se do estudo de métodos de alinhamento sentencial de textos paralelos escritos em português (do Brasil – PB) e em inglês.

[12] Relatório do projeto ARCADE disponível em http://www.lpl.univ-aix.fr/projects/arcade/report1-en/report1.html (18/02/2003).

A escolha do alinhamento sentencial para estudo nesse trabalho baseou-se na constatação de que técnicas com esse tipo de alinhamento possuem maior precisão em relação às técnicas de alinhamento de palavras (os dois níveis de resolução mais pesquisados na atualidade). Como embasamento teórico para essa escolha tem-se os resultados apresentados pelo projeto ARCADE[13] em uma avaliação de sistemas de alinhamento de textos paralelos nos níveis de sentença e palavra. Esses resultados mostram que o alinhamento sentencial, para os doze sistemas testados, foi bastante satisfatório, alcançando uma precisão acima de 95%, ao passo que o alinhamento de palavras alcançou apenas 75%, nos cinco sistemas participantes (Véronis & Langlais, 2000). Além disso, esse último tipo de alinhamento apresenta dificuldades especiais de tratamento consideradas sofisticadas para um primeiro trabalho sobre o assunto.

A escolha do par de línguas PB-inglês para estudo nesse trabalho foi motivada pelos seguintes fatores relacionados ao português brasileiro: a possibilidade de utilização dos recursos gerados em trabalhos futuros derivados (como tradução automática e construção de glossários) e a ausência de pesquisas na área de alinhamento de textos paralelos envolvendo o PB. Já a escolha do idioma inglês para a composição dos textos paralelos deve-se à sua posição de língua franca em vários cenários (científico, negócios, da *web*, etc.) e à disponibilidade de material digitalizado envolvendo as duas línguas. Esse material é constituído de resumos e *abstracts* de trabalhos na área de computação do Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (ICMC-USP) em São Carlos.

Cinco métodos foram selecionados para estudo no projeto PESA: dois métodos empíricos, um método lingüístico e dois métodos híbridos. Os métodos empíricos baseiam-se em critérios de alinhamento diferentes e são representantes dos dois grupos de estudo iniciais citados anteriormente: o método de Gale e Church (1991, 1993) baseia-se no tamanho das sentenças para alinhá-las, enquanto o método *Geometric Mapping and Alignment* (GMA) (Melamed, 2000) alinha as sentenças baseando-se nas palavras que as formam. Esses métodos serão referenciados no restante deste texto como GC (iniciais de seus autores) e GMA, respectivamente.

---

[13] O projeto ARCADE é um dos vários projetos de avaliação de processamento de línguas naturais e de fala iniciados e financiados pela rede de universidades francófonas (AUPELF-UREF). Um dos principais objetivos do projeto ARCADE é contribuir para o desenvolvimento de uma metodologia de avaliação de sistemas de alinhamento de textos paralelos nos níveis de sentença e de palavra, para melhor compreender as dificuldades dessas tecnologias e, assim, tentar melhorá-las. Homepage do projeto: http://www.lpl.univ-aix.fr/projects/arcade/index-en.html.

O método lingüístico, por sua vez, alinha as sentenças considerando suas cargas semânticas, ou seja, a quantidade de substantivos, verbos, adjetivos e advérbios (Papageorgiou et al., 1994; Piperidis et al., 2000). Por ser o único representante da classe de métodos lingüísticos, este método será referenciado no decorrer deste texto simplesmente como "método lingüístico".

Por fim, os métodos híbridos também possuem critérios de alinhamento diversos entre si. O primeiro é uma extensão do método empírico GMA que utiliza um recurso lingüístico como um de seus critérios de alinhamento. Este método recebeu a denominação de GSA+ em uma avaliação efetuada no projeto ARCADE e será referenciado da mesma forma neste texto. O segundo método híbrido é o *Translation Corpus Alinger* (TCA) proposto por Hofland (1996). O TCA utiliza diversos critérios de alinhamento como o tamanho das sentenças, cognatos, nomes próprios, entre outros.

A Figura 1 traz um diagrama com os métodos escolhidos para a implementação no projeto PESA, agrupados de acordo com as três grandes classes de métodos de alinhamento sentencial de textos paralelos: empíricos, lingüísticos e híbridos.

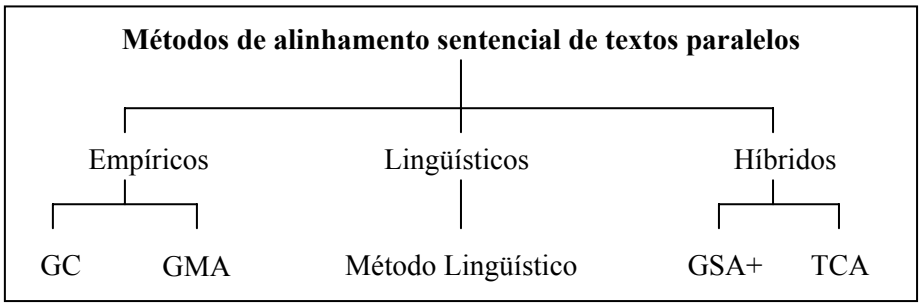

Figura 1 – Métodos de alinhamento sentencial escolhidos para a implementação no projeto PESA.

Esses métodos serão apresentados nos capítulos que se seguem, de acordo com as classes as quais pertencem. O Capítulo 4 traz os métodos empíricos; o Capítulo 5, o método lingüístico; e o Capítulo 6, os métodos híbridos.

Para que estes métodos pudessem ser implementados e avaliados fez-se necessária a construção de alguns recursos lingüísticos os quais são apresentados no próximo capítulo (Capítulo 3).

Capítulo 3

Recursos Lingüísticos

A utilização de recursos lingüísticos em projetos na área de Processamento de Línguas Naturais (PLN), em muitos casos, é indispensável e, na maioria das vezes, exige um esforço de construção bastante grande. Diversos projetos em PLN que lidam, por exemplo, com análise sintática, correção gramatical ou tradução automática utilizam material específico a respeito da(s) língua(s) – gramática, dicionário, corpus, etc. – com a(s) qual(is) estão envolvidos para garantir embasamento teórico ou mesmo para teste.

Neste contexto, este capítulo apresenta os recursos lingüísticos construídos para o projeto PESA. O processo de construção ou preparação de tais recursos é descrito com mais detalhes em (Caseli & Nunes, 2002).

Os recursos lingüísticos do projeto PESA podem ser divididos em dois grupos: os corpora e a lista de palavras âncoras. Os corpora são conjuntos de textos paralelos utilizados por todos os métodos de alinhamento sentencial nas fases de teste – corpora de teste – e de avaliação – corpora de referência. A lista de palavras âncoras é uma lista bilíngüe na qual cada palavra na língua fonte possui uma ou mais equivalências (traduções) na língua alvo.

Como no projeto PESA estão envolvidas apenas duas línguas – o PB e o inglês –, podemos dizer que um corpus paralelo é um conjunto de pares de textos (ou bitextos) nos quais um é a tradução do outro. A terminologia utilizada na área atribui ao texto original o nome de texto fonte e a sua tradução, texto alvo.

O processo de construção de um corpus de textos envolve a seleção representativa de dados com o intuito de gerar um corpo de evidências lingüísticas que possa suportar generalizações e ser utilizado para testar hipóteses. Assim, é muito importante a definição de alguns critérios como o domínio do qual o corpus é uma amostra. Para esse trabalho, decidiu-se pelos seguintes critérios de seleção: linguagem escrita, formal, científica, atual, proveniente de trabalhos acadêmicos da área de computação.

A restrição quanto ao domínio do conhecimento de onde proviria esse corpus foi resultado da opção pela seleção de textos de um domínio específico. De acordo com a literatura consultada, esse é um critério a ser adotado quando a construção do corpus representa um estágio do desenvolvimento de um produto ou da busca de um objetivo de pesquisa, o caso desse trabalho (Renouf, 1987; Sinclair, 1991 apud Martins et al., 2001).

Outro fator importante na delimitação do corpus foi a disponibilidade de material já digitalizado: 65 pares de resumos e *abstracts* (textos paralelos) de trabalhos acadêmicos na área de computação desenvolvidos no Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo (ICMC-USP) em São Carlos, apresentados na forma de artigos publicados em revistas especializadas, monografias de qualificação de mestrado, dissertações de mestrado e teses de doutorado. Esses trabalhos pertencem a subdomínios variados da Computação, como: banco de dados, computação de alto desempenho, computação gráfica e processamento de imagens, engenharia de software, hipermídia, inteligência computacional, matemática computacional, sistemas digitais, sistemas distribuídos e programação concorrente.

Esse conjunto inicial, na verdade, foi dividido em dois: o corpus autêntico e o corpus pré-editado. O primeiro é formado pelos 65 bitextos na forma em que foram originalmente redigidos, sem nenhuma alteração em sua forma ou em seu conteúdo. O pré-editado, por sua vez, também é formado pelos mesmos 65 pares de textos, porém com correções, alterações e marcações feitas por um tradutor humano para a eliminação de ambigüidades, equívocos e erros de gramática e/ou tradução para o inglês. Detalhes sobre o processo de coleta e pré-edição dos textos dos corpora autêntico e pré-editado podem ser obtidos em (Martins et al., 2001).

Assim, o corpus autêntico possui um total de 855 sentenças e 21.432 palavras; e o pré-editado, 849 sentenças e 21.492 palavras. A Tabela 1 detalha esses números para cada um dos idiomas envolvidos (PB e inglês).

Tabela 1: Quantidade de palavras e sentenças nos corpora autêntico e pré-editado (Martins et al, 2001).

	Corpus autêntico	**Corpus pré-editado**
Palavras em PB	11.349	11.306
Palavras em inglês	10.083	10.186
Sentenças em PB	416	418
Sentenças em inglês	439	431

A divisão do corpus em autêntico e pré-editado foi feita com o intuito de analisar o impacto da qualidade do corpus no desempenho dos métodos de alinhamento. O interesse nessa análise vem do fato de que, segundo a literatura consultada, quando um corpus está relativamente isento de ruídos (sem erros gramaticais ou de tradução) e provém de domínios técnicos, nos quais as traduções literais são esperadas, o alinhamento automático de sentenças torna-se mais eficaz (Gaussier et al., 2000).

A partir dos corpora autêntico e pré-editado foram construídos os de teste e de referência. Os corpora de teste são aqueles fornecidos aos métodos de alinhamento sentencial para que sejam alinhados e posteriormente comparados aos textos paralelos alinhados e considerados ideais – os dos corpora de referência. Os corpora de referência são compostos por textos alinhados semi-automaticamente por tradutores humanos e o alinhamento produzido, por ser feito por um especialista humano, é considerado ideal. O conteúdo dos corpora de teste e de referência é o mesmo; a única diferença é que os de referência possuem marcações indicando o alinhamento das sentenças.

Assim, os corpora de teste foram divididos em corpus autêntico de teste (CAT) e corpus pré-editado de teste (CPT); e os corpora de referência em corpus autêntico de referência (CAR) e corpus pré-editado de referência (CPR). Além desses, outros dois corpora foram construídos especialmente para o método lingüístico que requer corpora etiquetados morfologicamente como entrada. São eles: o corpus autêntico de teste etiquetado morfologicamente (CATE) e o corpus pré-editado de teste etiquetado morfologicamente (CPTE).

Por fim, o último tipo de recurso lingüístico construído foi a lista de palavras âncoras (ou LPA). Nessa lista, cada entrada é constituída de uma palavra na língua fonte e uma ou mais palavras na língua alvo de tal forma que as palavras na língua alvo são as traduções da palavra na língua fonte.

O Quadro 3 traz todos os recursos lingüísticos e as siglas atribuídas a eles.

Quadro 3: Siglas definidas para os recursos lingüísticos do PESA.

Recurso lingüístico	**Sigla**
Corpus autêntico de teste	CAT
Corpus pré-editado de teste	CPT
Corpus autêntico de teste etiquetado morfologicamente	CATE
Corpus pré-editado de teste etiquetado morfologicamente	CPTE
Corpus autêntico de referência	CAR
Corpus pré-editado de referência	CPR
Lista de palavras âncoras	LPA

Todos os métodos implementados no projeto PESA utilizam pelo menos dois dos recursos lingüísticos do Quadro 3, que são os corpora de teste (CAT, CPT, CATE e CPTE) e os de referência (CAR e CPR). A LPA é utilizada apenas no processo de alinhamento dos métodos híbridos como um dos critérios para se determinar as correspondências entre as sentenças dos textos fonte e alvo. O Quadro 4 mostra a distribuição desses recursos

lingüísticos em relação aos métodos e às fases nas quais eles são usados (alinhamento, teste ou avaliação).

Quadro 4: Recursos lingüísticos utilizados pelos métodos de alinhamento sentencial implementados no projeto PESA.

Método	Classificação	Alinhamento	Teste	Avaliação
GC	Empírico	-	CAT e CPT	CAR e CPR
GMA	Empírico	-	CAT e CPT	CAR e CPR
Lingüístico	Lingüístico	-	CATE e CPTE	CAR e CPR
GSA+	Híbrido	LPA	CAT e CPT	CAR e CPR
TCA	Híbrido	LPA	CAT e CPT	CAR e CPR

Assim, quando os textos do CAT (ou CATE) forem utilizados para testar algum método de alinhamento, os textos alinhados pelo método serão comparados com os textos do CAR. De maneira semelhante, os textos do CPT (ou CPTE) serão utilizados juntamente com os textos do CPR.

O processo de construção e preparação dos recursos lingüísticos para o projeto PESA produziu, além dos seis corpora e da lista de palavras âncoras, programas e ferramentas computacionais implementados para auxiliar o processo de geração de tais recursos. Entre essas ferramentas a mais importante é a TagAlign: uma ferramenta de pré-processamento de corpus que possui módulos de marcação automática de fronteiras, de alinhamento semi-automático, de avaliação dos corpora alinhados pelos métodos de alinhamento e, futuramente, de alinhamento sentencial automático (Caseli et al., 2002)[14]. As funcionalidades da TagAlign vão além do PESA e atingem projetos futuros que poderão utilizá-las na geração de novos recursos lingüísticos.

Outro fato importante é que todos os recursos lingüísticos gerados para o PESA podem ser usados em outros projetos do NILC ou de outras instituições parceiras. Um exemplo desse fato é a utilização dos corpora de referência (CAR e CPR) como entrada para os métodos de alinhamento lexical de textos paralelos estudado no projeto PEWA (*Portuguese English Word Alignment*).

O processo de construção desses recursos é explicado de forma sucinta nas próximas subseções e em detalhes em (Caseli & Nunes, 2002). A Seção 3.1 apresenta os corpora de teste CAT e CPT; a Seção 3.2, os corpora etiquetados morfologicamente CATE e CPTE; a Seção 3.3, os corpora de referência CAR e CPR e a Seção 3.4, a LPA. A última Seção (3.5)

[14] Disponível em http://www.nilc.icmc.usp.br/nilc/publications.htm (17/02/2003).

traz uma descrição do formato dos arquivos de entrada e saída dos métodos de alinhamento sentencial do projeto PESA.

3.1 Corpora *de Teste*

Os corpora de teste servem de entrada para os métodos de alinhamento sentencial de textos paralelos e por serem utilizados com o intuito de analisar o desempenho desses métodos, ou seja, testá-los, receberam a denominação "corpora de teste". Eles são compostos por 65 pares de textos paralelos que podem ser os originalmente redigidos (corpus autêntico) ou os pré-processados por um tradutor humano (corpus pré-editado).

Um ponto importante do processo de alinhamento está relacionado ao modo como o corpus de teste é fornecido ao sistema: de forma codificada ou não. A codificação implica a identificação do formato do texto e/ou de suas partes componentes; por exemplo, uma indicação de início de parágrafo, início de sentença etc. Alguns autores afirmam que esse processo enriquece o corpus, justamente porque revela detalhes sobre a estrutura do texto original.

Os corpora utilizados nesse trabalho foram codificados utilizando a linguagem XML (*Extensible Markup Language*)[15]. A XML é uma extensão da SGML (*Standard Generalized Markup Language*)[16] projetada com o intuito de permitir que textos codificados de acordo com as especificações XML possam ser servidos, recebidos e processados na *web* da mesma forma que os documentos HTML.

A XML foi desenvolvida pelo XML *Working Group* (originalmente conhecido como SGML *Editorial Review Board*) formado sob o comando do *World Wide Web Consortium* (W3C)[17] em 1996.

A XML descreve uma classe de objetos denominada classe de documentos XML e especifica, parcialmente, o comportamento de programas que processam esses documentos. Por definição, os documentos XML estão em conformidade com os documentos SGML.

A XML foi preferida em relação à SGML por ser uma extensão desta última e, assim, possuir as mesmas vantagens dela e algumas outras decorrentes dos aperfeiçoamentos efetuados. Por exemplo, a facilidade de troca de documentos XML pode ser comparada à facilidade encontrada na troca de documentos HTML.

---

[15]Em http://www.w3.org/XML/ (17/02/2003).
[16]Em http://www.w3.org/MarkUp/SGML/ (17/02/2003).
[17]Em http://www.w3.org (17/02/2003).

Dessa forma, a construção dos corpora de teste resumiu-se à codificação dos textos autênticos e pré-editados com etiquetas XML. Essa codificação limitou-se à marcação de fronteiras de texto, parágrafos e sentenças, inseridas automaticamente pela ferramenta de pré-processamento de textos desenvolvida no NILC, a TagAlign. As etiquetas inicial e final de parágrafos (e) e sentenças (<s> e </s>) foram inseridas de acordo com um algoritmo simples que leva em consideração os casos mais comuns para determinar as fronteiras de parágrafos e sentenças, como a existência de ponto final (.) e de letras maiúsculas.

A etiqueta é inserida no início do texto ou imediatamente antes de uma letra precedida por um caractere de mudança de linha (Enter). A etiqueta , por sua vez, é inserida antes um caractere de mudança de linha (Enter) ou antes do fim do arquivo. As fronteiras de sentença são determinadas de maneira similar. Uma etiqueta <s> é inserida quando uma letra (não necessariamente maiúscula) é precedida por espaços e um caractere finalizador de sentença (.?!); ou pelo início de um parágrafo (um caractere de mudança de linha) ou do texto. Já a etiqueta </s> é inserida quando um caractere finalizador de sentença for seguido de espaços e uma letra (não necessariamente maiúscula) ou antes de um caractere de mudança de linha (fim de um parágrafo) ou fim de arquivo.

Possíveis erros gerados em casos que não se enquadram nos descritos anteriormente, como abreviações do tipo “S. O. S.”, devem ser tratados em uma posterior verificação manual dos arquivos etiquetados.

Além das etiquetas de início e fim de parágrafos e sentenças, outras duas foram inseridas para a identificação do texto: a <text>, etiqueta inicial inserida na primeira linha do texto, e a </text>, inserida na última linha do texto. A etiqueta inicial possui dois atributos que indicam a língua na qual o texto foi escrito e o identificam. O primeiro atributo, lang, contém a abreviação da língua escolhida pelo usuário da TagAlign como aquela na qual o texto foi escrito. Na versão atual da TagAlign, o atributo lang pode receber os valores: en (inglês) ou pt (português). O segundo atributo, id, contém o nome do arquivo no qual o texto está armazenado, sem a extensão (.txt) nem o caminho.

Os corpora gerados nesse processo receberam as siglas de CAT (corpus autêntico de teste) e CPT (corpus pré-editado de teste). Um exemplo de um bitexto autêntico antes e depois do processo de marcação é mostrado na Figura 2 e na Figura 3, respectivamente.

Este trabalho apresenta os requisitos funcionais identificados no processo de Engenharia Reversa de Software que possam ser suportados por um Sistema Hipertexto. Por meio da modelagem conceitual e navegacional do domínio de informações relativas ao método de engenharia reversa Fusion-RE/I, foram estabelecidos os requisitos funcionais de um aplicativo hipermídia de suporte ao método, de forma a nortear o engenheiro de software responsável pelo processo de engenharia reversa e possibilitar o acompanhamento da evolução desse processo.	This paper presents the functional requirements of the reverse engineering process in order to be supported by hypertext systems. These requirements were defined by a conceptual and navigation modelling of the information domain related to a reverse engineering method called Fusion-RE/I. Thus, the software engineer responsable for the reverse engineering process has the specific guidelines to be follow and these guidelines can be used during the process evolution.

Figura 2 – Bitexto autêntico antes da inserção de marcações de fronteiras.

<text lang=pt id=art3R> <s>Este trabalho apresenta os requisitos funcionais identificados no processo de Engenharia Reversa de Software que possam ser suportados por um Sistema Hipertexto.</s><s>Por meio da modelagem conceitual e navegacional do domínio de informações relativas ao método de engenharia reversa Fusion-RE/I, foram estabelecidos os requisitos funcionais de um aplicativo hipermídia de suporte ao método, de forma a nortear o engenheiro de software responsável pelo processo de engenharia reversa e possibilitar o acompanhamento da evolução desse processo.</s> </text>	<text lang=en id=art3A> <s>This paper presents the functional requirements of the reverse engineering process in order to be supported by hypertext systems.</s><s>These requirements were defined by a conceptual and navigation modelling of the information domain related to a reverse engineering method called Fusion-RE/I.</s><s>Thus, the software engineer responsable for the reverse engineering process has the specific guidelines to be follow and these guidelines can be used during the process evolution.</s> </text>

Figura 3 – Bitexto autêntico depois da inserção de marcações de fronteiras (pertencente ao CAT).

O intuito desse processo de codificação efetuado seguindo-se o padrão XML foi produzir corpora padronizados que poderão ser utilizados em projetos futuros e até mesmo intercambiados com outros pesquisadores para a utilização em outros tipos de projetos.

Os corpora resultantes do processo de marcação (corpora de teste CAT e CPT) serviram de base para a construção dos demais corpora do projeto PESA: os etiquetados morfologicamente e os de referência apresentados nas próximas subseções (3.2 e 3.3, respectivamente).

3.2 Corpora *Etiquetados Morfologicamente*

Além dos corpora de teste apresentados na Seção anterior, outros dois foram produzidos para a fase de teste: o corpus autêntico de teste etiquetado morfologicamente (CATE) e o corpus pré-editado de teste etiquetado morfologicamente (CPTE); ambos resultantes de um processo

de etiquetação morfológica feito com o TreeTagger (Schmid, 1995) para o português e para o inglês.

O etiquetador TreeTagger foi o escolhido por ter sido apontado em Aires e Aluísio (2001) como o de melhor desempenho para o português brasileiro (95,93% de precisão[18]) em uma avaliação que envolveu outros etiquetadores – TBL (Brill, 1995), MXPOST (Ratnaparcki, 1996 apud Aires & Aluísio, 2001) e PoSiTagger (Aires, 2000). O corpus de treinamento e teste utilizado nesta avaliação era composto de textos didáticos, jornalísticos e literários. Esse corpus misto (com 100000 palavras) foi construído com o propósito de abranger textos simples, ou seja, que seguem uma estrutura formal fixa (didáticos), textos mais próximos da linguagem viva (jornalísticos) e textos com estrutura livre (literários). Embora o TreeTagger tenha sido apontado como o melhor etiquetador para o PB, deve-se ter em mente que, de acordo com a literatura, 95,93% não pode ser considerada uma boa precisão.

No caso do inglês, verificou-se que a diferença de precisão entre o TreeTagger e o TBL (Brill, 1995) – 96,36% e 97%, respectivamente – no contexto desse projeto, não era suficientemente grande para justificar a utilização de etiquetadores diferentes para o inglês e o PB, assim sendo, optou-se pela utilização do TreeTagger também na etiquetação dos *abstracts*. O TreeTagger tanto para o inglês quanto para o PB está disponível no NILC e, no caso do português, seu treinamento foi produto de um mestrado desenvolvido neste mesmo laboratório.

Além do etiquetador, outros programas de pré e pós-processamento dos textos precisaram ser implementados para formatar a entrada e a saída do TreeTagger. Para o português, teve-se que separar cada palavra ou caractere de pontuação (:.?!, etc) em uma linha diferente. No caso do inglês, foi necessário apenas converter o caractere de mudança de linha (Enter) para o padrão Unix (plataforma na qual o TreeTagger foi implementado). O programa de pós-processamento, por sua vez, efetuou o processo inverso "voltando" os textos etiquetados para o formato original. Uma verificação manual dos textos foi feita para corrigir possíveis erros gerados por casos não previstos.

Na Figura 4 tem-se um exemplo de um bitexto do CATE. Trata-se do mesmo bitexto apresentado na Figura 3 após o processo de etiquetação morfológica.

<text lang=pt id=art3R> <s>Este PRON trabalho N apresenta VERB os ART requisitos N funcionais ADJ identificados VERB no PREP+ART processo N de PREP Engenharia N Reversa ADJ de PREP Software N que PRON possam VERB ser VERB suportados VERB por PREP um ART Sistema NP Hipertexto NP.</s><s>Por PREP meio N da PREP+ART modelagem N conceitual ADJ e CONJ navegacional ADJ do PREP+ART domínio N de PREP informações N relativas ADJ ao PREP+ART método N de PREP engenharia N reversa ADJ Fusion-RE N /I NUME, foram VERB estabelecidos VERB os ART requisitos N funcionais ADJ de PREP um ART aplicativo N hipermídia ADJ de PREP suporte N ao PREP+ART método N, de PREP forma N a PREP nortear VERB o ART engenheiro N de PREP software N responsável ADJ pelo PREP+ART processo N de PREP engenharia N reversa ADJ e CONJ possibilitar VERB o ART acompanhamento N da PREP+ART evolução N desse PREP+PD processo N.</s> </text>	<text lang=en id=art3A> <s>This DT paper NN presents VBZ the DT functional JJ requirements NNS of IN the DT reverse JJ engineering NN process NN in IN order NN to TO be VB supported VBN by IN hypertext JJ systems NNS.</s><s>These DT requirements NNS were VBD defined VBN by IN a DT conceptual JJ and CC navigation NN modelling NN of IN the DT information NN domain NN related VBN to TO a DT reverse JJ engineering NN method NN called VBD Fusion-RE RB /I. FW </s><s>Thus RB, the DT software NN engineer NN responsable JJ for IN the DT reverse JJ engineering NN process NN has VBZ the DT specific JJ guidelines NNS to TO be VB follow VB and CC these DT guidelines NNS can MD be VB used VBN during IN the DT process NN evolution NN.</s> </text>

Figura 4 – Exemplo de um bitexto do CATE.

A partir da Figura 4 nota-se que os conjuntos de etiquetas (*tagsets*) utilizados no processo de etiquetação dos textos em PB e em inglês são diferentes. Este fato foi considerado na implementação do método lingüístico e será novamente abordado na Seção que trata sobre o mesmo (Seção 5.1).

3.3 Corpora *de Referência*

Após o processamento dos corpora de teste (CAT, CATE, CPT e CPTE) fornecidos como entrada para os métodos de alinhamento sentencial, esses mesmos textos serão retornados com as indicações de correspondência entre as sentenças dos textos fonte e alvo, ou seja, os textos de entrada alinhados. Esses corpora alinhados serão comparados com os corpora de referência para avaliar o método de alinhamento que os gerou.

Os corpora de referência foram construídos a partir dos corpora de teste (CAT e CPT) e são resultantes de um processo semi-automático de marcação de correspondências entre as sentenças dos textos fonte e alvo. Essas marcações foram inseridas com o auxílio da TagAlign

[18] A precisão, nesse caso, foi calculada como o número de palavras classificadas corretamente dividido pelo número de palavras do arquivo de teste (Aires, 2000).

e os corpora gerados receberam as siglas CAR – corpus autêntico de referência – e CPR – corpus pré-editado de referência. Por serem considerados ideais, esses corpora foram utilizados como parâmetro na avaliação dos métodos de alinhamento sentencial.

O processo de alinhamento semi-automático do qual resultaram os corpora de referência foi feito com o auxílio do módulo de alinhamento sentencial manual da ferramenta TagAlign. Para o sucesso desse processo é necessário que o usuário conheça as línguas nas quais os textos paralelos foram escritos e, assim, possa alinhar um bitexto sem muito esforço. As indicações de correspondência são inseridas como atributos da etiqueta <s>, colocando-se em id um identificador único para a sentença e em corresp o(s) identificador(es) da(s) sentença(s) correspondente(s) a ela. O formato desse identificador é mostrado a seguir:

<nome_arquivo>.<número_parágrafo>**.s**<número_sentença>

Um exemplo de textos paralelos sentencialmente alinhados é apresentado na Figura 5. Trata-se do bitexto da Figura 3 após o processo de alinhamento sentencial manual.

<text lang=pt id=art3R> <s id=art3R.1.s1 corresp=art3A.1.s1>Este trabalho apresenta os requisitos funcionais identificados no processo de Engenharia Reversa de Software que possam ser suportados por um Sistema Hipertexto.</s><s id=art3R.1.s2 corresp='art3A.1.s2 art3A.1.s3'>Por meio da modelagem conceitual e navegacional do domínio de informações relativas ao método de engenharia reversa Fusion-RE/I, foram estabelecidos os requisitos funcionais de um aplicativo hipermídia de suporte ao método, de forma a nortear o engenheiro de software responsável pelo processo de engenharia reversa e possibilitar o acompanhamento da evolução desse processo.</s> </text>	<text lang=en id=art3A> <s id=art3A.1.s1 corresp=art3R.1.s1>This paper presents the functional requirements of the reverse engineering process in order to be supported by hypertext systems.</s><s id=art3A.1.s2 corresp=art3R.1.s2>These requirements were defined by a conceptual and navigation modelling of the information domain related to a reverse engineering method called Fusion-RE/I.</s><s id=art3A.1.s3 corresp=art3R.1.s2>Thus, the software engineer responsable for the reverse engineering process has the specific guidelines to be follow and these guidelines can be used during the process evolution.</s> </text>

Figura 5 – Exemplo de um bitexto sentencial e manualmente alinhado.

A única diferença entre os corpora de teste e os corpora de referência, como pode ser observado ao se comparar a Figura 3 e a Figura 5, são as indicações de correspondência entre as sentenças, ou seja, o alinhamento sentencial.

3.4 Lista de palavras âncoras

O último recurso lingüístico construído para o projeto PESA refere-se à lista de palavras âncoras (ou LPA). A lista de palavras âncoras é uma lista de palavras na qual cada entrada possui uma palavra na língua fonte (PB) e uma ou mais traduções para a língua alvo (inglês). Esse recurso lingüístico é utilizado pelos métodos híbridos como um dos critérios para alinhar as sentenças.

A construção da LPA baseou-se na análise das palavras nos textos de três corpora da área de computação levando-se em consideração dois fatores: as palavras deveriam ser razoavelmente freqüentes e as equivalências deveriam ser diretas nas duas línguas. Os corpora utilizados na construção da LPA foram:

- O corpus*DT* formado por 52 textos científicos (dissertações e teses) da área de Ciências da Computação escritos em PB. Esse corpus é um dos resultados do projeto SciPo[19], desenvolvido no NILC (Feltrim et al., 2001).

- O corpus cmp-lg (*Computation and Language*) formado por 183 artigos científicos escritos em inglês apresentados nas conferências da *Association for Computational Linguistics* (ACL) e preparado pela corporação MITRE[20].

- O corpus HCI composto por 102 introduções de trabalhos específicos da área de HCI, escritos em inglês. Esse corpus também está disponível no NILC (Silva, 1999).

Para cada um dos corpora acima citados foi gerada uma lista com as palavras mais freqüentes utilizando a ferramenta computacional de processamento de corpus WordSmith[21]. Essas listas foram analisadas manualmente e uma lista final com cerca de 250 entradas foi gerada para o par de línguas PB-inglês no domínio da computação.

A LPA contém, além das palavras, algumas expressões multipalavras e generalizações representadas pelo caractere *. Palavras com * no final indicam truncamento, ou seja, palavras começadas com a mesma seqüência de caracteres têm a mesma tradução, como é o caso de ambient* e environment*. O Quadro 5 traz um trecho da LPA utilizada no projeto PESA. É importante ressaltar que nada foi feito para eliminar da LPA palavras muito

[19] Em http://www.nilc.icmc.usp.br/nilc/projects/scipo.htm (17/02/2003).
[20] Em http://www.itl.nist.gov/iaui/894.02/related_projects/tipster_summac/cmp_lg.html (17/02/2003).
[21] Em http://www.lexically.net/wordsmith/ (17/02/2003).

freqüentes (artigos, preposições, etc.) e que, por isso, podem gerar ambigüidades no alinhamento. Este tratamento, quando necessário, é efetuado pelos métodos que a utilizam.

Quadro 5: Trecho da LPA utilizada no projeto PESA.

PB	Inglês
a	the, at, to
abordagem	approach
al	al
além	beyond
algoritmo	algorithm
algumas	some, several
alguns	some, several
ambient*	environment*
ambos	both
análise	analysis
ao	to the, for the, at the

Outro fato importante a ser mencionado é que a lista de palavras âncoras não contém todas as possíveis traduções, mas apenas as mais freqüentes ou as que foram encontradas nos corpora analisados. Dessa forma, essa lista não deve ser considerada um léxico bilíngüe para o par de línguas PB-inglês, mas apenas uma parte dele.

O conteúdo da LPA é utilizado pelos métodos híbridos como um dos critérios de alinhamento. Todas as palavras presentes nas sentenças fonte e alvo são procuradas na LPA e os pares (palavra_fonte, palavra_alvo) encontrados se tornam pontos de correspondência candidatos e são utilizados para determinar as correspondências entre as sentenças sob análise. Dessa forma, a LPA é uma legítima representante da influência lingüística nos métodos híbridos, enquanto critérios estatísticos do tipo freqüência de ocorrência de palavras e outros, são representantes da influência empírica nesses métodos.

3.5 Entrada e Saída dos Métodos

O formato dos arquivos de entrada e saída foram padronizados para os cinco métodos do projeto PESA. Como entrada, os métodos de alinhamento sentencial geralmente recebem apenas um par de textos paralelos, porém, este padrão foi alterado no projeto PESA para facilitar o processo de alinhamento e permitir que todos os textos de um corpus de teste sejam alinhados com uma só chamada ao método. Assim, a entrada dos métodos passou a ser um arquivo com os caminhos para todos os pares de textos paralelos do CAT (ou CPT). Cada

linha desse arquivo contém os caminhos para o texto fonte e o texto alvo, separados por um caractere '|', como mostrado a seguir:

```
C:\Temp\Corpus\Resumos\art1R.txt|C:\Temp\Corpus\Abstracts\art1A.txt
C:\Temp\Corpus\Resumos\art2R.txt|C:\Temp\Corpus\Abstracts\art2A.txt
…
```

Esse arquivo pode ser gerado utilizando-se o módulo de geração de corpus paralelo da ferramenta de pré-processamento de textos TagAlign e será referenciado no restante deste texto como <corpus paralelo>.

Além do <corpus paralelo>, a entrada dos métodos pode possuir três parâmetros: as etiquetas de identificação do texto e de marcação de fronteiras de parágrafos e sentenças. Se essas três etiquetas não forem explicitamente informadas, o método considera as etiquetas padrão: text, p e s, respectivamente.

Cada um dos textos indicados no <corpus paralelo> deve estar marcado com essas etiquetas, o que também pode ser feito com o auxílio da TagAlign, como descrito na Seção 3.1. Esta Seção também traz um exemplo de um bitexto no formato fornecido como entrada para os métodos (Figura 3).

Com relação à saída, os métodos podem produzir diversos arquivos, de acordo com suas características; porém, pelo menos três estão presentes em todos: um arquivo com as indicações dos alinhamentos e outros dois arquivos com os textos de entrada alinhados. Um exemplo de um arquivo com indicações dos alinhamentos é apresentado na Figura 6.

```
1 <=> 1
2 <=> 2
3 <=> 3
4 <=> 4,5
```

Figura 6 – Exemplo de um arquivo de saída com indicações dos alinhamentos.

Nesta figura, os números das sentenças fonte aparecem à esquerda e os das sentenças alvo, à direita, separados pelos caracteres " <=> ". Os números representam a posição das sentenças no texto todo sem considerar as fronteiras de parágrafos. Quando um alinhamento é composto por várias sentenças, estas são separadas por vírgulas; e quando nenhuma sentença estiver presente em um dos lados (um caso de omissão), a ausência de sentenças é representada pela palavra *omitted*.

Os outros dois arquivos de saída contêm os textos de entrada alinhados. Esses textos são salvos em arquivos com o mesmo nome dos arquivos originais (fonte e alvo), porém com a extensão **.al**. Nesses arquivos, as correspondências entre as sentenças são indicadas pelos

atributos id e corresp inseridos na etiqueta inicial das sentenças, como descrito na Seção 3.3. Esta Seção também traz um exemplo de um bitexto no formato dos produzidos como saída pelos métodos (Figura 5).

Capítulo 4

Métodos Empíricos de Alinhamento Sentencial

Os métodos empíricos são aqueles que, em seu processo de alinhamento, não utilizam nenhum tipo de informação lingüística, apenas informações estatísticas, como a freqüência de ocorrência de palavras e/ou sua distribuição no texto, e técnicas de reconhecimento de padrão que consideram como possíveis traduções mútuas duas palavras de grafias similares (cognatos ou palavras homógrafas).

Os primeiros métodos de alinhamento sentencial desenvolvidos pelos dois grupos de estudos iniciais citados no Capítulo 2 são exemplos de métodos empíricos, pois utilizam apenas informações contidas nos textos, sem recursos lingüísticos extras, para efetuar o alinhamento. Um deles, o método de Gale e Church (1991, 1993), é apresentado com mais detalhes a seguir (Seção 4.1). Outro método empírico, baseado em reconhecimento de padrão, descrito em (Melamed, 2000), será detalhado em seguida. Ambos serão referenciados por siglas para facilitar a identificação, sendo o primeiro referenciado como GC (as iniciais de seus autores) e o segundo como GMA (*Geometric Mapping and Alignment*).

Os métodos empíricos apresentados neste capítulo foram selecionados para integrar o projeto PESA devido a diversos fatores, entre eles: apresentam uma precisão satisfatória para outras línguas; são representantes de métodos empíricos que utilizam critérios de alinhamento distintos; e são muito referenciados na literatura da área, inclusive como base para comparação de desempenho com outros métodos.

As próximas seções apresentam os métodos empíricos GC e GMA, suas características, o processo de alinhamento, os detalhes da implementação e os passos necessários para a adequação de cada método aos requisitos do projeto PESA. Na Seção 4.1 é apresentado o método GC e na Seção 4.2, o método GMA. Os resultados da avaliação destes métodos são relatados na Seção 7.1.

4.1 Método GC

O método GC (Gale & Church, 1991, 1993) foi um dos primeiros métodos de alinhamento sentencial propostos à comunidade científica, juntamente com o método de Kay e Röscheisen (1988, 1993). Esses dois representantes da classe de métodos empíricos até hoje servem de base para diversos novos métodos nessa área.

O método GC baseia-se em um modelo estatístico simples que leva em consideração apenas o tamanho das sentenças, em caracteres, para determinar as correspondências entre elas. Esse método parte do pressuposto de que o tamanho de uma sentença no texto fonte está fortemente relacionado ao tamanho de sua tradução no texto alvo: sentenças curtas tendem a ter traduções curtas, e sentenças longas, traduções longas. Além disso, verifica-se uma taxa relativamente fixa entre os tamanhos das sentenças em quaisquer duas línguas, medida em número de caracteres ou de palavras. No caso do par de línguas estudado neste trabalho constatou-se que os textos dos corpora utilizados na avaliação possuem uma taxa média de 0,89 caractere em inglês para cada caractere em PB.

Em uma primeira avaliação realizada com um corpus composto por relatórios econômicos do *Union Bank of Switzerland* (UBS) em três idiomas, inglês, francês e alemão, o método GC alinhou corretamente todas as sentenças, com exceção de 4% delas. Além disso, desse corpus foi extraído um sub-corpus composto por 80% dos textos que obtiveram melhor precisão no alinhamento, e a taxa de erro para esse sub-corpus baixou de 4% para 0,7%. Outros detalhes dessa avaliação são fornecidos na Seção que descreve os resultados da avaliação do método GC no projeto PESA (Seção 7.1.1).

O método GC também foi utilizado como base para comparação de diversos métodos de alinhamento sentencial de textos paralelos na avaliação realizada em (Véronis & Langlais, 2000). Ele foi escolhido como *baseline* por ser um método de simples implementação e por apresentar um bom desempenho considerando sua simplicidade.

O código utilizado nesta avaliação está disponível juntamente com a versão do método GMA usada neste trabalho, a 1.2.1. O código foi escrito em Java e outros programas foram implementados em Perl para adaptá-lo aos requisitos do projeto PESA.

A Seção seguinte (4.1.1) traz uma explicação geral do processo de alinhamento do método GC e alguns detalhes de sua implementação e adequação aos requisitos do projeto PESA.

4.1.1 O Alinhamento

Resumidamente, o processo de alinhamento do método GC possui dois passos. Primeiro, os parágrafos dos bitextos são alinhados entre si e, então, as sentenças dentro dos parágrafos são alinhadas. O alinhamento de parágrafos pode ser automático, acompanhado de uma verificação manual.

O método GC, da forma como foi proposto, possui a restrição de só alinhar textos com o mesmo número de parágrafos, sendo que esta restrição foi removida na implementação aqui descrita. Assim, as sentenças são alinhadas independentemente do parágrafo no qual estão contidas.

No processo de alinhamento é utilizada a técnica de programação dinâmica, freqüentemente empregada para alinhar duas seqüências de símbolos, como no caso de códigos genéticos de duas espécies diferentes. No alinhamento sentencial, a programação dinâmica mede a similaridade de duas sentenças verificando-se a facilidade de se transformar uma na outra. Nesse processo, são aplicadas seis operações básicas: remoção, inserção, substituição, contração, expansão e união.

Dessa forma, dado um número de alinhamentos possíveis, o método GC busca o "melhor" alinhamento que englobe o maior número de sentenças em uma determinada vizinhança. O melhor alinhamento é determinado utilizando-se uma medida de distância para comparar dois elementos individuais dentro das seqüências, e um algoritmo de programação dinâmica para minimizar as distâncias totais entre os elementos alinhados dentro de duas seqüências. Em outras palavras, o processo de alinhamento tenta encontrar as sentenças mais "próximas", ou seja, as sentenças candidatas à tradução.

O primeiro passo do método consiste em considerar as sentenças dos textos paralelos como pertencentes a um único parágrafo e então alinhá-las.

Com citado anteriormente, o objetivo do método GC é encontrar o alinhamento com maior probabilidade dado um conjunto de possibilidades. Para isso, calcula-se uma medida de distância que verifica a probabilidade de uma sentença na língua fonte ser a tradução de um conjunto de sentenças (zero, uma ou mais) na língua alvo e vice-versa. Essa probabilidade é calculada baseada em dois parâmetros: a média e a variância do número de caracteres na língua alvo por caractere na língua fonte.

A média (c) pode ser estimada pela soma do número de caracteres no texto escrito na língua alvo dividida pelo número de caracteres no texto escrito na língua fonte:

$$c = \frac{NúmeroDeCaracteresNaLínguaAlvo}{NúmeroDeCaracteresNaLínguaFonte} \quad (1)$$

A variância (s^2) é estimada em função dos comprimentos dos textos sendo alinhados e é determinada pela inclinação da linha de regressão robusta considerando a relação entre os tamanhos dos parágrafos na língua fonte (valores do eixo x) e o quadrado da diferença de tamanho entre os parágrafos nas duas línguas (valores do eixo y).

Dessa forma, a medida de distância (*d*) baseia-se em um modelo probabilístico o que, segundo os autores, permite que a informação seja combinada de uma forma consistente:

$$d = -\log Prob(match|\delta) \quad (2)$$

em que δ depende dos tamanhos das duas porções dos textos sob consideração (equação (3)) e o log é utilizado apenas para garantir que as distâncias produzirão resultados desejados (entre 0 e 1).

Essa medida de distância parte do pressuposto de que cada caractere na língua fonte dá origem a um número randômico de caracteres na língua alvo. Assume-se que estas variáveis randômicas são independentes e identicamente distribuídas com uma distribuição normal. δ é então:

$$\delta = \frac{(l_2 - l_1 c)}{\sqrt{l_1 s^2}} \quad (3)$$

em que l_2 e l_1 são os tamanhos (em caracteres) das porções sob consideração nos textos alvo e fonte, respectivamente; c é o número de caracteres esperados na língua alvo por caractere na língua fonte (equação (1)) e s^2 é a variância do número de caracteres na língua alvo por caractere na língua fonte.

Todos os cálculos envolvidos na determinação da distância *d* são efetuados na sub-rotina *two_side_distance*. Nela são utilizadas três constantes referentes às penalidades para os alinhamentos diferentes de 1-1, ou seja, 0-1 ou 1-0, 1-2 ou 2-1 e 2-2[22]. Essas constantes foram calculadas com base na probabilidade de ocorrência de cada categoria de alinhamento nos dois corpora disponíveis para a avaliação: o corpus autêntico e o corpus pré-editado resultando nos valores apresentados na Tabela 2.

[22] O método GC não considera as categorias de alinhamento *n*-*m* com *n*, *m* >2.

Tabela 2: Probabilidades dos alinhamentos.

	Corpus Autêntico		Corpus Pré-editado	
Categoria	**Freqüência**	**Probabilidade**	**Freqüência**	**Probabilidade**
1-1	353	0,874	395	0,949
1-0 ou 0-1	6	0,015	2	0,005
2-1 ou 1-2	41	0,101	17	0,041
2-2	4	0,010	2	0,005

Uma outra categoria não incluída na Tabela 2 e presente no corpus autêntico é a 2-3. Esta foi excluída da análise de probabilidade por não ser tratada pelo método GC que, por razões computacionais, possui a limitação de alinhar apenas pares *m-n* com $0 \leq m, n \leq 2$, possibilitando, assim, que o alinhamento ótimo seja eficientemente computado aplicando-se o algoritmo de programação dinâmica convencional. O único alinhamento da categoria 2-3 encontrado foi desconsiderado nessa análise.

A sub-rotina *two_side_distance* recebe quatro argumentos – x_1, y_1, x_2, y_2 – correspondentes às sentenças nos textos fonte e alvo, e calcula a distância de acordo com esses valores sendo:

1. $d(x_1, y_1; 0, 0)$ o custo da substituição de x_1 por y_1,
2. $d(x_1, 0; 0, 0)$ o custo da remoção de x_1,
3. $d(0, y_1; 0, 0)$ o custo da inserção de y_1,
4. $d(x_1, y_1; x_2, 0)$ o custo da contração de x_1 e x_2 para y_1,
5. $d(x_1, y_1; 0, y_2)$ o custo da expansão de x_1 para y_1 e y_2, e
6. $d(x_1, y_1; x_2, y_2)$ o custo da união de x_1 e x_2 correspondendo a y_1 e y_2.

Dessa forma, a medida de distância para cada par de sentenças proposto é calculada em relação aos comprimentos das sentenças dos dois textos e da variância dessa relação. Esses valores são submetidos a um algoritmo de programação dinâmica. A programação dinâmica é uma técnica para otimização de problemas nos quais a solução final é construída a partir de sucessivas escolhas locais, mas sem que, necessariamente, uma escolha local ótima faça parte da solução final ótima (Campbell et al., 1998).

No alinhamento de textos paralelos, o algoritmo de programação dinâmica tenta encontrar o alinhamento com a menor distância dentro da maior região (ou vizinhança). No método GC são calculadas seis distâncias (descritas acima) para as sentenças fonte, s_1 ... s_i, e suas traduções, t_1 ... t_j, e considera-se a melhor solução, D(i,j), a distância mínima entre todas as combinações possíveis. Assim:

$$D(i,j) = \min \begin{cases} D(i, j-1) + d(0, t_j; 0, 0) \\ D(i-1, j) + d(s_i, 0; 0, 0) \\ D(i-1, j-1) + d(s_i, t_j; 0, 0) \\ D(i-1, j-2) + d(s_i, t_j; 0, t_{j-1}) \\ D(i-2, j-1) + d(s_i, t_j; s_{i-1}, 0) \\ D(i-2, j-2) + d(s_i, t_j; s_{i-1}, t_{j-1}) \end{cases}$$

O caso mais comum acontece quando uma sentença no texto fonte corresponde a exatamente uma sentença no texto alvo, 1-1. Mas casos de omissão, 1-0, adição, 0-1, ou combinações de complexidade variável, *m-n* com $0 \leq m, n \leq 2$, também são encontrados.

Os valores padrão para os parâmetros do método GC – c, s^2 e as penalidades para alinhamentos 0-1 ou 1-0, 1-2 ou 2-1 e 2-2 – foram mantidos, pois as alterações desses valores trouxeram pouca variação no resultado final da avaliação sendo favoráveis no alinhamento dos textos do CPT, mas desfavoráveis nos do CAT. Assim, os valores dos parâmetros do GC utilizados nesta implementação são mostrados na Tabela 3.

Tabela 3: Parâmetros do GC.

Parâmetro	Valor Padrão
c	1
s^2	6,8
Penalidade 0-1	450
Penalidade 1-2	230
Penalidade 2-2	440

Além dos arquivos originais do método GC (escritos em Java), novos programas foram implementados para adequá-lo aos requisitos do projeto PESA entre eles o GCalign que recebe como parâmetro o arquivo com o corpus a ser alinhado (<corpus paralelo>, vide Seção 3.5). O programa GCalign gerencia todos os outros e executa basicamente três funções: pré-processamento dos textos, alinhamento e pós-processamento dos textos.

O pré-processamento dos textos tem o objetivo de armazenar todas as sentenças dos textos fonte e alvo sem as etiquetas de início (<s>) e fim (</s>), para que o tamanho delas possa ser calculado. Após o pré-processamento, os textos estão prontos para serem alinhados pelo método GC. O programa GCalign faz então uma chamada ao programa de alinhamento original (também denominado GCalign) passando como parâmetro uma string com os tamanhos das sentenças fonte e alvo no seguinte formato:

"tsf_1, tsf_2, ..., tsf_n, \\<BRK\\>, tsa_1, tsa_2,..., tsa_m"

em que tsf_i ($1 \leq i \leq n$) representa o tamanho da sentença fonte *i* e tsa_j ($1 \leq j \leq m$), o tamanho da sentença alvo *j* em um bitexto com *n* sentenças fonte e *m* sentenças alvo. Esta chamada é efetuada para cada um dos bitextos presentes no corpus paralelo passado como parâmetro. Por exemplo, para o par de textos paralelos art1R.txt e art1A.txt a chamada ao GCalign seria:

```
java GCalign -d '<BRK>' -i "85,42,196,154,\\<BRK\\>,52,37,191,84,57"
```

Os textos paralelos art1R.txt e art1A.txt são alinhados e a saída é salva no arquivo art1.txt no formato apresentado na Seção 3.5 (Figura 6). Os arquivos de entrada com marcações de alinhamento, como mostrado na Seção 3.5, também são retornados como a saída do método GC. A Figura 7 traz um exemplo destes arquivos, os textos art1R.txt e art1A.txt, alinhados pelo GC.

<text lang=pt id=art1R> <s id=**art1R.1.s1** corresp=art1A.1.s1>Neste artigo é apresentada uma ferramenta para validação e verificação de requisitos.</s><s id=**art1R.1.s2** corresp=art1A.1.s2>Essa ferramenta suporta a abordagem ERACE.</s><s id=**art1R.1.s3** corresp=art1A.1.s3>Tal abordagem parte do documento de requisitos do sistema e propõem a especificação das interações entre o sistema e seus agentes (cenários), e então os cenários são especificados detalhadamente.</s><s id=**art1R.1.s4** corresp='art1A.1.s4 art1A.1.s5'>Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.</s> </text>	<text lang=en id=art1A> <s id=art1A.1.s1 corresp=**art1R.1.s1**>A tool to support requirements trading is presented.</s><s id=art1A.1.s2 corresp=**art1R.1.s2**>The tool supports the ERACE approach.</s><s id=art1A.1.s3 corresp=**art1R.1.s3**>This approach starts from the system's requirement document and proposes to specify interactions between the system and its agents (scenarios), and then the scenarios are specified in detail.</s><s id=art1A.1.s4 corresp=**art1R.1.s4**>Heuristics to evolve from the requirements model to the analysis are also presented.</s><s id=art1A.1.s5 corresp=**art1R.1.s4**>An example to illustrates the approach is also presented.</s> </text>

Figura 7 – Exemplo de um bitexto alinhado pelo GC.

Os resultados da avaliação do método GC são apresentados na Seção 7.1.

4.2 Método GMA

O *Geometric Mapping and Alignment* (GMA) é um método empírico de alinhamento sentencial de textos paralelos que utiliza dois algoritmos em seu processo de alinhamento: o *Smooth Injective Map Recognizer* (SIMR) e o *Geometric Segment Alignment* (GSA) (Melamed, 2000). Embora pertença à mesma classe que o método GC – a classe dos métodos

empíricos – o GMA possui critérios de alinhamento diferentes deste. Enquanto o primeiro baseia-se na idéia de correlação entre os tamanhos das sentenças a serem alinhadas, o segundo utiliza a técnica de reconhecimento de padrão para mapear os pontos de correspondência entre os dois textos (SIMR) e, a partir dos mapeamentos resultantes, alinhar as sentenças (GSA).

Em uma avaliação realizada em (Melamed, 2000) com o corpus Hansard composto por textos paralelos inglês-francês, o método GMA apresentou uma precisão de 97,7% a 98,5%, de acordo com o tipo de texto analisado e a existência de um alinhamento prévio no nível de parágrafos.

Em uma outra avaliação apresentada pelo projeto ARCADE, no qual o GMA foi analisado, a precisão obtida foi de 94,2% em textos técnicos sem correspondências cruzadas, ou seja, as correspondências entre sentenças obedecem a mesma ordem em que elas aparecem nos textos. Além disso, sua performance foi a melhor entre os sistemas que não dispunham de recursos lingüísticos extras como léxicos ou glossários.

O GMA é um método de código aberto e a versão usada neste trabalho é a 1.2.1. O código foi escrito em Perl, Java e C++ e, além dos originais, outros programas foram criados em Perl para adaptá-lo aos requisitos do projeto PESA.

A próxima Seção (4.2.1) apresenta o processo de alinhamento do método GMA e as alterações feitas no método original para adaptá-lo aos requisitos do projeto PESA. A Seção 4.2.2, por sua vez, relata o processo de otimização dos parâmetros de um dos algoritmos utilizados pelo GMA, o SIMR, para textos paralelos PB-inglês.

4.2.1 O Alinhamento

Como mencionado na Seção anterior, o GMA utiliza dois algoritmos para alinhar os textos paralelos: o SIMR e o GSA.

O SIMR é um algoritmo genérico de reconhecimento de padrão particularmente bem-sucedido para mapeamento de correspondências em bitextos. O objetivo desse algoritmo é identificar palavras no texto alvo similares a palavras no texto fonte (segundo critérios explicados mais tarde) e retornar pares de coordenadas (x,y) indicando que na posição y do texto alvo existe uma palavra que pode ser considerada a tradução de uma palavra na posição x do texto fonte.

O mapeamento resultante, também denominado “mapeamento do bitexto”, está longe de ser considerado um bom alinhamento de palavras, mas é muito eficiente quando usado como passo intermediário no processo de alinhamento de segmentos.

Geometricamente, o problema de mapeamento de bitexto utilizando reconhecimento de padrão pode ser compreendido pela ilustração na Figura 8. Nesta figura, os bitextos são dispostos em dois eixos perpendiculares formando um retângulo que recebe o nome de espaço do bitexto. Por convenção, a cada palavra presente nos textos é atribuída a posição de seu caractere mediano.

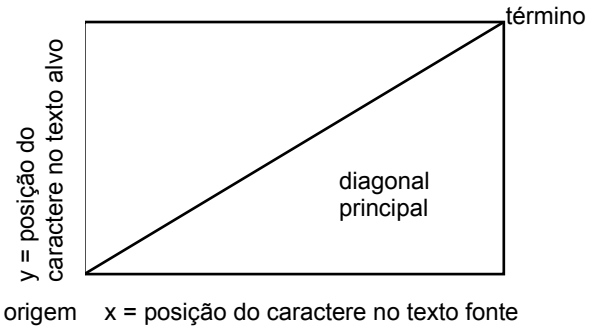

Figura 8 – Espaço do bitexto (Melamed, 2000).

A determinação dos pontos (*x*,*y*) que formam o mapeamento do bitexto engloba duas fases: a fase de geração (inclui a filtragem dos pontos) e a fase de reconhecimento. Na fase de geração, podem-se utilizar diversos recursos para determinar se duas palavras são a tradução uma da outra como uma lista de palavras âncoras (vide Seção 3.4) ou uma medida baseada em cognatos que, mesmo sendo mais simples, apresenta resultados satisfatórios no caso de línguas similares.

A versão empírica do método GMA, apresentada nesta Seção, utiliza uma medida baseada em cognatos denominada *Longest Common Subsequence Ratio* (ou LCSR) para gerar os pontos de correspondência candidatos. A LCSR de duas palavras é a razão do tamanho da maior subseqüência comum (não necessariamente contínua), denotada por LCS (*Longest Common Subsequence*) e o tamanho da maior palavra. Se o valor de LCSR para duas palavras, A e B, ultrapassar (ou for igual) a um determinado valor limite, então A e B são consideradas cognatas. Assim, o LCSR(A,B) é calculado como segue:

$$LCSR(A,B) = \frac{tamanho[LCS(A,B)]}{\max[tamanho(A), tamanho(B)]} \quad (4)$$

Por exemplo, a palavra *medicina* possui 7 caracteres que aparecem na mesma ordem na palavra *medicine*. Assim, o LCSR para estas palavras é 7/8 (ou 0,875). Já o LCSR de *mensagem* e *message* é apenas 6/8 (ou 0,75). O valor limite para o LCSR é um dos parâmetros do SIMR e foi otimizado junto com os demais como explicado na Seção 4.2.2.

Após a geração dos pontos de acordo com a LCSR, uma filtragem é realizada para a remoção daqueles que podem gerar ruídos. O filtro aplicado neste caso baseia-se no parâmetro de *nível de ambigüidade máximo do ponto*. Para cada ponto de correspondência *p=(x,y)* calcula-se o nível de ambigüidade (NA) da seguinte forma:

$$NA(p) = X + Y - 2 \quad (5)$$

em que X e Y são o número de pontos na coluna *x* e o número de pontos na linha *y*, respectivamente. O valor calculado é comparado com o parâmetro e os pontos que ultrapassarem este valor são ignorados automaticamente.

Depois da filtragem de ruído, tem-se a fase de reconhecimento na qual se testa as cadeias (seqüências lineares) de pontos em relação a três propriedades: injectividade[23], linearidade e inclinação constante.

A propriedade de injectividade garante que não existem dois pontos em uma cadeia com as mesmas coordenadas *x* ou *y*. Essa propriedade é testada na fase de filtragem de ruídos, na qual as cadeias com nível de ambigüidade muito alto (maior que o parâmetro de nível de ambigüidade máximo do ponto) são automaticamente rejeitadas. A propriedade de linearidade é entendida como a tendência dos pontos a se alinhar e é verificada calculando-se a raiz da distância média ao quadrado dos pontos da cadeia a partir da linha de mínimos quadrados dessa cadeia. Se a distância exceder o parâmetro de *dispersão máxima da cadeia ponto*, a cadeia é rejeitada.

Por fim, verifica-se a propriedade de inclinação constante: quando a inclinação de uma cadeia se aproxima da inclinação do bitexto, ou seja, de sua diagonal principal. Para isso compara-se o ângulo da linha de mínimos quadrados de cada cadeia à arctangente da inclinação do bitexto. Se a diferença exceder o parâmetro de *desvio máximo do ângulo* a cadeia é rejeitada.

Além dos parâmetros citados existe outro de fundamental importância em todo o processo de determinação das cadeias de pontos de correspondência: o tamanho da cadeia. O SIMR especifica um tamanho fixo (k) para a cadeia, com $6 \leq k \leq 11$, sendo que o valor exato de k depende da língua e deve ser otimizado junto com os outros parâmetros como mostrado na Seção 4.2.2. Todos esses parâmetros diminuem o espaço de busca por cadeias mantendo a complexidade do algoritmo dentro dos padrões aceitáveis.

[23] Tradução encontrada na área, mas não abonada, de injectivity.

Como resultado, o SIMR retorna um mapeamento dos pontos de correspondência dos textos fonte e alvo como mostrado na Figura 9.

O outro algoritmo utilizado pelo GMA é o GSA: um algoritmo que alinha segmentos de qualquer tamanho a partir dos mapeamentos retornados pelo SIMR e informações sobre as fronteiras dos segmentos (sentenças, no caso do alinhamento sentencial). Essas informações são os dados de entrada do GSA e podem ser geometricamente visualizadas na Figura 9, onde cada célula representa o produto de duas sentenças, uma de cada texto. Um ponto de correspondência na célula (*X*, *y*) indica que alguma palavra na sentença *X* corresponde a alguma palavra na sentença *y*, isto é, as sentenças *X* e *y* correspondem. Dessa forma, se as sentenças (X_1, ... X_n) alinham com as sentenças (y_1, ... y_m), então [(X_1, ... X_n), (y_1, ... y_m)] constitui um *bloco alinhado*. Os blocos alinhados na Figura 9 são demarcados com linhas sólidas.

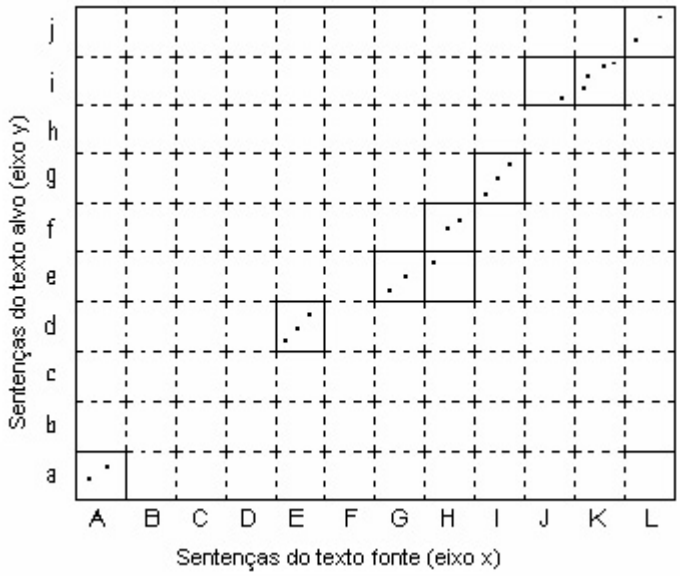

Figura 9 – Informações de entrada para o algoritmo GSA com indicações dos blocos alinhados (Melamed, 2000).

O primeiro passo do GSA é arranjar todas as células com pontos de correspondência em retângulos que não se sobreponham. Para isso as seguintes operações são executadas. Primeiro, se a entrada contém, por exemplo, os pares (*G*,*e*), (*H*,*e*), e (*H*,*f*) como ilustrado na Figura 9, então o GSA adiciona o par (*G*,*f*) inserindo um ponto de correspondência na célula (G, f) como mostra a Figura 10. Depois, o GSA força todos os segmentos a serem contínuos: se a sentença *Y* corresponde às sentenças *x* e *z*, mas não à *y*, sendo que a ordem delas no texto é *x*, *y* e *z*, então o par (*Y*,*y*) é adicionado.

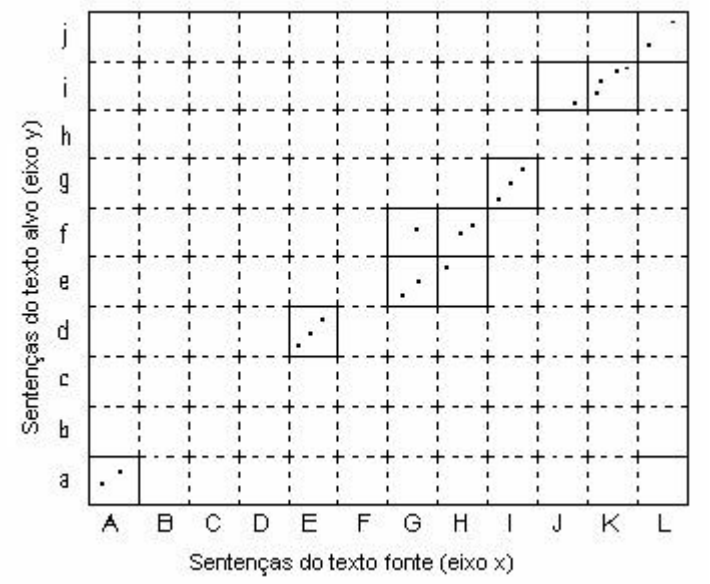

Figura 10 – Figura 9 após a inserção de um ponto de correspondência na célula (G, f).

Os passos seguintes tentam reduzir os erros produzidos pela ausência de pontos de correspondência em segmentos de um dos textos (por exemplo, as sentenças (B,C,D) e (b,c) na Figura 9), pela existência de alinhamentos 1-n, onde n > 1 (por exemplo, as células (H,e) e (H,f) na Figura 9), ou pela combinação desses casos. Para isso o GSA faz o realinhamento usando um método baseado em tamanho (o GC, por exemplo). Se esse realinhamento ultrapassar um nível de confiança pré-estabelecido, o GSA aceita o resultado produzido pelo realinhamento, caso contrário, o alinhamento indicado pelos pontos de correspondência do SIMR é mantido.

Além dos arquivos fonte originais da versão 1.2.1 do GMA, novos programas foram implementados para adequá-lo aos requisitos do projeto PESA, entre eles o GMAalign que, da mesma forma que o GCalign (vide Seção 4.1), recebe como parâmetro o arquivo com o corpus a ser alinhado (<corpus paralelo>, vide Seção 3.5). O programa GMAalign gerencia todos os outros e executa basicamente três funções: pré-processamento dos textos, alinhamento e pós-processamento dos textos.

O pré-processamento dos textos tem o objetivo de formatá-los de acordo com o formato esperado pelo GMA e, para isso, as etiquetas de identificação do texto (<text></text>) e de fronteiras de parágrafos ( ) são removidas. No caso das fronteiras de sentenças (<s></s>), as finais (</s>) são removidas e as iniciais (<s>) são substituídas por marcadores de segmento (<EOS>) durante a geração dos eixos pelo programa *axis*. Além disso, a cada *token* (palavra ou símbolo separados por espaços) é atribuída a posição de seu caractere mediano (vide Figura 8). Um trecho do eixo gerado para um texto em PB é mostrado na Figura 11.

<text lang=pt id=art1R> <s>Neste artigo é apresentada uma ferramenta para validação e verificação de requisitos.</s><s>Essa ferramenta suporta a abordagem ERACE.</s><s>Tal abordagem parte do documento de requisitos do sistema e propõem a especificação das interações entre o sistema e seus agentes (cenários), e então os cenários são especificados detalhadamente.</s><s>Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.</s> </text>	0 <EOS> 3 Neste 9.5 artigo 14 é 21 apresentada 29 uma 36.5 ferramenta 44.5 para 52 validação 58 e 65 verificação 72.5 de 79.5 requisitos 86 . 89 <EOS>

Figura 11 – Exemplo de um texto em PB e parte do eixo gerado para ele.

Após o pré-processamento, os textos estão prontos para serem alinhados pelo método GMA. Nesse momento o programa principal, GMAalign, faz uma chamada ao programa de alinhamento original (GMA.csh) para cada um dos bitextos presentes no corpus paralelo passado como parâmetro. Por exemplo, para o par de textos paralelos art1R.txt e art1A.txt a chamada ao GMA.csh seria:

```
GMA.csh config art1R.txt art1A.txt > art1.txt
```

na qual o arquivo de configuração config contém os valores dos parâmetros do GMA otimizados para o PB e o inglês (vide Seção 4.2.2). Os textos paralelos art1R.txt e art1A.txt são alinhados e a saída é salva no arquivo art1.txt no formato apresentado na Seção 3.5 (Figura 6). O mapeamento do bitexto (art1R-art1A) pode ser geometricamente visualizado na Figura 12 na qual também estão representadas as fronteiras das sentenças dos textos fonte e alvo, e os blocos alinhados, indicados por linhas sólidas.

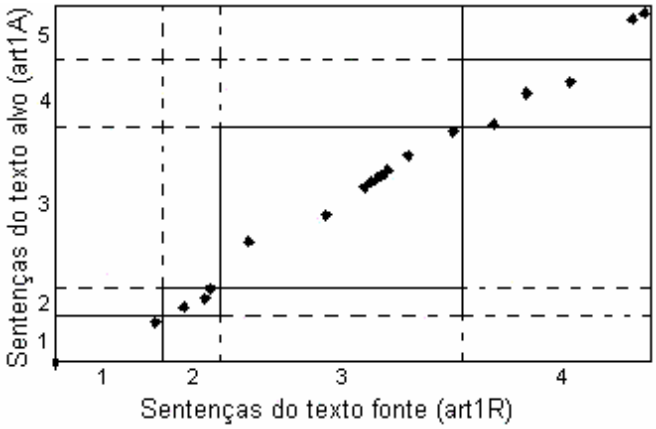

Figura 12 – Mapeamento do bitexto (art1R-art1A) com blocos alinhados.

Os arquivos de entrada com marcações de alinhamento, como mostrado na Seção 3.5, também são retornados pelo método GMA. A Figura 13 traz um exemplo desses arquivos, os textos art1R.txt e art1A.txt, alinhados pelo GMA. As palavras e caracteres de pontuação sublinhados representam os pontos de correspondência considerados pelo GMA no alinhamento das sentenças.

<text lang=pt id=art1R>
<s id=**art1R.1.s1**
corresp=art1A.1.s1>Neste artigo é apresentada uma ferramenta para validação e verificação de requisitos.</s><s id=**art1R.1.s2** corresp=art1A.1.s2>Essa ferramenta suporta a abordagem ERACE.</s><s id=**art1R.1.s3**
corresp=art1A.1.s3>Tal abordagem parte do documento de requisitos do sistema e propõem a especificação das interações entre o sistema e seus agentes (cenários), e então os cenários são especificados detalhadamente.</s><s id=**art1R.1.s4**
corresp='art1A.1.s4 art1A.1.s5'>Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.</s>

</text>

<text lang=en id=art1A>
<s id=art1A.1.s1 corresp=**art1R.1.s1**>A tool to support requirements trading is presented.</s><s id=art1A.1.s2
corresp=**art1R.1.s2**>The tool supports the ERACE approach.</s><s id=art1A.1.s3
corresp=**art1R.1.s3**>This approach starts from the system's requirement document and proposes to specify interactions between the system and its agents (scenarios), and then the scenarios are specified in detail.</s><s id=art1A.1.s4
corresp=**art1R.1.s4**>Heuristics to evolve from the requirements model to the analysis are also presented.</s><s id=art1A.1.s5
corresp=**art1R.1.s4**>An example to illustrates the approach is also presented.</s>

</text>

Figura 13 – Exemplo de um bitexto alinhado pelo GMA.

Além desses três arquivos de saída gerados para cada bitexto alinhado, um outro é produzido pelo SIMR contendo o mapeamento do bitexto, ou seja, as posições das palavras consideradas pontos de correspondência nos dois textos. Após serem processados, esses valores numéricos são transformados em uma lista bilíngüe de palavras (PB e inglês, nesta ordem) como mostra o Quadro 6.

Quadro 6: Lista de palavras consideradas pontos de correspondência pelo SIMR no alinhamento do bitexto art1R-art1A.

PB	Inglês
suporta	supports
ERACE	ERACE
documento	document
agentes	agents
cenários	scenarios
cenários	scenarios
heurísticas	Heuristics
modelo	model
caso	also
apresentado	presented

A próxima Seção (4.2.2) traz uma explicação sucinta do processo de otimização dos parâmetros citados nesta Seção, para o PB e o inglês.

4.2.2 Otimização dos Parâmetros

Como visto na Seção anterior, o SIMR utiliza vários parâmetros para encontrar o mapeamento do bitexto, são eles: o tamanho fixo da cadeia (*k*) e os limites para o LCSR, o nível de ambigüidade máximo da cadeia, a dispersão máxima do ponto e o desvio máximo do ângulo. Para garantir um bom desempenho do método, é aconselhado otimizar esses parâmetros para cada novo par de línguas ou tipo de texto utilizado.

Os parâmetros citados foram otimizados seguindo o processo descrito em Melamed (1996). Nesse processo foram utilizados dois bitextos alinhados, com cerca de 500 sentenças cada, escritos em PB e inglês. O primeiro bitexto, parte de um manual de PHP[24], foi submetido ao processo de otimização via *simulated annealing* (Vidal, 1993 apud Melamed, 2000). Essa técnica utiliza uma função que mede a diferença entre os pontos de correspondência verdadeiros e os mapeamentos do bitexto interpolados, produzidos com os parâmetros atuais. Em termos geométricos, a diferença é uma distância e a métrica usada nesse caso é a raiz da distância média ao quadrado (*root mean squared distance* - RMS).

Os valores gerados nesse processo foram validados com o alinhamento do segundo bitexto: parte da constituição brasileira de 1988[25]. Os valores dos parâmetros obtidos nesse processo são apresentados na Tabela 4.

Tabela 4: Parâmetros do SIMR otimizados para o PB e o inglês.

Parâmetros	Valor para PB
Tamanho da cadeia	6
Limite mínimo para LCSR	4
Nível de ambigüidade máximo da cadeia	0,11
Dispersão máxima do ponto	15
Desvio máximo do ângulo	0,65

A escolha de textos pertencentes a domínios distintos baseou-se em (Melamed, 1996) que afirma ser esta a situação ideal para a otimização dos parâmetros. Os corpora autêntico e pré-editado foram alinhados utilizando os parâmetros da Tabela 4 e também os parâmetros

[24] Versão em inglês disponível em http://www.php.net/manual/en/ e versão em português disponível em http://www.php.net/manual/pt_BR/index.php (17/02/2003).
[25] Versão em inglês disponível em e http://www.georgetown.edu/pdba/Constitutions/Brazil/english98.html e versão em português disponível em http://www.georgetown.edu/pdba/Constitutions/Brazil/brazil88.html (17/02/2003).

para o par português europeu – inglês, fornecidos com o método. Os resultados das avaliações com esses dois conjuntos de parâmetros são apresentados na Seção 7.1.

Capítulo 5

Métodos Lingüísticos de Alinhamento Sentencial

Os métodos lingüísticos diferem dos métodos empíricos por utilizarem informações específicas sobre as línguas envolvidas no processo de alinhamento de textos paralelos, como léxicos, listas de palavras âncoras, glossários e etiquetação morfológica. Portanto, recursos dependentes de língua, de difícil construção e que requerem conhecimento especializado das duas (ou mais) línguas envolvidas, como constatado na análise de seis sistemas de tradução automática inglês-portugês-inglês descrita em (Oliveira Jr. et al., 2000).

Essa classe de métodos não possui tantos representantes como as classes de métodos empíricos e híbridos devido, em parte, a alguns fatores como: as dificuldades de criação de recursos lingüísticos citadas anteriormente e a precisão satisfatória dos métodos empíricos. A boa precisão dos métodos empíricos frente à simplicidade de sua implementação incentiva a incorporação de recursos lingüísticos a esses métodos criando-se métodos híbridos ao invés da elaboração de novos métodos lingüísticos.

Mesmo assim, os métodos lingüísticos são muito importantes para o projeto PESA e por isso um deles foi escolhido como representante desta classe: o método lingüístico apresentado em (Papageorgiou et al., 1994; Piperidis et al., 2000). Este método, referenciado no restante deste texto como simplesmente "método lingüístico", utiliza a etiquetação morfológica das línguas envolvidas como recurso para o alinhamento dos textos paralelos.

Este método foi escolhido principalmente devido à precisão de 99% demonstrada nas avaliações realizadas em (Papageorgiou et al., 1994) e (Piperidis et al., 2000). Além disso, um dos principais interesses da avaliação de um método lingüístico é analisar as vantagens de um pré-processamento intenso dos corpora, como acontece no método lingüístico escolhido, em relação ao custo desse processo.

A próxima Seção (5.1) apresenta o método lingüístico, suas características, o processo de alinhamento, os detalhes da implementação e os passos necessários para sua adequação aos requisitos do projeto PESA. Os resultados da avaliação deste método são relatados na Seção 7.2.

5.1 O Método Lingüístico

O princípio básico do método lingüístico está relacionado ao ponto crítico da tradução: a preservação do significado (Piperidis et al., 2000). Tradicionalmente, as palavras de classe aberta – substantivos, verbos, adjetivos e advérbios – expressam a maior quantidade de informação significativa das sentenças. Assim, o critério de alinhamento do método lingüístico é a quantidade de palavras de classe aberta nas sentenças fonte e alvo, denominado pelos autores do método como *carga semântica*[26].

O método lingüístico implementado neste trabalho é uma versão da estrutura de programação dinâmica do método GC apresentada na Seção 4.1. Nesta versão os parâmetros passados para a estrutura de programação dinâmica são as cargas semânticas das sentenças ao invés de seus comprimentos.

Como já mencionado na Seção anterior, o método lingüístico apresentou uma precisão de 99% nas duas avaliações descritas em (Papageorgiou et al., 1994) e (Piperidis et al., 2000). O corpus usado nas duas avaliações era composto por sentenças de textos paralelos escritos em grego e inglês extraídos do corpus CELEX[27] e etiquetados com o etiquetador de Brill citado na Seção 3.2. Além da boa precisão, constatou-se que o modelo é robusto mesmo com erros de etiquetação. Outros detalhes sobre as avaliações do método lingüístico são apresentados na Seção 7.2.1.

A próxima Seção (5.1.1) apresenta o processo de alinhamento do método lingüístico e as peculiaridades de sua implementação no projeto PESA.

5.1.1 O Alinhamento

O método lingüístico alinha duas sentenças se, e somente se, suas cargas semânticas forem similares, ou seja, se a quantidade de substantivos, adjetivos, advérbios e verbos na sentença alvo for similar à quantidade destas classes na sentença fonte. Para que a similaridade semântica das sentenças possa ser verificada é necessário que os corpora possuam etiquetas (ou marcações) identificando as classes morfológicas das palavras. O processo de etiquetação dos corpora de teste (CAT e CPT) do projeto PESA foi apresentado na Seção 3.2. Neste processo foram gerados os dois corpora de teste (CATE e CPTE) utilizados na avaliação do

[26] A carga semântica de uma sentença é definida, neste caso, como a união de todas as classes abertas, ou etiquetas morfológicas, que podem ser atribuídas às palavras dessa sentença (Papageorgiou et al., 1994).
[27] O corpus CELEX é o sistema de documentação computadorizada na *European Community Law*, composto de regulamentos, artigos, recomendações, etc. (http://europa.eu.int/celex/htm/celex_en.htm).

método lingüístico. Um exemplo de um bitexto morfologicamente etiquetado é apresentado na Seção 3.2 (Figura 4).

Considerando-se que os textos paralelos são fornecidos para o método devidamente etiquetados, o primeiro passo é calcular a carga semântica das sentenças dos textos fonte e alvo. A carga semântica é calculada a partir de um modelo quantitativo construído aplicando-se Regressão Linear Múltipla a um conjunto de dados de exemplo manualmente alinhado no nível sentencial. Seja Y a soma das quantidades de etiquetas morfológicas atribuídas à sentença alvo e X_i a soma das quantidades de etiquetas na sentença fonte referentes aos verbos (X_1), aos substantivos (X_2), aos adjetivos (X_3) e aos advérbios (X_4), a dependência linear entre Y e X_i é apresentada em (6).

$$Y = b_0 + b_1 X_1 + b_2 X_2 + b_3 X_3 + b_4 X_4 + \varepsilon \quad (6)$$

em que os parâmetros b_i e ε são os coeficientes de regressão e o erro, respectivamente. Os coeficientes de regressão são estimados por meio do método dos mínimos quadrados e ε é estimado como sendo normalmente distribuído com média zero e variância σ^2.

Na avaliação efetuada no projeto PESA, os valores para os coeficientes de regressão e o erro foram estimados a partir de quatorze textos paralelos alinhados manualmente e selecionados aleatoriamente. O número total de palavras nesses quatorze pares de textos (4656) representa aproximadamente 10% do número total de palavras nos corpora autêntico e pré-editado (42924), sendo sete pares do CATE e sete do CPTE. Os textos utilizados na estimativa são mostrados no Quadro 7.

Quadro 7: Pares de textos paralelos selecionados para estimar os valores dos coeficientes e do erro da equação (6).

CATE	**CPTE**
art1	art8
art2	cgpi1
art11	es4
bd1	h10
es6	ic2
h7	mc1
sdpc8	sdpc5

A partir dos alinhamentos sentenciais presentes nestes textos e aplicando-se o método dos mínimos quadrados, estimou-se uma variância de 10,67 e a equação final de regressão apresentada em (7).

$$Y = 0{,}466 + 0{,}627\,X_1 + 1{,}03\,X_2 + 1{,}03\,X_3 + 1{,}28\,X_4 \quad (7)$$

As etiquetas referentes às classes abertas representadas na equação (2) como X_1, X_2, X_3 e X_4 são mostradas no Quadro 8 para os textos em PB e em inglês.

Quadro 8: Etiquetas das classes abertas referentes à X_1, X_2, X_3 e X_4.

	PB	**inglês**
X_1	VERB	VBD, VBN, VBP, VBG, VBZ, VB
X_2	N, NP	NNS, NN, NP, NPS, PP
X_3	ADJ	JJ, JJR, JJS
X_4	ADV	RB, RBR, RBS

Após se determinar as cargas semânticas de duas sentenças, a relação entre a sentença na língua alvo e a sentença na língua fonte, denotada por Y, é usada no cálculo da pontuação probabilística atribuída à comparação delas. Essa pontuação é calculada como a área sob N(0, σ^2) especificada pelo erro estimado e é usada em uma estrutura de programação dinâmica, como a apresentada no método GC (Seção 4.1). Uma visão geral desse processo é apresentada na Figura 14.

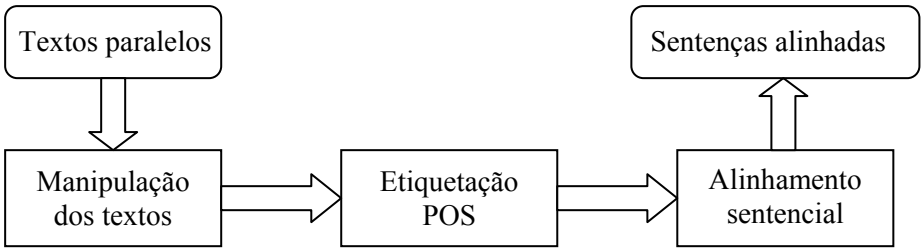

Figura 14 – Visão geral do processo de alinhamento sentencial (Piperidis et al, 2000).

Na implementação do método lingüístico foram utilizados, além dos arquivos originais do método GC (vide Seção 4.1) referentes à estrutura de programação dinâmica, novos programas para adequá-lo aos requisitos do projeto PESA. Da mesma forma que para os métodos empíricos, um programa recebe como parâmetro o arquivo com o corpus a ser alinhado (<corpus paralelo>, vide Seção 3.5) e gerencia todos os outros para a execução das três tarefas básicas: pré-processamento dos textos, alinhamento e pós-processamento dos textos.

Após o pré-processamento dos textos, no qual são removidas as etiquetas de início (<s>) e fim (</s>) de todas as sentenças, as cargas semânticas são calculadas e esses valores são passados como parâmetro para a estrutura de programação dinâmica da mesma forma que

no método GC. Assim, por exemplo, para o par de textos paralelos art1R.txt e art1A.txt a chamada ao GCalign (que faz a programação dinâmica) seria:

```
java GCalign -d '<BRK>' -i “7,4,15,12,\\<BRK\\>,6,4,16,7,5”
```

na qual, os comprimentos das sentenças (vide Seção 4.1.1) foram substituídos pelas suas cargas semânticas.

Os textos paralelos art1R.txt e art1A.txt são alinhados e a saída é salva no arquivo art1.txt no formato apresentado na Seção 3.5 (Figura 6). Os arquivos de entrada com marcações de alinhamento, como mostrado na Seção 3.5, também são retornados como a saída do método lingüístico. A Figura 15 traz um exemplo destes arquivos, os textos art1R.txt e art1A.txt, alinhados pelo método lingüístico. As etiquetas sublinhadas representam aquelas consideradas no cálculo da carga semântica das sentenças.

```
<text lang=pt id=art1R>
<p><s id=art1R.1.s1
corresp=art1A.1.s1>Neste PREP+PD artigo
N é VERB apresentada ADJ uma ART
ferramenta VERB para PREP validação N e
CONJ verificação N de PREP requisitos
N.</s><s id=art1R.1.s2
corresp=art1A.1.s2>Essa PRON ferramenta
VERB suporta VERB a ART abordagem
ADJ ERACE N.</s><s id=art1R.1.s3
corresp=art1A.1.s3>Tal PRON abordagem
ADJ parte N do PREP+ART documento N
de PREP requisitos N do PREP+ART
sistema N e CONJ propõem VERB a ART
especificação N das PREP+ART interações
N entre PREP o ART sistema N e CONJ
seus PRON agentes N (cenários N), e
CONJ então ADV os ART cenários N são
VERB especificados VERB detalhadamente
ADV.</s><s id=art1R.1.s4
corresp='art1A.1.s4 art1A.1.s5'>Também
ADV são VERB apresentadas VERB
heurísticas ADJ para PREP a ART evolução
N do PREP+ART modelo N de PREP
requisitos N para PREP modelos N de
PREP análise N, exemplificadas VERB
através ADV do PREP+ART estudo N de
PREP caso N apresentado ADJ.</s>
</p>
</text>
```

```
<text lang=en id=art1A>
<p><s id=art1A.1.s1 corresp=art1R.1.s1>A
DT tool NN to TO support VB requirements
NNS trading NN is VBZ presented
VBN.</s><s id=art1A.1.s2
corresp=art1R.1.s2>The DT tool NN
supports VBZ the DT ERACE JJ approach
NN.</s><s id=art1A.1.s3
corresp=art1R.1.s3>This DT approach NN
starts VBZ from IN the DT system NN 's
POS requirement NN document NN and CC
proposes VBZ to TO specify VB interactions
NNS between IN the DT system NN and CC
its PP$ agents NNS (scenarios NNS), and
CC then RB the DT scenarios NNS are VBP
specified VBN in IN detail NN.</s><s
id=art1A.1.s4
corresp=art1R.1.s4>Heuristics NP to TO
evolve VB from IN the DT requirements
NNS model NN to TO the DT analysis NN
are VBP also RB presented VBN.</s><s
id=art1A.1.s5 corresp=art1R.1.s4>An DT
example NN to TO illustrates VBZ the DT
approach NN is VBZ also RB presented
VBN.</s>
</p>
</text>
```

Figura 15 – Exemplo de um bitexto alinhado pelo método lingüístico.

Os resultados da avaliação do método lingüístico são apresentados na Seção 7.2.

Capítulo 6

Métodos Híbridos de Alinhamento Sentencial

A classe de métodos híbridos é a que tem atraído o maior número de propostas de técnicas, todas com o objetivo de unir a simplicidade dos métodos empíricos com a maior precisão oferecida pelo uso de recursos lingüísticos. Dessa forma, os métodos híbridos englobam as duas abordagens anteriormente apresentadas, agrupando em um único método as características e recursos dos métodos empíricos e lingüísticos.

Nesse contexto, serão apresentados a seguir dois métodos híbridos. O primeiro trata-se de uma extensão do método empírico GMA (apresentado na Seção 4.2) que utiliza uma lista de palavras âncoras (vide Seção 3.4), além de cognatos, para alinhar as sentenças. Este método recebeu a denominação de GSA+ no projeto ARCADE e a mesma denominação é usada neste texto. O segundo método híbrido é o *Translation Corpus Aligner* (TCA) que utiliza diversos critérios de alinhamento como: nomes próprios, caracteres especiais, uma lista de palavras âncoras e o tamanho das sentenças.

Os dois métodos híbridos escolhidos para implementação no projeto PESA usam uma lista de palavras âncoras (LPA) como um de seus critérios de alinhamento. O processo de construção desta LPA é descrito na Seção 3.4, onde também é apresentado um trecho da LPA (Quadro 5). O formato da LPA apresentado neste quadro não é o mesmo processado pelos métodos. Um trecho da LPA no formato passado como parâmetro para os métodos híbridos é apresentado na Figura 16, na qual uma palavra (ou expressão multipalavras) fonte é separada de sua correspondente (possivelmente uma expressão multipalavras) na língua alvo pela seqüência de caracteres " <> ".

Nessa figura, o caractere * indica truncamento da palavra e permite o casamento parcial das palavras da LPA com as palavras sendo avaliadas. Assim, por exemplo, para o par ambient* <> environment* presente na LPA poderiam ser gerados pontos de correspondência para os pares ambiente <> environment, ambiental <> environmental, ambiente <> environments, ambientes <> environments, etc.

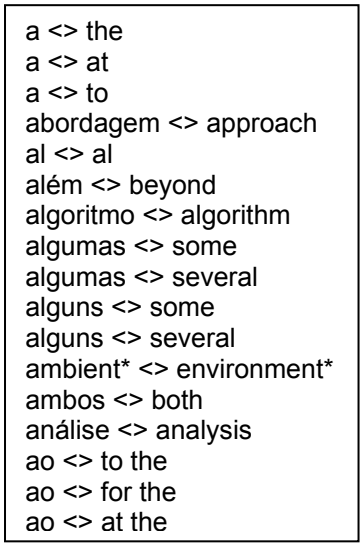

Figura 16 – Exemplo de um trecho da LPA no formato passado como parâmetro para os métodos.

Além do casamento parcial, os métodos híbridos implementados também tratam casos de palavras com apóstrofo no idioma inglês. Por exemplo, na frase "algorithm's performance" a palavra "algorithm" é reduzida a esta forma para que a entrada na LPA possa ser encontrada e o casamento com a palavra "algoritmo" seja feito.

O GSA+ foi selecionado para ser implementado, principalmente por ser uma extensão do método empírico GMA possibilitando, assim, uma análise do impacto da utilização de um recurso lingüístico em um método empírico. Já o principal motivo para a seleção do TCA foi o fato deste método ter sido utilizado em um projeto envolvendo o português europeu (Santos & Oksefjell, 2000), o que instiga a curiosidade de verificar o seu desempenho em relação a outro português, o brasileiro.

As próximas seções apresentam os métodos híbridos GSA+ e TCA. A Seção 6.1 descreve resumidamente o método GSA+, já que ele pouco se difere do método apresentado na Seção 4.2 (o GMA). A Seção 6.2, por sua vez, apresenta o método TCA, suas características, o processo de alinhamento, os detalhes da implementação e os passos necessários para sua adequação aos requisitos do projeto PESA.

6.1 Método GSA+

O GSA+ é praticamente o mesmo método GMA descrito na Seção 4.2, a única diferença, que o torna um método híbrido, está na fase de geração dos pontos de correspondência candidatos pelo algoritmo SIMR. No GSA+, além das palavras consideradas cognatas (utilizando a

LCSR), outros pontos de correspondência candidatos são gerados a partir de uma lista de palavras âncoras (LPA).

Os pares de palavras fonte e alvo na área sob análise são buscados na LPA e, caso ambas as palavras sejam encontradas, o par é considerado um ponto de correspondência. Assim, dada uma palavra fonte PF e uma palavra alvo PA, o par (PF, PA) é considerado um ponto de correspondência candidato se e somente se a correspondência entre PF e PA estiver expressa em LPA, ou seja, se existir uma entrada em LPA no formato: PF <> PA ou PA <> PF.

Além da LPA, outras listas podem ser usadas para filtrar as correspondências espúrias. Essas listas, denominadas *stoplists*, são compostas por palavras de classes fechadas e/ou pares de falsos cognatos. No projeto PESA, o GSA+ foi avaliado com e sem a utilização destas *stoplists* e os resultados são apresentados na Seção 7.3.

O método GSA+ foi avaliado em (Melamed, 2000) utilizando-se o corpus Hansard com textos paralelos inglês-francês, apresentando uma precisão de até 98,9% em textos com traduções mais literais alinhados previamente no nível de parágrafo. Em uma outra avaliação efetuada no projeto ARCADE durante a tarefa de alinhamento sentencial, o GSA+ obteve uma precisão de 95,6% em textos técnicos sem correspondências cruzadas. A partir destes valores pode-se verificar que o GSA+ se saiu melhor do que o GMA em ambas as avaliações. Esta comparação também foi feita para os corpora CAT e CPT (vide Seção 7.3.1).

Com relação ao código, a única diferença entre o GSA+ e o GMA está no arquivo de configuração passado como parâmetro para o método. Este arquivo deve indicar, além dos parâmetros padrão (os mesmos do GMA, vide Seção 4.2.2), a localização da LPA e das *stoplists* (se estas forem usadas).

O GSA+, da mesma forma que o método GMA, executa as funções de pré-processamento dos textos, alinhamento e pós-processamento e gera os mesmos arquivos de saída (vide Seção 4.2.1).

Os arquivos art1R e art1A alinhados pelo método GMA (vide Figura 13) também foram alinhados da pelo método GSA+ e o alinhamento produzido foi igual ao do método empírico. Porém, pontos de correspondência diferentes foram considerados neste alinhamento como pode ser observado na Figura 17.

<text lang=pt id=art1R>
<s id=**art1R.1.s1**
corresp=art1A.1.s1>Neste artigo é apresentada uma ferramenta para validação e verificação de requisitos.</s><s id=**art1R.1.s2** corresp=art1A.1.s2>Essa ferramenta suporta a abordagem ERACE.</s><s id=**art1R.1.s3**
corresp=art1A.1.s3>Tal abordagem parte do documento de requisitos do sistema e propõem a especificação das interações entre o sistema e seus agentes (cenários), e então os cenários são especificados detalhadamente.</s><s id=**art1R.1.s4**
corresp='art1A.1.s4 art1A.1.s5'>Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.</s>

</text>

<text lang=en id=art1A>
<s id=art1A.1.s1 corresp=**art1R.1.s1**>A tool to support requirements trading is presented.</s><s id=art1A.1.s2
corresp=**art1R.1.s2**>The tool supports the ERACE approach.</s><s id=art1A.1.s3
corresp=**art1R.1.s3**>This approach starts from the system's requirement document and proposes to specify interactions between the system and its agents (scenarios), and then the scenarios are specified in detail.</s><s id=art1A.1.s4
corresp=**art1R.1.s4**>Heuristics to evolve from the requirements model to the analysis are also presented.</s><s id=art1A.1.s5
corresp=**art1R.1.s4**>An example to illustrates the approach is also presented.</s>

</text>

Figura 17 – Exemplo de um bitexto alinhado pelo GSA+.

Da mesma forma que o método empírico GMA, além dos três arquivos de saída gerados para cada bitexto alinhado, um outro é produzido pelo SIMR contendo o mapeamento do bitexto. No caso do GSA+ esses mapeamentos incluem também as palavras âncoras encontradas nos bitextos, como pode ser observado no Quadro 9.

Quadro 9: Exemplo de uma lista de pontos de correspondência gerada pelo SIMR (no GSA+).

PB	Inglês	
ferramenta	tool	palavra âncora
ferramenta	tool	palavra âncora
suporta	supports	
ERACE	ERACE	
documento	document	
requisitos	requirement	palavra âncora
agentes	agents	
cenários	scenarios	
cenários	scenarios	
heurísticas	Heuristics	
modelo	model	
apresentado	presented	

Os resultados da avaliação do método GSA+ são apresentados na Seção 7.3.

6.2 Método TCA

O método TCA foi desenvolvido no âmbito do projeto *English-Norwegian Parallel Corpus* (ENPC)[28] com o objetivo de alinhar automaticamente textos em inglês e norueguês (Hofland, 1996).

O TCA alinha as sentenças dos textos paralelos levando em consideração várias informações, como: nomes próprios, etiquetas e caracteres especiais, uma lista de palavras âncoras e o tamanho das sentenças, em caracteres.

Na avaliação realizada em (Hofland, 1996), com um corpus paralelo inglês-norueguês desenvolvido no projeto ENPC, o método TCA obteve uma precisão de 98%. Em outra avaliação, feita em (Santos & Oksefjell, 2000), o método obteve 97,1% de precisão em um corpus inglês-português europeu também desenvolvido como parte do projeto ENPC.

O método TCA foi implementado em Perl e é o único método do projeto PESA para plataforma Windows.

A Seção seguinte (6.2.1) apresenta o processo de alinhamento do método TCA, detalhes de sua implementação e adequação aos requisitos do projeto PESA.

6.2.1 O Alinhamento

Como mencionado, o TCA utiliza diversos critérios para determinar os pontos de correspondência entre duas sentenças, entre eles uma lista de palavras âncoras (LPA). A LPA do projeto PESA possui o formato apresentado na Figura 16 o qual difere um pouco do original proposto em (Hofland, 1996). No formato utilizado no projeto PESA, uma palavra (ou expressão multipalavras) na língua fonte é separada de sua correspondentes na língua alvo (uma palavra ou expressão multipalavras) pela seqüência de caracteres " <> ".

Além da LPA, o TCA utiliza outros critérios para determinar os pontos de correspondência entre duas sentenças como: nomes próprios, etiquetas e caracteres especiais e o tamanho das sentenças, em caracteres.

Assim, o TCA aplica cada um destes critérios nas *n* sentenças sob consideração. Esta "janela" de *n* sentenças é movida pelos textos fonte e alvo com uma sobreposição de 5 sentenças e nunca é movida ao mesmo tempo em ambos os textos. O número de sentenças na janela (*n*) é um dos parâmetros do método.

---

[28] Site do ENPC: http://www.hf.uio.no/iba/prosjekt/ (17/02/2003).

O primeiro passo do programa consiste em verificar a existência de palavras âncoras, palavras com inicial maiúscula (candidatas a nomes próprios) e alguns caracteres especiais como '?' e '!' em cada uma das *n* sentenças fonte e alvo da janela. Três listas são criadas com as palavras e caracteres encontrados: a lista de palavras âncoras, a lista de candidatas a nomes próprios e a lista de caracteres especiais.

As listas criadas para o bitexto art1R-art1A do CAT são mostradas na Figura 18.

art1R.1.s1: Neste artigo é apresentada uma ferramenta para validação e verificação de requisitos. **Palavras âncoras**: é **Palavras com inicial maiúscula**: Neste **Caracteres especiais**: **art1R.1.s2**: Essa ferramenta suporta a abordagem ERACE. **Palavras âncoras**: abordagem **Palavras com inicial maiúscula**: Essa, ERACE **Caracteres especiais**: **art1R.1.s3**: Tal abordagem parte do documento de requisitos do sistema e propõem a especificação das interações entre o sistema e seus agentes (cenários), e então os cenários são especificados detalhadamente. **Palavras âncoras**: abordagem, parte **Palavras com inicial maiúscula**: Tal **Caracteres especiais**: **art1R.1.s4**: Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado. **Palavras âncoras**: são **Palavras com inicial maiúscula**: Também **Caracteres especiais**:	**art1A.1.s1**: A tool to support requirements trading is presented. **Palavras âncoras**: requirements, is **Palavras com inicial maiúscula**: A **Caracteres especiais**: **art1A.1.s2**: The tool supports the ERACE approach. **Palavras âncoras**: approach **Palavras com inicial maiúscula**: The, ERACE **Caracteres especiais**: **art1A.1.s3**: This approach starts from the system's requirement document and proposes to specify interactions between the system and its agents (scenarios), and then the scenarios are specified in detail. **Palavras âncoras**: approach, system, system, are **Palavras com inicial maiúscula**: This **Caracteres especiais**: **art1A.1.s4**: Heuristics to evolve from the requirements model to the analysis are also presented. **Palavras âncoras**: requirements, analysis, are **Palavras com inicial maiúscula**: Heuristics **Caracteres especiais**: **art1A.1.s5**: An example to illustrates the approach is also presented. **Palavras âncoras**: example, approach, is **Palavras com inicial maiúscula**: Na **Caracteres especiais**:

Figura 18 – Listas com as informações das sentenças do bitexto art1R-art1A.

Após a leitura das *n* sentenças nas duas línguas, uma matriz *n*x*n* é construída inicialmente com todas as células iguais à zero. As listas geradas para cada uma das sentenças na língua fonte é então checada com relação às listas geradas para cada uma das sentenças na língua alvo. Se uma correspondência entre uma palavra âncora na sentença fonte *i* e uma palavra âncora na sentença alvo *j* estiver presente em LPA, então o valor da célula *i,j* da

matriz é incrementado de 1. O mesmo ocorre para palavras candidatas a nomes próprios e caracteres especiais, sendo que nestes casos as palavras e caracteres nas listas fonte e alvo devem ser exatamente iguais para que o valor da matriz seja incrementado. A matriz resultante da combinação das sentenças fonte e alvo da Figura 18 é apresentada na Tabela 5. As linhas correspondem ao texto original escrito em PB (nesse exemplo, o art1R) e as colunas correspondem à tradução para o inglês (nesse caso, o art1A).

Tabela 5*:* Matriz resultante da combinação das sentenças fonte e alvo da Figura 18.

	art1A.1.s1	**art1A.1.s2**	**art1A.1.s3**	**art1A.1.s4**	**art1A.1.s5**
art1R.1.s1	1	0	0	0	1
art1R.1.s2	1	2	1	0	1
art1R.1.s3	0	1	1	0	1
art1R.1.s4	0	0	1	1	0

Além dessas listas, o programa de alinhamento utiliza cognatos para encontrar os pontos de correspondência entre as sentenças. Duas técnicas podem ser usadas para encontrar os cognatos: truncamento e coeficiente de similaridade de Dice (Hofland, 1996).

No truncamento, se os k primeiros caracteres são iguais em duas palavras de línguas diferentes, então elas são consideradas cognatas (com $k > 0$ dependente da língua em questão). O coeficiente de similaridade de Dice (S), por sua vez, verifica quantas seqüências de duas letras (bigramas) as palavras têm em comum, expressando essa quantidade como uma porcentagem em relação ao número total de seqüências. Assim:

$$S = \frac{2a}{(b+c)} \tag{8}$$

em que a é a quantidade de bigramas em comum nas duas palavras e b e c são os números totais de bigramas em cada uma das duas palavras. Por exemplo, o coeficiente de Dice para as palavras *phenomenal* e *fenomenal* é:

ph he en no om me en na al X fe en no om me en na al

$$S = \frac{2*7}{9+8} = \frac{14}{17} = 0{,}82$$

Na implementação do TCA no projeto PESA optou-se pelo uso do coeficiente de Dice. O valor 0,64 mostrou-se um bom limite para o par PB-inglês como demonstrado na Seção 6.2.2. Palavras com coeficiente de Dice maior ou igual a este valor são consideradas cognatas.

Os cognatos encontrados para as sentenças da Figura 18 são apresentados no Quadro 10.

Quadro 10: Cognatos encontrados nas sentenças da Figura 18.

Sentença Fonte	Sentença Alvo	Cognatos
art1R.1.s2	art1A.1.s1	suporta <=> support
art1R.1.s2	art1A.1.s2	suporta <=> supports
art1R.1.s3	art1A.1.s3	agentes <=> agents documento <=> document
art1R.1.s4	art1A.1.s4	modelos <=> model modelo <=> model

A existência de um par de palavras cognatas também incrementa o valor da matriz. Assim, a matriz da Tabela 5 após o incremento referente aos cognatos é apresentada na Tabela 6. Os números na primeira coluna e na primeira linha se referem ao tamanho das sentenças fonte e alvo, respectivamente, medido em caracteres.

Tabela 6: Matriz da Tabela 5 incrementada de acordo com a existência de cognatos.

		52	37	191	84	57
		art1A.1.s1	**art1A.1.s2**	**art1A.1.s3**	**art1A.1.s4**	**art1A.1.s5**
85	**art1R.1.s1**	1	0	0	0	1
42	**art1R.1.s2**	1	3	1	0	1
196	**art1R.1.s3**	0	1	3	0	1
154	**art1R.1.s4**	0	0	1	3	0

Após a construção da matriz, o próximo passo é encontrar uma combinação entre as sentenças nas duas línguas de tal forma que a soma dos valores da matriz seja maximizada. As combinações possíveis são: 1-1, 1-0, 0-1, 2-1 e 1-2. O método TCA não considera alinhamentos *x-y* com $x, y \geq 2$. Assim, para cada posição *i,j* na matriz da Tabela 6 as cinco combinações mostradas em (9) são testadas.

1. P_i com I_j (1-1)
2. P_i com I_{j+1} (1-0)
3. P_{i+1} com I_j (0-1) (9)
4. P_i e P_{i+1} com I_j (2-1)
5. P_i com I_j e I_{j+1} (1-2)

em que P_i representa a sentença em PB que aparece na i-ésima posição da janela, e I_j, a sentença em inglês que aparece na j-ésima posição da janela.

Para cada combinação, o valor da matriz é ajustado de acordo com a correspondência entre o tamanho das sentenças, em caracteres, nas duas línguas. O índice de correspondência é

medido como o valor absoluto da diferença de tamanho dividido pela média dos dois tamanhos como mostra a equação (10).

$$ind = \frac{2 \times |tamanho1 - tamanho2|}{(tamanho1 + tamanho2)} \quad (10)$$

na qual tamanho1 é o tamanho da sentença fonte e tamanho2, o da sentença alvo.

Uma boa correspondência (baixo índice) aumenta o valor na matriz e uma má correspondência (alto índice) diminui o valor. Valores limites para esses índices são passados como parâmetros do programa – *lowind, highind* e *toohigh* – e devem ser otimizados para cada novo par de línguas e domínio do corpus. Assim, um baixo índice é menor ou igual a *lowind* e um alto índice é maior ou igual a *highind*. O terceiro parâmetro é utilizado para eliminar combinações 1-2 e 2-1 com índices de correspondência muito altos (maior do que *toohigh*). Esses parâmetros do TCA foram otimizados para o projeto PESA como explicado na Seção 6.2.2.

Dessa forma, a partir da matriz dada na Tabela 6, os cálculos para se determinar qual a melhor combinação quando, por exemplo, $i = 4$ e $j = 4$, são mostrados na Tabela 7. Nesse exemplo, tem-se que a melhor combinação de todas apresentadas em (9) é a 5, pois o valor da matriz para a linha 4 e as colunas 4 e 5 é aumentado devido ao baixo índice de correspondência calculado para esta combinação. No arquivo de saída mostrado na Figura 19, pode-se verificar que a sentença fonte art1R.1.s4 corresponde às sentenças alvo art1A.1.s4 e art1A.1.s5.

Tabela 7: Cálculo dos índices de correspondência para as sentenças art1R.1.s4, art1A.1.s4 e art1A.1.s5.

Combinação	Valor da matriz	Índice de correspondência
1	3	0,59
2	0	0,92
3	-	-
4	-	-
5	3 + **1**	0,09

Todas as possíveis combinações para as *n* sentenças da janela são testadas, a janela é expandida com uma sobreposição de 5 sentenças e, quando todas as combinações tiverem sido testadas, a combinação de maior valor é considerada como o alinhamento resultante.

O programa implementado para gerenciar todo o processo de alinhamento do método TCA, o TCAalign, recebe como parâmetro o arquivo com o corpus a ser alinhado (<corpus

paralelo>, vide Seção 3.5) e executa as funções básicas de todos os outros métodos já descritos nas Seções anteriores: pré-processamento dos textos, alinhamento e pós-processamento dos textos.

O bitexto alinhado pelo TCA para o bitexto de entrada art1R-art1A é mostrado na Figura 19.

<text lang=pt id=art1R> <s id=**art1R.1.s1** corresp=art1A.1.s1>Neste artigo é apresentada uma ferramenta para validação e verificação de requisitos.</s><s id=**art1R.1.s2** corresp=art1A.1.s2>Essa ferramenta suporta a abordagem ERACE.</s><s id=**art1R.1.s3** corresp=art1A.1.s3>Tal abordagem parte do documento de requisitos do sistema e propõem a especificação das interações entre o sistema e seus agentes (cenários), e então os cenários são especificados detalhadamente.</s><s id=**art1R.1.s4** corresp='art1A.1.s4 art1A.1.s5'>Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.</s> </text>	<text lang=en id=art1A> <s id=art1A.1.s1 corresp=**art1R.1.s1**>A tool to support requirements trading is presented.</s><s id=art1A.1.s2 corresp=**art1R.1.s2**>The tool supports the ERACE approach.</s><s id=art1A.1.s3 corresp=**art1R.1.s3**>This approach starts from the system's requirement document and proposes to specify interactions between the system and its agents (scenarios), and then the scenarios are specified in detail.</s><s id=art1A.1.s4 corresp=**art1R.1.s4**>Heuristics to evolve from the requirements model to the analysis are also presented.</s><s id=art1A.1.s5 corresp=**art1R.1.s4**>An example to illustrates the approach is also presented.</s> </text>

Figura 19 – Exemplo de um bitexto alinhado pelo TCA.

A próxima Seção (6.2.2) traz uma explicação sucinta do processo de otimização dos parâmetros citados nesta Seção para o PB e o inglês.

6.2.2 Otimização dos Parâmetros

Como visto na Seção anterior, o TCA utiliza vários parâmetros em seu processo de alinhamento sentencial, são eles: o número de sentenças na janela (*n*) e os limites para o coeficiente de Dice e os índices de correspondência de tamanho (*lowind, highind* e *toohigh*). Esses parâmetros foram otimizados para os corpora usados no projeto PESA como mostrado a seguir.

O número de sentenças na janela foi considerado o mesmo proposto pelo autor (*n* = 15) em (Hofland, 1996), já que os corpora de teste CAT e CPT são compostos por textos pequenos em que o número médio de sentenças é 6,55.

O limite para o coeficiente de Dice foi determinado com base em exemplos de cognatos para o par PB-inglês. O Quadro 11 traz alguns destes exemplos. O valor limite considerado para os casos estudados e utilizado no projeto PESA foi de 0,64.

Quadro 11: Exemplos de cognatos para o par PB-inglês.

PB	Inglês	Coeficiente de Dice
ministro	minister	0,71
instantâneo	instant	0,75
erros	errors	0,67
apresentação	presentation	0,64
excelente	excellent	0,87

Os demais parâmetros referem-se ao índice de correspondência de tamanho calculado para sentenças fonte e alvo na janela do TCA. Esses parâmetros foram determinados com base em uma amostra dos corpora de teste como mostra a Tabela 8.

Tabela 8: Valores dos índices de correspondência calculados para alguns textos do CAT e CPT.

CAT	Índice			CPT	Índice		
	Menor	Médio	Maior		Menor	Médio	Maior
art1	0,03	0,22	0,48	art8	0,02	0,15	0,84
es4	0,02	0,25	0,46	art12	0,11	0,17	0,22
es8	0,12	0,16	0,21	h1	0,01	0,18	0,35
h6	0,02	0,20	0,35	h5	0,06	0,14	0,26
ic3	0,04	0,21	0,43	ic6	0,04	0,08	0,14
sdpc3	0,05	0,13	0,49	sdpc1	0	0,10	0,23
Médias	**0,05**	**0,20**	**0,40**		**0,04**	**0,14**	**0,34**

Assim, foram considerados como valores para *lowind, highind* e *toohigh*: 0,20, 0,40 e 0,50, respectivamente.

A Tabela 9 traz os valores dos parâmetros do TCA otimizados para o projeto PESA como descrito nesta Seção.

Tabela 9: Parâmetros do TCA.

Parâmetro	Valor Padrão
n	15
limite do coeficiente de Dice	0,64
lowind	0,20
highind	0,40
toohigh	0,50

Os resultados da avaliação do método TCA são apresentados na Seção 7.3.

Capítulo 7

Avaliação dos Métodos de Alinhamento Sentencial

Após a submissão dos corpora de teste aos métodos de alinhamento sentencial, é necessário que se avalie o desempenho dos mesmos comparando-se o alinhamento por eles produzido com os alinhamentos considerados ideais (corpora de referência). Para isso, as pesquisas atuais nesse campo utilizam três métricas: *precision, recall* e *F-measure*, calculadas com base no alinhamento de referência[29].

Precision é a porcentagem de alinhamentos corretos em relação a todos que foram propostos nos textos paralelos que compõem um corpus. *Recall* é a porcentagem de alinhamentos corretos, entre todos os possíveis (no corpus de referência). E *F-measure* é a medida de freqüência, calculada como o dobro da razão entre o produto *recall* x *precision* e a soma *recall* + *precision* (Véronis & Langlais, 2000).

Dessa forma, *precision* indica a capacidade do método de alinhamento em encontrar as correspondências corretas. Já *recall* indica a capacidade do método de alinhamento em encontrar as correspondências. Por fim, *F-measure* combina as duas anteriores em uma única métrica eficiente.

Portanto:

$$precision = \frac{NúmeroAlinhamentosCorretos}{NúmetoAlinhamentos\,\mathrm{Pr}\,opostos} \tag{11}$$

$$recall = \frac{NúmeroAlinhamentosCorretos}{NúmeroAlinhamentos\,\mathrm{Re}\,ferência} \tag{12}$$

$$F = 2\frac{recall \times precision}{recall + precision} \tag{13}$$

[29] As métricas *precision, recall* e *F-measure* são usadas neste texto com suas denominações em inglês porque esta é a forma na qual são mais conhecidas, mas podem ser encontradas em português como precisão, cobertura e medida-f, respectivamente.

Precision mede a consistência: quanto maior, maior o número de alinhamentos corretos dentre os encontrados. *Recall*, por sua vez, pode ser entendida como uma medida de completude: quanto maior *recall*, maior a capacidade do método em encontrar alinhamentos. Já *F-measure* mede a distância entre *recall* e *precision*, e quanto maior, mais próximos são esses valores, portanto, maior a capacidade de o método encontrar alinhamentos sendo eles corretos.

Dessa forma, se *precision* for 1, todos os alinhamentos propostos estão corretos, mas não se garante que todos os alinhamentos existentes no alinhamento de referência foram encontrados. Por outro lado, um *recall* igual a 1 indica que todos os alinhamentos existentes no corpus de referência foram encontrados, mas nada garante que alinhamentos errados também não tenham sido propostos. A situação ideal é, portanto, quando *recall* e *precision* são 1, o que caracteriza *F-measure* também igual a 1, e significa que o método alinhou perfeitamente os bitextos, encontrando corretamente todos os alinhamentos existentes.

Após a implementação dos métodos apresentados nos capítulos anteriores (Capítulo 4, Capítulo 5 e Capítulo 6) e da submissão dos corpora de teste CAT, CPT, CATE e CPTE – os dois últimos no caso específico do método lingüístico –, os corpora alinhados por cada método foram avaliados individualmente segundo os critérios explicados a seguir.

- **Métricas** – os valores de *precision*, *recall* e *F-measure* foram calculados para os corpora autêntico (CAT ou CATE) e pré-editado (CPT ou CPTE) alinhados pelo método, com base nos corpora de referência (CAR e CPR);

- **Categorias de alinhamento** – a quantidade de alinhamentos encontrados pelo método em cada categoria (0-1, 1-0, 1-1, 1-2, 2-1 e 2-2) foi comparada à quantidade dos corpora de referência;

- **Taxa de erro (ou acerto) por categoria** – além da análise da quantidade de alinhamentos em cada categoria, analisou-se também a taxa de erro (ou acerto) em cada uma delas para determinar em quais delas o método obteve pior (ou melhor) desempenho.

- **Avaliação comparativa** – nos casos em que mais de um método da mesma classe foi avaliado (os empíricos e os híbridos, neste projeto), fez-se uma avaliação comparativa entre eles baseada na quantidade de alinhamentos corretos, errados e parcialmente corretos em cada um.

No projeto PESA, os métodos foram avaliados considerando-se, além dos casos de alinhamentos corretos e errados, um caso não abordado pelas métricas *precision, recall* e *F-measure*: o alinhamento parcialmente correto. Um exemplo de alinhamento parcialmente correto é mostrado no Quadro 13. A versão (totalmente) correta do alinhamento é apresentada no Quadro 12.

Quadro 12: Exemplo de um alinhamento sentencial (totalmente) correto.

PB	Inglês
<s id=**art1R.1.s4** corresp='**art1A.1.s4 art1A.1.s5**'>Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.</s>	<s id=**art1A.1.s4** corresp=**art1R.1.s4**>Heuristics to evolve from the requirements model to the analysis are also presented.</s><s id=**art1A.1.s5** corresp=**art1R.1.s4**>An example to illustrates the approach is also presented.</s>

Quadro 13: Exemplo de um alinhamento sentencial parcialmente correto.

PB	Inglês
<s id=**art1R.1.s4** corresp=**art1A.1.s4**>Também são apresentadas heurísticas para a evolução do modelo de requisitos para modelos de análise, exemplificadas através do estudo de caso apresentado.</s>	<s id=**art1A.1.s4** corresp=**art1R.1.s4**>Heuristics to evolve from the requirements model to the analysis are also presented.</s>
	<s id=**art1A.1.s5** corresp=''>An example to illustrates the approach is also presented.</s>

A partir da constatação dessa limitação, a análise da taxa de erro dos métodos foi feita verificando-se a quantidade de alinhamentos corretos – total e parcialmente – e errados nos corpora alinhados por eles.

As próximas subseções trazem os resultados da avaliação de cada método separados por classe. A Seção 7.1 apresenta a avaliação dos métodos empíricos; a Seção 7.2, a avaliação do método lingüístico; e a Seção 7.3; a dos métodos híbridos. Além da avaliação dos métodos segundo os critérios descritos anteriormente, também são apresentados alguns exemplos de bitextos alinhados por eles (art10R-art10A, es7R-es7A, es12R-es12A, art8R-art8A e bd1R-bd1A). Por fim, a Seção 7.4 apresenta as conclusões dessas avaliações.

7.1 Avaliação dos Métodos Empíricos

Os métodos empíricos analisados no projeto PESA – GC e GMA – foram implementados e avaliados segundo os critérios citados no início deste capítulo e os resultados são apresentados a seguir.

Com relação às métricas *precision, recall* e *F-measure*, os valores calculados para os corpora de teste (CAT e CPT) após serem alinhados pelo método empírico GC são apresentados na Tabela 10.

Tabela 10: Métricas calculadas para os corpora alinhados pelo método empírico GC.

Métricas	**CAT**	**CPT**
precision	0,9125	0,9759
recall	0,9012	0,9736
F	0,9068	0,9747

Com base nos valores da Tabela 10, pode-se perceber que o desempenho do método GC no CPT foi melhor do que no CAT comprovando, assim, o que já havia sido relatado na literatura: o alinhamento sentencial é mais eficaz em textos limpos (sem erros gramaticais ou de tradução) (Gaussier et al., 2000).

No caso do método GMA, realizaram-se duas avaliações distintas utilizando-se os parâmetros calculados para o português europeu e o inglês, fornecidos com o método; e para o PB e o inglês, otimizados nesse trabalho. A Tabela 11 apresenta os valores desses parâmetros sob as denominações português europeu e português brasileiro (PB), já que a outra língua (o inglês) é a mesma em ambos.

Tabela 11: Parâmetros para o português europeu e o brasileiro quando alinhados com o inglês.

Parâmetros	**Português Brasileiro**	**Português Europeu**
Tamanho da cadeia	6	8
Limite mínimo para LCSR	4	8
Nível de ambigüidade máximo da cadeia	0,11	0,26
Dispersão máxima do ponto	15	19
Desvio máximo do ângulo	0,65	0,66

A Tabela 12 apresenta os valores das métricas calculados para o CAT e o CPT após serem alinhados pelo método GMA com os dois conjuntos de parâmetros da Tabela 11.

Tabela 12: Métricas calculadas para os corpora alinhados pelo GMA com os parâmetros da Tabela 11.

	Português Europeu		**Português Brasileiro**	
Métricas	**CAT**	**CPT**	**CAT**	**CPT**
precision	0,9410	0,9856	0,9485	0,9904
recall	0,9457	0,9880	0,9556	0,9928
F	0,9433	0,9868	0,9520	0,9916

Os valores da Tabela 12 não evidenciam uma diferença significativa entre os valores calculados para os corpora alinhados com os parâmetros para o português europeu e o brasileiro, por isso optou-se pela utilização dos valores para o PB, já que estes foram maiores

e o PB é o idioma em estudo no projeto PESA. Esta tabela também comprova o fato do alinhamento sentencial ser mais eficaz em textos sem ruídos (CPT).

Outra análise efetuada com os textos alinhados pelos métodos GC e GMA examinou quais são as categorias de alinhamentos mais freqüentes em cada um desses métodos comparadas às categorias dos corpora de referência. Os resultados dessa análise para os métodos GC e GMA são apresentados na Tabela 12 e na Tabela 14, respectivamente.

Tabela 13: Análise das categorias de alinhamentos dos corpora alinhados pelo método GC.

	Corpus de Referência		**Corpus Alinhado pelo GC**	
Categoria	**Autêntico**	**Pré-editado**	**Autêntico**	**Pré-editado**
0-1 ou 1-0	6	2	-	-
1-1	353	395	348	395
2-1 ou 1-2	41	17	48	19
2-2	4	2	4	1
2-3 ou 3-2	1	-	-	-
Total	**405**	**416**	**400**	**415**

Tabela 14: Análise das categorias de alinhamentos dos corpora alinhados pelo método GMA.

	Corpus de Referência		**Corpus Alinhado pelo GMA**	
Categoria	**Autêntico**	**Pré-editado**	**Autêntico**	**Pré-editado**
0-1 ou 1-0	6	2	5	1
1-1	353	395	359	398
2-1 ou 1-2	41	17	43	18
2-2	4	2	1	-
2-3 ou 3-2	1	-	-	-
Total	**405**	**416**	**408**	**417**

Embora em alguns casos os números apresentados nas tabelas anteriores para os corpora de referência e os corpora alinhados pelos métodos sejam bem próximos, nada se pode dizer sobre sua correção. Para isso, é preciso analisar a taxa de erro dos métodos nestas categorias considerando-se também os alinhamentos parcialmente corretos. Os resultados da análise da taxa de erro dos métodos, por categoria de alinhamento, são apresentados na Tabela 15 (método GC) e na Tabela 16 (método GMA).

Tabela 15: Análise da taxa de erro do método GC.

	CAT			**CPT**		
Categoria	**Parcialmente**	**Corretos**	**Errados**	**Parcialmente**	**Corretos**	**Errados**
0-1	0	0	0	0	0	0
1-0	0	0	0	0	0	0
1-1	12	330	6	3	389	3
1-2	8	27	0	2	14	0
2-1	3	7	3	1	2	0
2-2	2	1	1	0	0	1
Total	**25**	**365**	**10**	**6**	**405**	**4**

Tabela 16: Análise da taxa de erro do método GMA.

	CAT			CPT		
Categoria	**Parcialmente**	**Corretos**	**Errados**	**Parcialmente**	**Corretos**	**Errados**
0-1	0	0	2	0	0	0
1-0	0	1	2	0	1	0
1-1	9	347	3	2	395	1
1-2	2	31	0	1	15	0
2-1	1	7	2	0	2	0
2-2	0	1	0	0	0	0
Total	**12**	**387**	**9**	**3**	**413**	**1**

Como a classe de métodos empíricos, avaliada nesta Seção, possui dois representantes no projeto PESA (GC e GMA) uma avaliação comparativa entre eles também foi efetuada. Os resultados dessa comparação são apresentados na Tabela 17.

Tabela 17: Análise comparativa dos métodos GC e GMA.

	Corpus Alinhado pelo GC		Corpus Alinhado pelo GMA	
Alinhamentos Propostos	**Autêntico**	**Pré-editado**	**Autêntico**	**Pré-editado**
Parcialmente Corretos	25 (6,25%)	6 (1,45%)	12 (2,94%)	3 (0,72%)
Totalmente Corretos	365 (91,25%)	405 (97,59%)	387 (94,85%)	413 (99,04%)
Errados	10 (2,5%)	4 (0,96%)	9 (2,21%)	1 (0,24%)
Total	**400**	**415**	**408**	**417**

Os valores da Tabela 17 evidenciam que o método GMA obteve um desempenho melhor do que o método GC, porém a diferença entre os valores relatados não é suficientemente grande para apontar um deles como o melhor.

Em termos computacionais, os recursos utilizados por ambos foram os mesmos e nenhuma dependência do tempo de processamento em relação ao tamanho dos textos sendo alinhados, que não a linear, foi verificada.

As próximas seções apresentam algumas considerações sobre os métodos empíricos GC e GMA e alguns exemplos de bitextos alinhados por ambos.

7.1.1 Considerações sobre o Método GC

Nesta Seção são apresentadas algumas considerações sobre o desempenho do método GC no alinhamento dos corpora de teste CAT e CPT. Para enriquecer essa análise, os resultados da primeira avaliação do método – descrita em (Gale & Church, 1991, 1993) – foram comparados com os da avaliação efetuada no projeto PESA. Essas duas avaliações serão referenciadas no restante desta Seção como **avaliação1** e **avaliação2**, respectivamente.

A precisão relatada na avaliação1 foi de 95,8% em um corpus composto por quinze relatórios econômicos do *Union Bank of Switzerland (UBS)* escritos em inglês, francês e alemão. Os textos em inglês totalizavam 14680 palavras, 725 sentenças e 188 parágrafos. O critério utilizado para se determinar a taxa de erro do método foi o de comparar os alinhamentos produzidos por ele com os gerados por juízes humanos considerando-se apenas duas possibilidades para os alinhamentos: corretos ou incorretos. Alinhamentos parcialmente corretos não foram considerados.

O programa cometeu 35 erros num total de 620 alinhamentos (5,6%) para o par inglês-Francês e 19 erros em 695 alinhamentos (2,7%) para o par inglês-Alemão. No total foram 54 erros em 1315 alinhamentos (4,2%). Além disso, observou-se que a taxa de erro se manteve constante para os pares inglês-francês e inglês-alemão, caracterizando-o como um método independente da língua, pelo menos nessa avaliação.

A taxa de erro por categoria de alinhamento também foi analisada e os resultados são mostrados na Tabela 18.

Tabela 18: Análise da taxa de erro por categoria (avaliação1) (Gale & Church, 1991, 1993).

	Inglês-Francês			**Inglês-Alemão**			**Total**		
Categoria	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
1-0 ou 0-1	8	8	100	5	5	100	13	13	100
1-1	542	14	2,6	625	9	1,4	1167	23	2,0
2-1 ou 1-2	59	8	14	58	2	3,4	117	10	9
2-2	9	3	33	6	2	33	15	5	33
3-1 ou 1-3	1	1	100	1	1	100	2	2	100
3-2 ou 2-3	1	1	100	0	0	0	1	1	100
Total	**620**	**35**		**695**	**19**		**1315**	**54**	

Com base na Tabela 18, constatou-se que os alinhamentos 1-1 foram os mais fáceis de serem alinhados, com uma taxa de erro de 2,0% no total dos alinhamentos produzidos; seguidos pelos alinhamentos 2-1 (ou 1-2), que apresentaram uma taxa de erro quatro vezes maior do que os primeiros (1-1). Os alinhamentos 2-2 apresentaram 33% de erro, mas a maioria foi alinhada corretamente. Os demais casos apresentaram 100% de erro e são eles: os alinhamentos 3-1 (ou 1-3) e 3-2 (ou 2-3) não considerados pelo método e, por isso, todos os três alinhamentos existentes foram incorretamente alinhados; e os alinhamentos 1-0 (ou 0-1). Essa última categoria de alinhamento foi apontada pelos autores do método como a que necessitava de mais estudos e possivelmente da utilização de informações específicas sobre as línguas envolvidas.

Enquanto que na avaliação1, o método GC apresentou uma precisão de 95,8%, na avaliação2, para o corpus CPT, apresentou uma precisão maior: 97,59%. Já para o CAT,

apresentou uma precisão menor: 91,25%, cuja diferença pode ser explicada pelo fato de os textos do CAT possuírem erros (ou ruídos) que não deveriam estar presentes nos relatórios econômicos que compunham o corpus utilizado na avaliação1.

Na avaliação2, o GC cometeu 40 erros num total de 405 alinhamentos (9,88%) para o CAT e 11 erros em 416 alinhamentos (2,64%) para o CPT. No total foram 51 erros em 821 alinhamentos (6,21%). Com relação à taxa de erro por categoria de alinhamento, notou-se que na avaliação2 o método GC apresentou os mesmos resultados relatados na avaliação1, como pode ser observado na Tabela 19.

Tabela 19: Análise da taxa de erro por categoria nos corpora alinhados pelo método GC (avaliação2).

	Corpus Autêntico			**Corpus Pré-editado**			**Total**		
Categoria	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
0-1 ou 1-0	6	6	100	2	2	100	8	8	100
1-1	353	23	6,52	395	6	1,52	748	29	3,88
2-1 ou 1-2	41	7	17,07	17	1	5,88	58	8	13,79
2-2	4	3	75	2	2	100	6	5	83,33
2-3	1	1	100	0	0	0	1	1	100
Total	**405**	**40**		**416**	**11**		**821**	**51**	

Com base nos valores da Tabela 19, pode-se perceber que a menor taxa de erro foi constatada nos alinhamentos 1-1 (3,88%); seguidos pelos alinhamentos 2-1 (ou 1-2), que apresentaram uma taxa de erro três vezes maior do que os primeiros (1-1). Os alinhamentos 2-2 apresentaram uma taxa de erro de 83,33%, ou seja, a minoria foi alinhada corretamente. Os demais casos apresentaram 100% de erro e são eles: um único caso de alinhamento 2-3 que por ser uma categoria não considerada pelo método foi incorretamente alinhada; e oito casos de omissão (0-1 ou 1-0). A taxa de erro de 100% nos casos de omissão comprova o que já havia sido relatado na avaliação1: o tratamento destes casos, possivelmente, requer a utilização de informações específicas sobre as línguas envolvidas.

A seguir são apresentados alguns exemplos de bitextos alinhados pelo método GC e os respectivos alinhamentos de referência correspondentes.

Exemplo 7.1.1-1: Um alinhamento 1-2 considerado como um alinhamento 1-1 seguido de um alinhamento 1-2.

art10R-art10A do CAR

<s id=**art10R.1.s1** corresp='**art10A.1.s1 art10A.1.s2**'>O SPP2 (Servidor de Processamento Paralelo), desenvolvido no Laboratório de Computação de Alto Desempenho (LCAD-ICMC-USP) utiliza computadores convencionais conectados por uma rede de comunicação de alta velocidade.</s>	<s id=**art10A.1.s1** corresp=**art10R.1.s1**>Conventional computers connected by high-speed communication networks present a very low cost alternative to the MPPs (Massively Parallel Processors) for applications that demand high computing power.</s><s id=**art10A.1.s2** corresp=**art10R.1.s1**>The SPP2 (Parallel Processing Server), developed at the LCAD-ICMC-USP, is one of these systems.</s>

art10R-art10A do CAT após ser alinhado pelo método GC

<s id=**art10R.1.s1** corresp=**art10A.1.s1**>O SPP2 (Servidor de Processamento Paralelo), desenvolvido no Laboratório de Computação de Alto Desempenho (LCAD-ICMC-USP) utiliza computadores convencionais conectados por uma rede de comunicação de alta velocidade.</s><s id=**art10R.1.s2** corresp='**art10A.1.s2 art10A.1.s3**'>Pesquisadores da Universidade de Illinois desenvolveram uma camada de software de alto desempenho para a troca de mensagens entre máquinas conectadas por redes de alta velocidade Myrinet denominada Fast Messages, e que apresenta baixa latência na transmissão de mensagens e alta taxa de transferência.</s>	<s id=**art10A.1.s1** corresp=**art10R.1.s1**>Conventional computers connected by high-speed communication networks present a very low cost alternative to the MPPs (Massively Parallel Processors) for applications that demand high computing power.</s><s id=**art10A.1.s2** corresp=**art10R.1.s2**>The SPP2 (Parallel Processing Server), developed at the LCAD-ICMC-USP, is one of these systems.</s><s id=**art10A.1.s3** corresp=**art10R.1.s2**>The Fast Messages is a high-performance communication system developed at University of Illinois that can be used to build more complex message passing systems.</s>

Exemplo 7.1.1-2: Um alinhamento 1-0 seguido de um alinhamento 1-1 considerado como um alinhamento 2-1.

es7R-es7A do CAR

<s id=**es7R.1.s3** corresp=**''**>Dessa forma, quando diante da manutenção do produto, o engenheiro de software encontra uma documentação informal e incompleta, que não reflete o software existente.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse contexto é que se encontra a Engenharia Reversa de Software, com o propósito de recuperar as informações de projeto perdidas durante a fase de desenvolvimento, e de documentar o real estado do software.</s>	<s id=**es7A.1.s3** corresp=**es7R.1.s4**>In this context Reverse Engineering of Software can help by means of recovering the project information lost during the development phase and documenting the current software state.</s>

es7R-es7A do CAT após ser alinhado pelo método GC

<s id=**es7R.1.s3** corresp=**es7A.1.s3**>Dessa forma, quando diante da manutenção do produto, o engenheiro de software encontra uma documentação informal e incompleta, que não reflete o software existente.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse contexto é que se encontra a Engenharia Reversa de Software, com o propósito de recuperar as informações de projeto perdidas durante a fase de desenvolvimento, e de documentar o real estado do software.</s>	<s id=**es7A.1.s3** corresp='**es7R.1.s3 es7R.1.s4**'>In this context Reverse Engineering of Software can help by means of recovering the project information lost during the development phase and documenting the current software state.</s>

Exemplo 7.1.1-3: Um alinhamento 2-2 considerado como dois alinhamentos 1-1.

es12R-es12A do CAR

<s id=**es12R.3.s1** corresp='**es12A.3.s1 es12A.3.s2**'>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp='**es12A.3.s1 es12A.3.s2**'>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp='**es12R.3.s1 es12R.3.s2**'>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp='**es12R.3.s1 es12R.3.s2**'>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

es12R-es12A do CAT após ser alinhado pelo método GC

<s id=**es12R.3.s1** corresp=**es12A.3.s1**>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp=**es12A.3.s2**>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp=**es12R.3.s1**>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp=**es12R.3.s2**>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

Exemplo 7.1.1-4: Um alinhamento 1-2 considerado como dois alinhamentos 1-1.

art8R-art8A do CAR

<s id=**art8R.1.s6** corresp='**art8A.1.s6 art8A.1.s7**'>O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total.</s>	<s id=**art8A.1.s6** corresp=**art8R.1.s6**>The Clustered Knapsack Problem consists of determining the suitable capacities of each cluster and how these clusters should be filled.</s><s id=**art8A.1.s7** corresp=**art8R.1.s6**>The objective is to maximize a total utility value.</s>

art8R-art8A do CAT após ser alinhado pelo método GC

<s id=**art8R.1.s6** corresp=**art8A.1.s6**>O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total.</s><s id=**art8R.1.s7** corresp=**art8A.1.s7**>Nesse trabalho, propomos uma modelagem matemática não linear inteira para o problema e verificamos algumas heurísticas para sua resolução.</s>	<s id=**art8A.1.s6** corresp=**art8R.1.s6**>The Clustered Knapsack Problem consists of determining the suitable capacities of each cluster and how these clusters should be filled.</s><s id=**art8A.1.s7** corresp=**art8R.1.s7**>The objective is to maximize a total utility value.</s>

Exemplo 7.1.1-5: Um alinhamento 1-1 seguido de um alinhamento 1-0 considerado como um alinhamento 2-1.

bd1R-bd1A do CAR

<s id=**bd1R.1.s4** corresp=**bd1A.1.s3**>Por exemplo, se duas organizações devem trocar dados sobre pessoas, não importa se para as diferentes organizações as pessoas são clientes, empregados, alunos ou pacientes, o significado de "pessoa" é sempre entendido pelos membros das organizações.</s><s id=**bd1R.1.s5** corresp=**''**>O mesmo ocorre com qualquer entidade que se deseje trocar informações.</s>	<s id=**bd1A.1.s3** corresp=**bd1R.1.s4**>For example, if two organizations should interchange data on people, it does not care, for the different organizations, if the people are customers, employees, students or patient, the means of what are "people" is always understood by each organization.</s>

bd1R-bd1A do CAT após ser alinhado pelo método GC

<s id=**bd1R.1.s4** corresp=**bd1A.1.s3**>Por exemplo, se duas organizações devem trocar dados sobre pessoas, não importa se para as diferentes organizações as pessoas são clientes, empregados, alunos ou pacientes, o significado de "pessoa" é sempre entendido pelos membros das organizações.</s><s id=**bd1R.1.s5** corresp=**bd1A.1.s3**>O mesmo ocorre com qualquer entidade que se deseje trocar informações.</s>	<s id=**bd1A.1.s3** corresp='**bd1R.1.s4 bd1R.1.s5**'>For example, if two organizations should interchange data on people, it does not care, for the different organizations, if the people are customers, employees, students or patient, the means of what are "people" is always understood by each organization.</s>

7.1.2 Considerações sobre o Método GMA

De maneira semelhante à Seção anterior (7.1.1), nesta Seção são apresentadas algumas considerações sobre o desempenho do método GMA no alinhamento dos corpora de teste CAT e CPT. As denominações **avaliação1** e **avaliação2** são usadas, respectivamente, para: a avaliação descrita em (Melamed, 2000) e a avaliação efetuada no projeto PESA.

Na avaliação1, a precisão relatada ficou entre 97,7% e 98,4% de acordo com a "dificuldade" de alinhamento dos textos no corpus utilizado para teste: textos de debates do

Parlamento Canadense extraídos do *Canadian Hansard Corpus*. Esses textos paralelos, escritos nas línguas oficiais do país (inglês e francês), foram divididos em dois conjuntos de acordo com a "dificuldade" de alinhamento: o "easy" Hansard, com 7123 alinhamentos; e o "hard" Hansard, com 2693 alinhamentos. A avaliação foi feita com base em um alinhamento de referência e, assim como a avaliação1 do método GC citada na Seção anterior (7.1.1), apenas duas possibilidades foram consideradas para os alinhamentos: corretos ou incorretos. Além disso, o método GMA foi comparado com outros dois métodos de alinhamento sentencial de textos paralelos, entre eles o GC. Os valores da avaliação1 para os métodos GC e GMA são mostrados na Tabela 20.

Tabela 20: Comparação da taxa de erro dos métodos GC e GMA (avaliação1) (Melamed, 2000).

	"easy" Hansard			**"hard" Hansard**			**Total**		
Algoritmo	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
GC	7123	128	1,8	2693	80	3,0	9816	208	2,12
GMA	7123	115	1,6	2693	61	2,3	9816	176	1,79

Os resultados descritos na avaliação1 se aproximam dos relatados na avaliação desempenhada no projeto PESA na qual o método GC apresentou 6,21% de erro para os corpora CAT e CPT (vide Seção 7.1.1) e o GMA obteve uma taxa quase três vezes menor (2,56%) para os mesmos corpora, como pode ser observado na Tabela 21.

Tabela 21: Comparação da taxa de erro dos métodos GC e GMA para os corpora CAT e CPT (avaliação2).

	Corpus Autêntico			**Corpus Pré-editado**			**Total**		
Algoritmo	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
GC	405	40	9,88	416	11	2,64	821	51	6,21
GMA	405	18	4,44	416	3	0,72	821	21	2,56

A precisão relatada na avaliação1 – 98,21% para o GMA e 97,88% para o GC – foi um pouco maior do que a relatada na avaliação2 – 97,44% para o GMA e 93,79% para o GC –, mas o GMA apresentou melhor desempenho em ambas. Esta diferença entre os valores das duas avaliações pode estar relacionada ao fato de textos com ruído (do CAT) terem sido utilizados na avaliação2, o que afeta o desempenho dos métodos de alinhamento sentencial de textos paralelos segundo (Gaussier et al., 2000).

A avaliação1 não verificou a taxa de erro por categoria de alinhamento para o método GMA, por isso apenas os valores da avaliação2 são apresentados na tabela 22.

Tabela 22: Análise da taxa de erro por categoria de alinhamentos dos corpora alinhados pelo método GMA (avaliação2).

	Corpus Autêntico			Corpus Pré-editado			Total		
Categoria	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
0-1 ou 1-0	6	5	83,33	2	1	50	8	6	75
1-1	353	6	1,70	395	0	0	748	6	0,80
2-1 ou 1-2	41	3	7,32	17	0	0	58	3	5,17
2-2	4	3	75	2	2	100	6	5	83,33
2-3	1	1	100	0	0	0	1	1	100
Total	**405**	**18**		**416**	**3**		**821**	**21**	

Com base nos valores da Tabela 22, pode-se perceber que a menor taxa de erro foi constatada nos alinhamentos 1-1 (0,80% no total); seguidos pelos alinhamentos 2-1 (ou 1-2), que apresentaram uma taxa de erro (5,17%) quase sete vezes maior do que os primeiros (1-1). Os alinhamentos 2-2 apresentaram a mesma taxa de erro que o método GC (83,33%) caracterizando que a minoria foi alinhada corretamente. O único caso de alinhamento 2-3, como na avaliação2 do GC, também apresentou taxa de erro de 100% no método GMA. Já 25% dos casos de omissão (0-1 e 1-0), que foram completamente perdidos no método GC, foram alinhados corretamente pelo método GMA. Este fato comprova o que havia sido suposto nas avaliações (1 e 2) do método GC (vide Seção 7.1.1): a existência de informação específica a respeito das línguas envolvidas (cognatos, neste caso) melhora o desempenho do método de alinhamento sentencial nos casos de omissão.

A seguir são apresentados os mesmos bitextos da Seção anterior (7.1.1) alinhados pelo método GMA. Note que o GMA apresentou o mesmo resultado que o GC no Exemplo 7.1.1-3; alinhou corretamente as sentenças dos exemplos 7.1.1-4 e 7.1.1-5; e não obteve sucesso nos outros dois casos (Exemplo 7.1.1-1 e Exemplo 7.1.1-2) assim como o GC, porém os alinhamentos gerados por ele diferem dos alinhamentos gerados pelo GC.

Exemplo 7.1.2-1: Um alinhamento 1-2 considerado como um alinhamento 0-1 e um 1-1.

art10R-art10A do CAR

<s id=**art10R.1.s1** corresp='**art10A.1.s1 art10A.1.s2**'>O SPP2 (Servidor de Processamento Paralelo), desenvolvido no Laboratório de Computação de Alto Desempenho (LCAD-ICMC-USP) utiliza computadores convencionais conectados por uma rede de comunicação de alta velocidade.</s>	<s id=**art10A.1.s1** corresp=**art10R.1.s1**>Conventional computers connected by high-speed communication networks present a very low cost alternative to the MPPs (Massively Parallel Processors) for applications that demand high computing power.</s><s id=**art10A.1.s2** corresp=**art10R.1.s1**>The SPP2 (Parallel Processing Server), developed at the LCAD-ICMC-USP, is one of these systems.</s>

art10R-art10A do CAT após ser alinhado pelo método GMA

<s id=**art10R.1.s1** corresp=**art10A.1.s2**>O SPP2 (Servidor de Processamento Paralelo), desenvolvido no Laboratório de Computação de Alto Desempenho (LCAD-ICMC-USP) utiliza computadores convencionais conectados por uma rede de comunicação de alta velocidade.</s>	<s id=**art10A.1.s1** corresp=**''**>Conventional computers connected by high-speed communication networks present a very low cost alternative to the MPPs (Massively Parallel Processors) for applications that demand high computing power.</s><s id=**art10A.1.s2** corresp=**art10R.1.s1**>The SPP2 (Parallel Processing Server), developed at the LCAD-ICMC-USP, is one of these systems.</s>

Exemplo 7.1.2-2: Um alinhamento 1-0 seguido de um alinhamento 1-1 considerado como dois alinhamentos 1-1.

es7R-es7A do CAR

<s id=**es7R.1.s3** corresp=**''**>Dessa forma, quando diante da manutenção do produto, o engenheiro de software encontra uma documentação informal e incompleta, que não reflete o software existente.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse contexto é que se encontra a Engenharia Reversa de Software, com o propósito de recuperar as informações de projeto perdidas durante a fase de desenvolvimento, e de documentar o real estado do software.</s>	<s id=**es7A.1.s3** corresp=**es7R.1.s4**>In this context Reverse Engineering of Software can help by means of recovering the project information lost during the development phase and documenting the current software state.</s>

es7R-es7A do CAT após ser alinhado pelo método GMA

<s id=**es7R.1.s3** corresp=**es7A.1.s2**>Dessa forma, quando diante da manutenção do produto, o engenheiro de software encontra uma documentação informal e incompleta, que não reflete o software existente.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse contexto é que se encontra a Engenharia Reversa de Software, com o propósito de recuperar as informações de projeto perdidas durante a fase de desenvolvimento, e de documentar o real estado do software.</s>	<s id=**es7A.1.s2** corresp=**es7R.1.s3**>The maintenance of such software is problematic, since its documentation rarely reflects the implemented code.</s><s id=**es7A.1.s3** corresp=**es7R.1.s4**>In this context Reverse Engineering of Software can help by means of recovering the project information lost during the development phase and documenting the current software state.</s>

Exemplo 7.1.2-3: Um alinhamento 2-2 considerado como dois alinhamentos 1-1.

es12R-es12A do CAR

<s id=**es12R.3.s1** corresp='**es12A.3.s1 es12A.3.s2**'>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp='**es12A.3.s1 es12A.3.s2**'>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp='**es12R.3.s1 es12R.3.s2**'>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp='**es12R.3.s1 es12R.3.s2**'>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

es12R-es12A do CAT após ser alinhado pelo método GMA.

<s id=**es12R.3.s1** corresp=**es12A.3.s1**>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp=**es12A.3.s2**>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp=**es12R.3.s1**>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp=**es12R.3.s2**>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

Exemplo 7.1.2-4: Um alinhamento 1-2 alinhado corretamente pelo método GMA.

art8R-art8A do CAT após ser alinhado pelo método GMA

<s id=**art8R.1.s6** corresp='**art8A.1.s6 art8A.1.s7**'>O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total.</s>	<s id=**art8A.1.s6** corresp=**art8R.1.s6**>The Clustered Knapsack Problem consists of determining the suitable capacities of each cluster and how these clusters should be filled.</s><s id=**art8A.1.s7** corresp=**art8R.1.s6**>The objective is to maximize a total utility value.</s>

Exemplo 7.1.2-5: Um alinhamento 1-1 seguido de um alinhamento 1-0 alinhado corretamente pelo método GMA.

bd1R-bd1A do CAT após ser alinhado pelo método GMA

<s id=**bd1R.1.s4** corresp=**bd1A.1.s3**>Por exemplo, se duas organizações devem trocar dados sobre pessoas, não importa se para as diferentes organizações as pessoas são clientes, empregados, alunos ou pacientes, o significado de "pessoa" é sempre entendido pelos membros das organizações.</s><s id=**bd1R.1.s5** corresp=''>O mesmo ocorre com qualquer entidade que se deseje trocar informações.</s>	<s id=**bd1A.1.s3** corresp=**bd1R.1.s4**>For example, if two organizations should interchange data on people, it does not care, for the different organizations, if the people are customers, employees, students or patient, the means of what are "people" is always understood by each organization.</s>

7.2 Avaliação do Método Lingüístico

O método lingüístico analisado no projeto PESA foi implementado e avaliado segundo os critérios citados no início deste capítulo e os resultados são relatados nesta Seção.

Após serem alinhados pelo método lingüístico, os corpora CATE e CPTE, foram comparados com os corpora de referência CAR e CPR, respectivamente, resultando nos valores para *precision, recall* e *F-measure* apresentados na Tabela 23.

Tabela 23: Métricas calculadas para os corpora alinhados pelo método lingüístico.

Métricas	CAT	CPT
precision	0,8589	0,9784
recall	0,8716	0,9784
F	0,8652	0,9784

Os valores da Tabela 23, assim como para os métodos empíricos, comprovam o que já havia sido relatado na literatura: o alinhamento sentencial é mais eficaz em textos limpos (sem erros gramaticais ou de tradução) (Gaussier et al., 2000).

A análise das categorias de alinhamentos mais freqüentes nos corpora alinhados pelo método lingüístico comparadas às categorias dos corpora de referência apresentou os resultados mostrados na Tabela 24. A análise da taxa de erro do método nestas categorias considerando-se também os alinhamentos parcialmente corretos é apresentada na Tabela 25.

Tabela 24: Análise das categorias de alinhamentos dos corpora alinhados pelo método lingüístico.

	Corpus de Referência		**Corpus Alinhado pelo método lingüístico**	
Categoria	**Autêntico**	**Pré-editado**	**Autêntico**	**Pré-editado**
0-1 ou 1-0	6	2	0	0
1-1	353	395	377	397
2-1 ou 1-2	41	17	34	19
2-2	4	2	0	0
2-3 ou 3-2	1	-	0	0
Total	**405**	**416**	**411**	**416**

Tabela 25: Análise da taxa de erro do método lingüístico.

	CATE			**CPTE**		
Categoria	**Parcialmente**	**Corretos**	**Errados**	**Parcialmente**	**Corretos**	**Errados**
0-1	0	0	0	0	0	0
1-0	0	0	0	0	0	0
1-1	21	328	28	3	391	3
1-2	7	20	1	2	14	0
2-1	1	5	0	1	2	0
2-2	0	0	0	0	0	0
Total	**29**	**353**	**29**	**6**	**407**	**3**

Em termos computacionais, os recursos utilizados pelo método lingüístico foram os mesmos dos métodos empíricos e, também, nenhuma dependência do tempo de processamento em relação ao tamanho dos textos sendo alinhados, que não a linear, foi verificada.

A próxima Seção apresenta algumas considerações sobre o método lingüístico e alguns exemplos de bitextos alinhados por ele.

7.2.1 Considerações sobre o Método Lingüístico

Nesta Seção são apresentadas algumas considerações sobre o desempenho do método ligüístico no alinhamento dos corpora de teste CATE e CPTE. Para enriquecer essa análise, os resultados da avaliação do método descrita em (Piperidis et al., 2000) foram comparados com os da avaliação efetuada no projeto PESA. Essas duas avaliações serão referenciadas no restante desta Seção como **avaliação1** e **avaliação2**, respectivamente.

A precisão relatada na avaliação1 foi de 99% em um corpus composto por cerca de 3000 sentenças do corpus CELEX escritas em grego e inglês. O corpus CELEX é o sistema de documentação computadorizada da *European Community Law*[30], composto por regulamentos, artigos, recomendações, etc. O critério utilizado na avaliação1 limitou-se a classificar os alinhamentos em corretos ou incorretos. Alinhamentos parcialmente corretos não foram considerados.

O método cometeu 5 erros num total de 3219 alinhamentos (0,16%) como pode ser observado na Tabela 26. Com base nesta tabela, constatou-se que os alinhamentos mais fáceis foram os das categorias 1-1 e 2-2; seguidos pelos alinhamentos 2-1 (ou 1-2) que apresentaram uma taxa de erro de 8,33%. Os casos mais difíceis, assim como para os métodos empíricos (vide seções 7.1.1 e 7.1.2) foram os de omissão (1-0 ou 0-1), com 40% de erro. Porém, a maioria dos alinhamentos desta categoria foi alinhada corretamente.

Tabela 26: Análise da taxa de erro por categoria no corpus alinhado pelo método lingüístico (avaliação1) (Piperidis et al., 2000).

	Sentenças do corpus CELEX		
Categoria	**Total**	**Errados**	**%**
1-0 ou 0-1	5	2	40
1-1	3178	0	0
2-1 ou 1-2	36	3	8,33
2-2	0	0	0
Total	**3219**	**5**	

Na avaliação2, o método lingüístico apresentou precisões menores do que a relatada na avaliação1: 87,16% para o CAT e 97,84% para o CPT. Esses valores menores podem ser conseqüência da utilização do etiquetador TreeTagger sem um treinamento prévio com textos de mesmo domínio. Além disso, este método foi o que apresentou a maior diferença entre as precisões do CPT e CAT indicando que nessa classe de métodos, os corpora com ruídos (CAT) são muito mais prejudiciais para o processo de alinhamento do que nos demais

[30] Página: http://europa.eu.int/celex/htm/celex_en.htm (17/02/2003).

métodos, pelo menos neste experimento. Por fim, da mesma forma que para os métodos empíricos, a baixa precisão relatada para o CAT pode ser explicada pelo fato de os textos do CAT possuírem erros que não deveriam estar presentes nos textos que compunham o corpus utilizado na avaliação1.

Na avaliação2, o método lingüístico cometeu 52 erros num total de 405 alinhamentos (12,84%) para o CAT e 9 erros em 416 alinhamentos (2,16%) para o CPT. No total foram 61 erros em 821 alinhamentos (7,43%). Com relação à taxa de erro por categoria de alinhamento, notaram-se algumas diferenças em relação à avaliação2, como pode ser observado na Tabela 27.

Tabela 27: Análise da taxa de erro por categoria nos corpora alinhados pelo método lingüístico (avaliação2).

	CATE			**CPTE**			**Total**		
Categoria	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
0-1 ou 1-0	6	6	100	2	2	100	8	8	100
1-1	353	25	7,08	395	4	1,01	748	29	3,88
2-1 ou 1-2	41	16	39,02	17	1	5,88	58	17	29,31
2-2	4	4	100	2	2	100	6	6	100
2-3	1	1	100	0	0	0	1	1	100
Total	**405**	**52**		**416**	**9**		**821**	**61**	

Com base nos valores da Tabela 27, pode-se perceber que a menor taxa de erro foi constatada nos alinhamentos 1-1 (3,88%); seguidos pelos alinhamentos 2-1 (ou 1-2), que apresentaram uma taxa de erro sete vezes maior do que os primeiros (1-1). Os demais casos apresentaram 100% de erro e são eles: seis alinhamentos 2-2, um único alinhamento 2-3 e oito casos de omissão (0-1 ou 1-0).

A seguir são apresentados alguns exemplos de bitextos alinhados pelo método lingüístico, nos quais o mesmo resultado que o GC foi obtido no Exemplo 7.2.1-1, Exemplo 7.2.1-3 e Exemplo 7.2.1-4; e mesmo resultado que o GMA no Exemplo 7.2.1-2. O Exemplo 7.2.1-5 não foi alinhado corretamente pelo método lingüístico, nem da mesma forma que pelos outros dois métodos.

Exemplo 7.2.1-1: Um alinhamento 1-2 considerado como um alinhamento 1-1 seguido de um alinhamento 1-2.

art10R-ar10A do CAR

<s id=**art10R.1.s1** corresp='**art10A.1.s1 art10A.1.s2**'>O SPP2 (Servidor de Processamento Paralelo), desenvolvido no Laboratório de Computação de Alto Desempenho (LCAD-ICMC-USP) utiliza computadores convencionais conectados por uma rede de comunicação de alta velocidade.</s>	<s id=**art10A.1.s1** corresp=**art10R.1.s1**>Conventional computers connected by high-speed communication networks present a very low cost alternative to the MPPs (Massively Parallel Processors) for applications that demand high computing power.</s><s id=**art10A.1.s2** corresp=**art10R.1.s1**>The SPP2 (Parallel Processing Server), developed at the LCAD-ICMC-USP, is one of these systems.</s>

art10R-art10A do CATE após ser alinhado pelo método lingüístico

<s id=**art10R.1.s1** corresp=**art10A.1.s1**>O ART SPP2 NP (Servidor N de PREP Processamento N Paralelo ADJ), desenvolvido VERB no PREP+ART Laboratório N de PREP Computação NP de PREP Alto ADJ Desempenho N (LCAD-ICMC-USP NP) utiliza VERB computadores N convencionais ADJ conectados VERB por PREP uma ART rede N de PREP comunicação N de PREP alta ADJ velocidade N.</s><s id=**art10R.1.s2** corresp='**art10A.1.s2 art10A.1.s3**'>Pesquisadores N da PREP+ART Universidade NP de PREP Illinois VERB desenvolveram VERB uma ART camada N de PREP software N de PREP alto ADJ desempenho N para PREP a ART troca N de PREP mensagens N entre PREP máquinas N conectadas VERB por PREP redes N de PREP alta ADJ velocidade N Myrinet NP denominada VERB Fast NP Messages NP, e CONJ que PRON apresenta VERB baixa ADJ latência N na PREP+ART transmissão N de PREP mensagens N e CONJ alta ADJ taxa N de PREP transferência N.</s>	<s id=**art10A.1.s1** corresp=**art10R.1.s1**>Conventional JJ computers NNS connected VBN by IN high-speed JJ communication NN networks NNS present VBP a DT very RB low JJ cost NN alternative NN to TO the DT MPPs NP (Massively RB Parallel JJ Processors NPS) for IN applications NNS that WDT demand VBP high JJ computing NN power NN.</s><s id=**art10A.1.s2** corresp=**art10R.1.s2**>The DT SPP2 NP (Parallel JJ Processing NP Server NN), developed VBN at IN the DT LCAD-ICMC-USP NP, is VBZ one CD of IN these DT systems NNS.</s><s id=**art10A.1.s3** corresp=**art10R.1.s2**>The DT Fast NP Messages NNS is VBZ a DT high-performance JJ communication NN system NN developed VBN at IN University NP of IN Illinois NP that WDT can MD be VB used VBN to TO build VB more RBR complex JJ message NN passing VBG systems NNS.</s>

Exemplo 7.2.1-2: Um alinhamento 1-0 seguido de um alinhamento 1-1 considerado como dois alinhamentos 1-1.

es7R-es7A do CAR

<s id=**es7R.1.s3** corresp=**''**>Dessa forma, quando diante da manutenção do produto, o engenheiro de software encontra uma documentação informal e incompleta, que não reflete o software existente.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse contexto é que se encontra a Engenharia Reversa de Software, com o propósito de recuperar as informações de projeto perdidas durante a fase de desenvolvimento, e de documentar o real estado do software.</s>	<s id=**es7A.1.s3** corresp=**es7R.1.s4**>In this context Reverse Engineering of Software can help by means of recovering the project information lost during the development phase and documenting the current software state.</s>

es7R-es7A do CAT após ser alinhado pelo método lingüístico

<s id=**es7R.1.s3** corresp=**es7A.1.s2**>Dessa PREP+PD forma N, quando LOCU diante LOCU da PREP+ART manutenção N do PREP+ART produto N, o ART engenheiro N de PREP software N encontra VERB uma ART documentação N informal ADJ e CONJ incompleta ADJ, que PRON não ADV reflete VERB o ART software N existente ADJ.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse PREP+PD contexto N é VERB que CONJ se PRON encontra VERB a ART Engenharia N Reversa ADJ de PREP Software N, com PREP o ART propósito N de PREP recuperar VERB as ART informações N de PREP projeto N perdidas ADJ durante PREP a ART fase N de PREP desenvolvimento N, e CONJ de PREP documentar VERB o ART real ADJ estado N do PREP+ART software N.</s>	<s id=**es7A.1.s2** corresp=**es7R.1.s3**>The DT maintenance NN of IN such JJ software NN is VBZ problematic JJ, since IN its PP$ documentation NN rarely RB reflects VBZ the DT implemented VBN code NN.</s><s id=**es7A.1.s3** corresp=**es7R.1.s4**>In IN this DT context NN Reverse VBP Engineering NP of IN Software NP can MD help VB by IN means NNS of IN recovering VBG the DT project NN information NN lost VBN during IN the DT development NN phase NN and CC documenting VBG the DT current JJ software NN state NN.</s>

Exemplo 7.2.1-3: Um alinhamento 2-2 considerado como dois alinhamentos 1-1.

es12R-es12A do CAR

<s id=**es12R.3.s1** corresp='**es12A.3.s1 es12A.3.s2**'>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp='**es12A.3.s1 es12A.3.s2**'>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp='**es12R.3.s1 es12R.3.s2**'>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp='**es12R.3.s1 es12R.3.s2**'>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

es12R-es12A do CAT após ser alinhado pelo método lingüístico

<s id=**es12R.3.s1** corresp=**es12A.3.s1**>Dessa PREP+PD forma N, neste PREP+PD trabalho N é VERB apresentada ADJ uma ART ferramenta VERB de PREP injeção N de PREP defeitos N de PREP software N, denominada VERB ITool N, baseada VERB em PREP um ART esquema N de PREP injeção N de PREP defeitos N.</s><s id=**es12R.3.s2** corresp=**es12A.3.s2**>Esse PRON esquema N caracteriza VERB o ART mapeamento N de PREP uma ART taxonomia N de PREP defeitos N de PREP software N (Taxonomia N de PREP DeMillo N) para PREP os ART operadores N de PREP mutação N do PREP+ART critério N de PREP teste N Análise N de PREP Mutantes N para PREP a ART linguagem N C RES.</s>	<s id=**es12A.3.s1** corresp=**es12R.3.s1**>In IN this DT perspective NN, in IN this DT work NN a DT software NN fault NN injection NN tool NN, named VBN ITool NP, is VBZ presented VBN.</s><s id=**es12A.3.s2** corresp=**es12R.3.s2**>This DT tool NN is VBZ based VBN on IN a DT fault NN injection NN scheme NN that WDT defines VBZ the DT mapping NN of IN a DT software NN fault NN taxonomy NN (DeMillo NP 's POS Taxonomy NN) to TO the DT mutation NN operators NNS of IN the DT Mutation NN Analysis NN criterion NN for IN C NP language NN.</s>

Exemplo 7.2.1-4: Um alinhamento 1-2 considerado como dois alinhamentos 1-1.

art8R-art8A do CAR

<s id=**art8R.1.s6** corresp='**art8A.1.s6 art8A.1.s7**'>O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total.</s>	<s id=**art8A.1.s6** corresp=**art8R.1.s6**>The Clustered Knapsack Problem consists of determining the suitable capacities of each cluster and how these clusters should be filled.</s><s id=**art8A.1.s7** corresp=**art8R.1.s6**>The objective is to maximize a total utility value.</s>

art8R-art8A do CAT após ser alinhado pelo método lingüístico

<s id=**art8R.1.s6** corresp=**art8A.1.s6**>O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total.</s><s id=**art8R.1.s7** corresp=**art8A.1.s7**>Nesse trabalho, propomos uma modelagem matemática não linear inteira para o problema e verificamos algumas heurísticas para sua resolução.</s>	<s id=**art8A.1.s6** corresp=**art8R.1.s6**>The Clustered Knapsack Problem consists of determining the suitable capacities of each cluster and how these clusters should be filled.</s><s id=**art8A.1.s7** corresp=**art8R.1.s7**>The objective is to maximize a total utility value.</s>

Exemplo 7.2.1-5: Um alinhamento 1-1 seguido de um alinhamento 1-0 considerado como dois alinhamentos 1-1.

bd1R-bd1A do CAR

<s id=**bd1R.1.s4** corresp=**bd1A.1.s3**>Por exemplo, se duas organizações devem trocar dados sobre pessoas, não importa se para as diferentes organizações as pessoas são clientes, empregados, alunos ou pacientes, o significado de "pessoa" é sempre entendido pelos membros das organizações.</s><s id=**bd1R.1.s5** corresp=**''**>O mesmo ocorre com qualquer entidade que se deseje trocar informações.</s>	<s id=**bd1A.1.s3** corresp=**bd1R.1.s4**>For example, if two organizations should interchange data on people, it does not care, for the different organizations, if the people are customers, employees, students or patient, the means of what are "people" is always understood by each organization.</s>

bd1R-bd1A do CAT após ser alinhado pelo método lingüístico

<s id=bd1R.1.s4 corresp=bd1A.2.s1>Por LOCU exemplo LOCU, se PRON duas NUME organizações N devem VERB trocar VERB dados N sobre PREP pessoas N, não ADV importa VERB se PRON para PREP as ART diferentes ADJ organizações N as ART pessoas N são VERB clientes N, empregados N, alunos N ou CONJ pacientes N, o ART significado N de PREP "pessoa N "é VERB sempre ADV entendido VERB pelos PREP+ART membros N das PREP+ART organizações N.</s><s id=bd1R.1.s5 corresp=bd1A.2.s2>O ART mesmo ADJ ocorre VERB com PREP qualquer ADJ entidade N que PRON se PRON deseje VERB trocar VERB informações N.</s>	<s id=bd1A.2.s1 corresp=bd1R.1.s4>This DT work VB states NNS that IN some DT form NN of IN primitive JJ, common JJ definition NN can MD exist VB for IN the DT data NN elements NNS that WDT must MD be VB shared VBN, from IN which WDT many JJ elements NNS of IN a DT database NN schema NN should MD be VB recognized VBN.</s><s id=bd1A.2.s2 corresp=bd1R.1.s5>Thus RB, it PP searches VBZ for IN primitive JJ structures NNS that WDT should MD be VB used VBN by IN the DT several JJ systems NNS with IN the DT purpose NN of IN integrating VBG them PP.</s>

7.3 Avaliação dos Métodos Híbridos

Os métodos híbridos implementados no projeto PESA – GSA+ e TCA – foram avaliados segundo os critérios citados no início deste capítulo e os resultados são apresentados nesta Seção.

O GSA+ foi testado com e sem a utilização de *stoplists* (vide Seção 6.1) para o PB e o inglês. Uma *stoplist* é composta por palavras muito freqüentes em uma determinada língua e que, por este motivo, podem atrapalhar o alinhamento. A *stoplist* para o inglês é fornecida com o método GSA+, enquanto que a *stoplist* para o português foi gerada a partir da Diadorim[31] e contém basicamente preposições, artigos, conjunções e pronomes. O melhor resultado do GSA+ foi obtido com a utilização das *stoplists* para as duas línguas como demonstram os valores de *precision*, *recall* e *F-measure* mostrados na Tabela 28.

Tabela 28: Métricas calculadas para os corpora alinhados pelo GSA+ com e sem as *stoplists* para o PB e o inglês.

	Com *Stoplists*		**Sem *Stoplists***	
Métricas	**CAT**	**CPT**	**CAT**	**CPT**
precision	0,9507	0,9904	0,9458	0,8541
recall	0,9531	0,9928	0,9481	0,8582
F	0,9519	0,9916	0,9470	0,8561

Os valores da Tabela 28 evidenciam mais uma vez o fato de o alinhamento sentencial ser mais eficaz em textos limpos (sem erros gramaticais ou de tradução) (Gaussier et al., 2000).

Um teste semelhante ao do GSA+ com as *stoplists* foi feito para o TCA e, da mesma forma que o primeiro, os melhores resultados foram obtidos quando palavras muito freqüentes não estavam presentes na LPA. Os valores das métricas calculados para o TCA são mostrados na Tabela 29.

Tabela 29: Métricas calculadas para os corpora alinhados pelo TCA com e sem as *stoplists* para o PB e o inglês.

	Com *Stoplists*		**Sem *Stoplists***	
Métricas	**CAT**	**CPT**	**CAT**	**CPT**
precision	0,9017	0,9420	0,8953	0,8993
recall	0,9062	0,9375	0,8444	0,8798
F	0,9039	0,9398	0,8691	0,8894

[31] Base de dados lexicais em: http://www.nilc.icmc.usp.br/nilc/tools/intermed.htm (17/02/2003).

As métricas do método híbrido GSA+ também foram comparadas às do método empírico GMA, para se verificar o impacto da adição de um recurso lingüístico a um método empírico. Nesta comparação constatou-se que os valores foram praticamente os mesmos como mostra a Tabela 30. Este fato pode estar relacionado à irrelevância da LPA para os corpora de teste CAT e CPT.

Tabela 30: Métricas calculadas para os métodos GSA+ e GMA.

	GSA+		**GMA**	
Métricas	**CAT**	**CPT**	**CAT**	**CPT**
precision	0,9507	0,9904	0,9485	0,9904
recall	0,9531	0,9928	0,9556	0,9928
F	0,9519	0,9916	0,9520	0,9916

Outra análise efetuada com os textos alinhados pelos métodos híbridos visou identificar as categorias de alinhamento encontradas pelo método GSA+ - mostrada na Tabela 31 – e pelo método TCA – mostrada na Tabela 32.

Tabela 31: Análise das categorias de alinhamentos dos corpora alinhados pelo método GSA+.

	Corpus de Referência		**Corpus Alinhado pelo GSA+**	
Categoria	**Autêntico**	**Pré-editado**	**Autêntico**	**Pré-editado**
0-1 ou 1-0	6	2	2	1
1-1	353	395	358	398
2-1 ou 1-2	41	17	46	18
2-2	4	2	-	-
2-3 ou 3-2	1	-	-	-
Total	**405**	**416**	**406**	**417**

Tabela 32: Análise das categorias de alinhamentos dos corpora alinhados pelo método TCA.

	Corpus de Referência		**Corpus Alinhado pelo TCA**	
Categoria	**Autêntico**	**Pré-editado**	**Autêntico**	**Pré-editado**
0-1 ou 1-0	6	2	15	7
1-1	353	395	340	382
2-1 ou 1-2	41	17	52	25
2-2	4	2	-	-
2-3 ou 3-2	1	-	-	-
Total	**405**	**416**	**407**	**414**

As taxas de erro em cada categoria considerando-se os alinhamentos corretos – total e parcialmente – e errados nos corpora alinhados pelos métodos GSA+ e TCA também foram verificadas. Os resultados são mostrados na Tabela 33 e na Tabela 34, respectivamente.

Tabela 33: Análise da taxa de erro do método GSA+.

	CAT			CPT		
Categoria	**Parcialmente**	**Corretos**	**Errados**	**Parcialmente**	**Corretos**	**Errados**
0-1	0	0	1	0	0	0
1-0	0	1	0	0	1	1
1-1	9	345	4	2	395	0
1-2	2	32	0	1	15	0
2-1	1	8	3	0	2	0
2-2	0	0	0	0	0	0
Total	**12**	**386**	**8**	**3**	**413**	**1**

Tabela 34: Análise da taxa de erro do método TCA.

	CAT			CPT		
Categoria	**Parcialmente**	**Corretos**	**Errados**	**Parcialmente**	**Corretos**	**Errados**
0-1	0	0	1	0	0	1
1-0	0	3	11	0	1	5
1-1	11	328	1	5	376	1
1-2	14	30	0	10	12	0
2-1	1	6	1	1	1	1
2-2	0	0	0	0	0	0
Total	**26**	**367**	**14**	**16**	**390**	**8**

Como a classe de métodos híbridos avaliada nesta Seção possui dois representantes no projeto PESA (GSA+ e TCA), uma avaliação comparativa entre eles também foi efetuada. Os resultados dessa comparação são apresentados na Tabela 35.

Tabela 35: Análise comparativa dos métodos GSA+ e TCA.

	Corpus Alinhado pelo GSA+		Corpus Alinhado pelo TCA	
Alinhamentos Propostos	**Autêntico**	**Pré-editado**	**Autêntico**	**Pré-editado**
Parcialmente Corretos	12 (2,96%)	3 (0,72%)	26 (6,39%)	16 (3,87%)
Totalmente Corretos	386 (95,07%)	413 (99,04%)	367 (90,17%)	390 (94,20%)
Errados	8 (1,97%)	1 (0,24%)	14 (3,44%)	8 (1,93%)
Total	**406**	**417**	**407**	**414**

Os valores da Tabela 35 evidenciam que o método GSA+ obteve um desempenho melhor do que o método TCA. Além disso, em termos computacionais o TCA demorou o dobro do tempo para alinhar os corpora do que o GSA+. Mas é importante ressaltar que nenhuma dependência do tempo de processamento em relação ao tamanho dos textos sendo alinhados, que não a linear, foi verificada em ambos os métodos.

As próximas seções apresentam algumas considerações sobre os métodos híbridos GSA+ e TCA e alguns exemplos de bitextos alinhados por ambos.

7.3.1 Considerações sobre o método GSA+

Nesta Seção são apresentadas algumas considerações sobre o desempenho do método GSA+ no alinhamento dos corpora de teste CAT e CPT. Os resultados desta análise serão comparados com os resultados da avaliação do método descrita em (Melamed, 2000). Esta avaliação será referenciada nesta Seção como **avaliação1**, e a efetuada no projeto PESA, como **avaliação2**.

Na avaliação1, a precisão relatada ficou entre 98,2% e 98,7% de acordo com a "dificuldade" de alinhamento dos textos no corpus utilizado para teste: textos de debates do Parlamento Canadense extraídos do *Canadian Hansard Corpus*. Esses textos paralelos, escritos nas línguas em inglês e francês, foram divididos em dois conjuntos de acordo com a "dificuldade" de alinhamento: o "easy" Hansard, com 7123 alinhamentos; e o "hard" Hansard, com 2693 alinhamentos. A avaliação foi feita com base em um alinhamento de referência considerando-se apenas alinhamentos corretos e incorretos (os parcialmente corretos não foram contados). Além disso, o método GSA+ foi comparado com outros métodos de alinhamento sentencial de textos paralelos, entre eles os métodos empíricos GC e GMA. Os valores da avaliação1 para os métodos GC, GMA e GSA+ são mostrados na tabela 36.

Tabela 36: Comparação da taxa de erro dos métodos GC, GMA e GSA+ (avaliação1) (Melamed, 2000).

	"easy" Hansard			**"hard" Hansard**			**Total**		
Algoritmo	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
GC	7123	128	1,8	2693	80	3,0	9816	208	2,12
GMA	7123	115	1,6	2693	61	2,3	9816	176	1,79
GSA+	7123	90	1,3	2693	48	1,8	9816	138	1,41

Os resultados descritos na avaliação1 se aproximam dos relatados na avaliação desempenhada no projeto PESA como mostra a Tabela 37.

Tabela 37: Comparação da taxa de erro dos métodos GC e GMA para os corpora CAT e CPT (avaliação2).

	Corpus Autêntico			**Corpus Pré-editado**			**Total**		
Algoritmo	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
GC	405	40	9,88	416	11	2,64	821	51	6,21
GMA	405	18	4,44	416	3	0,72	821	21	2,56
GSA+	405	19	4,69	416	3	0,72	821	22	2,68

As precisões relatadas na avaliação1 – 97,88% (GC), 98,21% (GMA) e 98,59% (GSA+) – foram um pouco maiores do que as relatadas na avaliação2 – 93,79% (GC), 97,44%

(GMA) e 97,32% (GSA+). Esta diferença entre os valores das duas avaliações pode estar relacionada ao fato de textos com ruído (do CAT) terem sido utilizados na avaliação2, o que afeta o desempenho dos métodos de alinhamento sentencial de textos paralelos segundo (Gaussier et al., 2000).

A avaliação1 não verificou a taxa de erro por categoria de alinhamento para o método GSA+, por isso apenas os valores da avaliação2 são apresentados na Tabela 38.

Tabela 38: Análise da taxa de erro por categoria de alinhamentos dos corpora alinhados pelo método GSA+ (avaliação2).

	Corpus Autêntico			**Corpus Pré-editado**			**Total**		
Categoria	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
0-1 ou 1-0	6	5	83,33	2	1	50	8	6	75
1-1	353	8	2,27	395	0	0	748	8	1,07
2-1 ou 1-2	41	1	2,44	17	0	0	58	1	1,72
2-2	4	4	100	2	2	100	6	6	100
2-3	1	1	100	0	0	0	1	1	100
Total	**405**	**19**		**416**	**3**		**821**	**22**	

Com base nos valores da Tabela 38 pode-se perceber que a menor taxa de erro foi constatada nos alinhamentos 1-1 (1,07% no total); seguidos pelos alinhamentos 2-1 (ou 1-2), que apresentaram uma taxa de erro (1,72%) muito próxima à dos primeiros (1-1). Os alinhamentos 2-2 apresentaram uma taxa de erro de 100%, maior do que a relatada para o método GMA nesta categoria (83,33%). O alinhamento 2-3, como nos outros métodos, também apresentou taxa de erro de 100%. Já 25% dos casos de omissão (0-1 e 1-0), que foram completamente perdidos no método GC, foram alinhados corretamente pelo método GSA+ assim como pelo método GMA. Este fato comprova, mais uma vez, que a existência de informação específica a respeito das línguas envolvidas (cognatos e LPA, neste caso) melhora o desempenho do método de alinhamento sentencial nos casos de omissão (vide Seção 7.1.1).

A seguir são apresentados alguns exemplos de bitextos alinhados pelo método GSA+ comparados aos mesmos bitextos alinhados pelo método empírico GMA (Seção 7.1.2). O GSA+ apresentou melhor desempenho do que o GMA em apenas um caso (Exemplo 7.3.1-1) e o mesmo resultado nos outros quatro (Exemplo 7.3.1-2, Exemplo 7.3.1-3, Exemplo 7.3.1-4 e Exemplo 7.3.1-5).

Exemplo 7.3.1-1: Um alinhamento 1-2 considerado como um alinhamento 0-1 seguido de um 1-1 pelo GMA (vide Exemplo 7.1.2-1) e alinhado corretamente pelo GSA+.

art10R-art10A do CAT após ser alinhado pelo método GSA+

<s id=**art10R.1.s1** corresp='**art10A.1.s1 art10A.1.s2**'>O SPP2 (Servidor de Processamento Paralelo), desenvolvido no Laboratório de Computação de Alto Desempenho (LCAD-ICMC-USP) utiliza computadores convencionais conectados por uma rede de comunicação de alta velocidade.</s>	<s id=**art10A.1.s1** corresp=**art10R.1.s1**>Conventional computers connected by high-speed communication networks present a very low cost alternative to the MPPs (Massively Parallel Processors) for applications that demand high computing power.</s><s id=**art10A.1.s2** corresp=**art10R.1.s1**>The SPP2 (Parallel Processing Server), developed at the LCAD-ICMC-USP, is one of these systems.</s>

Exemplo 7.3.1-2: Um alinhamento 1-0 seguido de um alinhamento 1-1 considerado como dois alinhamentos 1-1 pelo GMA e pelo GSA+.

es7R-es7A do CAR

<s id=**es7R.1.s3** corresp=**''**>Dessa forma, quando diante da manutenção do produto, o engenheiro de software encontra uma documentação informal e incompleta, que não reflete o software existente.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse contexto é que se encontra a Engenharia Reversa de Software, com o propósito de recuperar as informações de projeto perdidas durante a fase de desenvolvimento, e de documentar o real estado do software.</s>	<s id=**es7A.1.s3** corresp=**es7R.1.s4**>In this context Reverse Engineering of Software can help by means of recovering the project information lost during the development phase and documenting the current software state.</s>

es7R-es7A do CAT após ser alinhado pelo método GSA+

<s id=**es7R.1.s3** corresp=**es7A.1.s2**>Dessa forma, quando diante da manutenção do produto, o engenheiro de software encontra uma documentação informal e incompleta, que não reflete o software existente.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse contexto é que se encontra a Engenharia Reversa de Software, com o propósito de recuperar as informações de projeto perdidas durante a fase de desenvolvimento, e de documentar o real estado do software.</s>	<s id=**es7A.1.s2** corresp=**es7R.1.s3**>The maintenance of such software is problematic, since its documentation rarely reflects the implemented code.</s><s id=**es7A.1.s3** corresp=**es7R.1.s4**>In this context Reverse Engineering of Software can help by means of recovering the project information lost during the development phase and documenting the current software state.</s>

Exemplo 7.3.1-3: Um alinhamento 2-2 considerado como dois alinhamentos 1-1 pelo GMA e pelo GSA+.

es12R-es12A do CAR

<s id=**es12R.3.s1** corresp='**es12A.3.s1 es12A.3.s2**'>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp='**es12A.3.s1 es12A.3.s2**'>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp='**es12R.3.s1 es12R.3.s2**'>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp='**es12R.3.s1 es12R.3.s2**'>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

es12R-es12A do CAT após ser alinhado pelo método GSA+

<s id=**es12R.3.s1** corresp=**es12A.3.s1**>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp=**es12A.3.s2**>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp=**es12R.3.s1**>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp=**es12R.3.s2**>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

Exemplo 7.3.1-4: Um alinhamento 1-2 alinhado corretamente pelos métodos GMA e GSA+.

art8R-art8A do CAT após ser alinhado pelo método GSA+

<s id=**art8R.1.s6** corresp='**art8A.1.s6 art8A.1.s7**'>O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total.</s>	<s id=**art8A.1.s6** corresp=**art8R.1.s6**>The Clustered Knapsack Problem consists of determining the suitable capacities of each cluster and how these clusters should be filled.</s><s id=**art8A.1.s7** corresp=**art8R.1.s6**>The objective is to maximize a total utility value.</s>

Exemplo 7.3.1-5: Um alinhamento 1-1 seguido de um alinhamento 1-0 alinhado corretamente pelos métodos GMA e GSA+.

bd1R-bd1A do CAT após ser alinhado pelo método GSA+

<s id=**bd1R.1.s4** corresp=**bd1A.1.s3**>Por exemplo, se duas organizações devem trocar dados sobre pessoas, não importa se para as diferentes organizações as pessoas são clientes, empregados, alunos ou pacientes, o significado de "pessoa" é sempre entendido pelos membros das organizações.</s><s id=**bd1R.1.s5** corresp=**''**>O mesmo ocorre com qualquer entidade que se deseje trocar informações.</s>	<s id=**bd1A.1.s3** corresp=**bd1R.1.s4**>For example, if two organizations should interchange data on people, it does not care, for the different organizations, if the people are customers, employees, students or patient, the means of what are "people" is always understood by each organization.</s>

7.3.2 Considerações sobre o método TCA

Nesta Seção são apresentadas algumas considerações sobre o desempenho do método TCA no alinhamento dos corpora de teste CAT e CPT. Para enriquecer essa análise, os resultados da avaliação descrita em (Hofland, 1996) foram comparados com os da avaliação efetuada no projeto PESA. Essas duas avaliações serão referenciadas no restante desta Seção como **avaliação1** e **avaliação2**, respectivamente.

A precisão média relatada na avaliação1 foi de 98,02% em um corpus composto por 51 bitextos em inglês e norueguês, coletados no projeto *English-Norwegian Parallel Corpus* (ENPC)[32]. Esse corpus era composto por 93868 sentenças (60678 em inglês e 33190 em norueguês) e 1,3 milhões de palavras. Os textos alinhados pelo TCA foram verificados manualmente considerando-se apenas duas possibilidades para os alinhamentos: corretos ou incorretos (os parcialmente corretos foram desconsiderados).

Uma análise das categorias de alinhamentos nos textos alinhados pelo TCA realizada na avaliação1 constatou que o TCA prioriza alinhamentos da categoria 1-1 (96,40%) (Hofland, 1996). Na avaliação2 também se verificou a preferência do TCA para alinhamentos 1-1 sendo a porcentagem total de 91,11% nos dois corpora de teste CAT e CPT.

Enquanto que na avaliação1, o método TCA apresentou uma precisão média 98,02% e em outra avaliação, realizada em (Santos & Oksefjell, 2000) para textos em inglês e português europeu, o método apresentou uma precisão de 97,1%; na avaliação2, constataram-se precisões menores tanto para o CAT (90,62%) quanto para o CPT (93,75%).

Na avaliação2, o TCA cometeu 38 erros num total de 405 alinhamentos (9,38%) para o CAT e 26 erros em 416 alinhamentos (6,25%) para o CPT. No total foram 64 erros em 821 alinhamentos (7,80%), como pode ser observado na Tabela 39.

Tabela 39: Análise da taxa de erro por categoria nos corpora alinhados pelo método TCA (avaliação2).

	Corpus Autêntico			**Corpus Pré-editado**			**Total**		
Categoria	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**	**Total**	**Errados**	**%**
0-1 ou 1-0	6	3	50	2	1	50	8	4	50
1-1	353	25	7,08	395	19	4,81	748	44	5,88
2-1 ou 1-2	41	5	12,2	17	4	23,53	58	9	15,52
2-2	4	4	100	2	2	100	6	6	100
2-3	1	1	100	0	0	0	1	1	100
Total	**405**	**38**		**416**	**26**		**821**	**64**	

[32] Site do ENPC: http://www.hf.uio.no/iba/prosjekt/ (17/02/2003)

Com base nos valores da Tabela 39, pode-se perceber que a menor taxa de erro foi constatada nos alinhamentos 1-1 (5,88%); seguidos pelos alinhamentos 2-1 (ou 1-2), que apresentaram uma taxa de erro três vezes maior do que os primeiros (1-1). Os alinhamentos 2-2 e 2-3 apresentaram uma taxa de erro de 100%, ou seja, todos os alinhamentos destas categorias forma alinhados incorretamente. Por outro lado, este método foi o que apresentou melhor desempenho no alinhamento dos casos de omissão (0-1 ou 1-0): 50% deles foram alinhados corretamente. Este fato comprova, mais uma vez, que o tratamento destes casos requer a utilização de informações específicas sobre as línguas envolvidas.

A seguir são apresentados alguns exemplos de bitextos alinhados pelo método TCA e os respectivos alinhamentos corretos do corpus de referência correspondente.

Com relação aos exemplos apresentados nas seções anteriores para os outros quatro métodos, o TCA alinhou corretamente os exemplos 7.3.2-1, 7.3.2-2, 7.3.2-4 e 7.3.2-5. O outro exemplo, Exemplo 7.3.2-3 foi alinhado da mesma forma que os demais métodos.

Exemplo 7.3.2-1: Um alinhamento 1-2 alinhado corretamente pelo método TCA.

art10R-art10A do CAT após ser alinhado pelo método TCA

<s id=**art10R.1.s1** corresp='**art10A.1.s1 art10A.1.s2**'>O SPP2 (Servidor de Processamento Paralelo), desenvolvido no Laboratório de Computação de Alto Desempenho (LCAD-ICMC-USP) utiliza computadores convencionais conectados por uma rede de comunicação de alta velocidade.</s>	<s id=**art10A.1.s1** corresp=**art10R.1.s1**>Conventional computers connected by high-speed communication networks present a very low cost alternative to the MPPs (Massively Parallel Processors) for applications that demand high computing power.</s><s id=**art10A.1.s2** corresp=**art10R.1.s1**>The SPP2 (Parallel Processing Server), developed at the LCAD-ICMC-USP, is one of these systems.</s>

Exemplo 7.3.2-2: Um alinhamento 1-0 seguido de um alinhamento 1-1 alinhado corretamente pelo método TCA.

es7R-es7A do CAT após ser alinhado pelo método TCA

<s id=**es7R.1.s3** corresp=**''**>Dessa forma, quando diante da manutenção do produto, o engenheiro de software encontra uma documentação informal e incompleta, que não reflete o software existente.</s><s id=**es7R.1.s4** corresp=**es7A.1.s3**>Nesse contexto é que se encontra a Engenharia Reversa de Software, com o propósito de recuperar as informações de projeto perdidas durante a fase de desenvolvimento, e de documentar o real estado do software.</s>	<s id=**es7A.1.s3** corresp=**es7R.1.s4**>In this context Reverse Engineering of Software can help by means of recovering the project information lost during the development phase and documenting the current software state.</s>

Exemplo 7.3.2-3: Um alinhamento 2-2 considerado como dois alinhamentos 1-1.

es12R-es12A do CAR

<s id=**es12R.3.s1** corresp='**es12A.3.s1 es12A.3.s2**'>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp='**es12A.3.s1 es12A.3.s2**'>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp='**es12R.3.s1 es12R.3.s2**'>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp='**es12R.3.s1 es12R.3.s2**'>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

es12R-es12A do CAT após ser alinhado pelo método TCA

<s id=**es12R.3.s1** corresp=**es12A.3.s1**>Dessa forma, neste trabalho é apresentada uma ferramenta de injeção de defeitos de software, denominada ITool, baseada em um esquema de injeção de defeitos.</s><s id=**es12R.3.s2** corresp=**es12A.3.s2**>Esse esquema caracteriza o mapeamento de uma taxonomia de defeitos de software (Taxonomia de DeMillo) para os operadores de mutação do critério de teste Análise de Mutantes para a linguagem C.</s>	<s id=**es12A.3.s1** corresp=**es12R.3.s1**>In this perspective, in this work a software fault injection tool, named ITool, is presented.</s><s id=**es12A.3.s2** corresp=**es12R.3.s2**>This tool is based on a fault injection scheme that defines the mapping of a software fault taxonomy (DeMillo's Taxonomy) to the mutation operators of the Mutation Analysis criterion for C language.</s>

Exemplo 7.3.2-4: Um alinhamento 1-2 alinhado corretamente pelo TCA.

art8R-art8A do CAT após ser alinhado pelo método TCA

<s id=**art8R.1.s6** corresp='**art8A.1.s6 art8A.1.s7**'>O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total.</s>	<s id=**art8A.1.s6** corresp=**art8R.1.s6**>The Clustered Knapsack Problem consists of determining the suitable capacities of each cluster and how these clusters should be filled.</s><s id=**art8A.1.s7** corresp=**art8R.1.s6**>The objective is to maximize a total utility value.</s>

Exemplo 7.3.2-5: Um alinhamento 1-1 seguido de um alinhamento 1-0 alinhado corretamente pelo TCA.

bd1R-bd1A do CAT após ser alinhado pelo método TCA

<s id=**bd1R.1.s4** corresp=**bd1A.1.s3**>Por exemplo, se duas organizações devem trocar dados sobre pessoas, não importa se para as diferentes organizações as pessoas são clientes, empregados, alunos ou pacientes, o significado de "pessoa" é sempre entendido pelos membros das organizações.</s><s id=**bd1R.1.s5** corresp=**''**>O mesmo ocorre com qualquer entidade que se deseje trocar informações.</s>	<s id=**bd1A.1.s3** corresp=**bd1R.1.s4**>For example, if two organizations should interchange data on people, it does not care, for the different organizations, if the people are customers, employees, students or patient, the means of what are "people" is always understood by each organization.</s>

7.4 Conclusões

Todos os métodos de alinhamento sentencial de textos paralelos avaliados no projeto PESA mostraram-se computacionalmente tratáveis e eficientes, e mantiveram a complexidade linear em relação ao tamanho dos textos. Além disso, eles apresentaram as características desejáveis para os métodos de alinhamento de textos paralelos, escalabilidade e confiabilidade, citadas no Capítulo 2. Com relação à escalabilidade, todos os métodos foram modificados para se adequarem aos requisitos do projeto PESA e, em alguns, outras alterações removeram as limitações das versões originais.

Uma das constatações mais relevantes nas avaliações descritas nesta Seção é a de que o desempenho de todos os métodos foi melhor nos corpora sem ruídos, comprovando a hipótese levantada no momento da construção dos corpora de teste e baseada no que já havia sido relatado na literatura: o alinhamento automático de sentenças torna-se mais eficaz quando o corpus está relativamente isento de ruídos (sem erros gramaticais ou de tradução) e provém de domínios técnicos, nos quais as traduções literais são esperadas (Gaussier et al., 2000).

A precisão apresentada pelos métodos em relação aos limites relatados na literatura – acima de 95% (vide Capítulo 2) foi a seguinte: no alinhamento do CAT apenas o método GSA+ obteve uma precisão maior do que este valor (95,07%) e no alinhamento do CPT, apenas o método TCA obteve uma precisão menor do que esse valor (94,20%). Os métodos que obtiveram melhor desempenho geral (CAT e CPT) foram: o método empírico GMA e sua versão híbrida (o GSA+) com precisões de aproximadamente 95% e 99% para os corpora CAT e CPT, respectivamente.

As categorias com maiores taxas de erro foram: 2-3, 2-2 e os casos de omissão (0-1 e 1-0). A taxa de erro para a categoria 2-3 foi de 100% em todos os métodos, ou seja, nenhum deles alinhou corretamente o único caso de alinhamento 2-3 (pertencente ao CAT). Com relação aos alinhamentos 2-2, apenas os métodos empíricos GC e GMA não apresentaram 100% de erro, a taxa de erro desses métodos foi de 83,33%. Nos casos de omissão, o método que obteve a menor taxa de erro foi o TCA (50%), seguido pelos métodos GMA e GSA+ (75% cada). Os demais métodos apresentaram 100% de erro nas categorias 0-1 e 1-0. A categoria de menor taxa de erro, por sua vez, foi a 1-1: de 0,80% a 5,88%.

Capítulo 8

Conclusão e Contribuições

O objetivo principal do projeto PESA – estudo, implementação e avaliação de métodos de alinhamento sentencial de textos paralelos para o par de línguas PB-inglês – foi alcançado com sucesso.

Os métodos implementados apresentaram precisões de 85,89% a 95,07% no alinhamento de textos com ruídos (erros gramaticais e de tradução) e de 94,20% a 99,04% em textos limpos (sem tais erros). Considerando-se os casos dos textos limpos, as precisões apresentadas, em sua maioria, estão de acordo com os limites relatados na literatura – acima de 95%. Além disso, os resultados apontam muita semelhança no desempenho de todos os métodos, o que impossibilita a eleição de um deles como o melhor.

Como trabalho futuro, pretende-se analisar o desempenho destes métodos em outros corpora paralelos, com textos de tamanhos e domínios diferentes. Uma destas avaliações é apresentada no Apêndice A: Avaliação com o corpusALCA. Este corpus, ainda em construção, é composto por textos jurídicos extraídos da documentação oficial da ALCA (Área de Livre Comércio das Américas) disponível na *web* nos idiomas PB e inglês.

Na avaliação descrita no Apêndice A, todos os métodos apresentaram precisões acima de 98% sendo que um deles alinhou corretamente todos os textos do corpus (precisão de 100%). A justificativa para uma maior precisão geral nesta avaliação, pode estar relacionada à predominância da categoria de alinhamento 1-1 a qual foi apontada na Seção 7.4 como a de menor taxa de erro. Porém, os valores apresentados na avaliação do corpusALCA não permitem separar a influência do domínio e do tamanho dos textos no desempenho dos métodos de alinhamento sentencial. Portanto, novos testes deverão ser realizados para se determinar a influência de cada um desses fatores e, assim, chegar a conclusões mais confiáveis.

Além dos cinco programas de alinhamento sentencial de textos paralelos, outra ferramenta computacional produzida no projeto PESA foi a TagAlign: uma ferramenta de pré-processamento dos textos construída para auxiliar algumas tarefas do projeto PESA. Entre essas tarefas estão a etiquetação de fronteiras de parágrafos e sentenças nos textos dos corpora de teste, feita automaticamente pela TagAlign; a marcação semi-automática dos alinhamentos sentenciais nos textos dos corpora de referência e a avaliação dos métodos de alinhamento

sentencial pela comparação automática dos corpora alinhados por eles com os corpora de referência.

Outros recursos não computacionais também foram produzidos no projeto PESA como: os corpora de teste, os corpora de referência, os corpora etiquetados morfologicamente, as listas de palavras âncoras e diversos relatórios técnicos e artigos. Muitos desses recursos poderão ser usados como recursos lingüísticos em trabalhos futuros, como uma pesquisa equivalente a esta para o alinhamento de palavras e construção de glossários (projeto PEWA, em andamento no NILC) e a construção de ferramentas de apoio à tradução automática.

Apesar da grande importância de todos os recursos citados, a contribuição mais relevante do projeto PESA está na análise detalhada e inédita da performance de métodos de alinhamento sentencial de textos paralelos envolvendo os idiomas português brasileiro e inglês. Essa contribuição é de extrema relevância devido à pequena quantidade de pesquisas na área de Lingüística Computacional envolvendo o português (tanto o europeu quanto o brasileiro).

Assim, não se pode considerar o projeto PESA como finalizado, mas sim como ponto de partida para diversas pesquisas que se aproveitarão dos recursos e resultados produzidos por ele para estender um pouco mais os estudos envolvendo nossa língua. Uma dessas pesquisas – o projeto de doutorado proposto pela autora deste texto – pretende extrair equivalências de tradução dos corpora paralelos alinhados sentencialmente. Essas equivalências podem ser lexicais, gramaticais ou referentes à estrutura das línguas que compõem os corpora. Inicialmente serão utilizados os corpora de referência do projeto PESA, porém outros estão sendo construídos também para o par PB-espanhol.

Apêndice A

Avaliação com o CorpusALCA

Após a implementação e avaliação dos métodos de alinhamento sentencial de textos paralelos com os corpora de teste CAT e CPT, sentiu-se a necessidade de avaliá-los com um outro corpus composto por textos de domínio diferente do científico e/ou com um número maior de sentenças. Assim surgiu o corpusALCA, composto, até o momento, por 4 textos paralelos extraídos da documentação oficial da ALCA[33] num total de 725 sentenças e 22069 palavras. O corpusALCA satisfaz os dois requisitos citados anteriormente: seus textos são de um domínio diferente do científico (o jurídico) e possuem um número maior de sentenças (quase 100 sentenças cada). A Tabela 40 apresenta esses números divididos de acordo com a língua.

Tabela 40: Quantidade de palavras e sentenças nos corpusALCA.

	Corpus ALCA
Palavras em PB	11217
Palavras em inglês	10852
Sentenças em PB	362
Sentenças em inglês	363

Os textos presentes no corpusALCA foram processados de maneira semelhante aos textos dos demais corpora do projeto PESA, como explicado no Capítulo 3. Assim surgiram o corpusALCA de teste (ou CALCAT), o corpusALCA de teste etiquetado morfologicamente (ou CALCATE) e o corpusALCA de referência (ou CALCAR).

Além dos corpora, uma lista de palavras âncoras (LPA) para o domínio jurídico também foi construída: a LPA_ALCA. A LPA_ALCA foi gerada a partir de sentenças da Constituição Brasileira de 1988[34] e dos Protocolos de Brasília e de Olivos extraídos da documentação oficial do Mercosul (Mercado Comum do Sul)[35]. Trata-se de textos paralelos dos quais foram extraídas as equivalências de traduções entre as palavras em PB e em inglês de maneira semelhante à descrita na Seção 3.4.

Na construção da LPA_ALCA, ao contrário da LPA apresentada na Seção 3.4, as palavras muito freqüentes (artigos, preposições, conjunções, etc.) foram desconsideradas,

[33] Página oficial da ALCA: http://www.ftaa-alca.org/alca_p.asp.
[34] Disponível eletronicamente em: http://www.georgetown.edu/pdba/Constitutions/Brazil/english98.html (versão em inglês) e http://www.georgetown.edu/pdba/Constitutions/Brazil/brazil88.html (versão em PB) (17/02/2003)
[35] Disponível eletronicamente em: http://www.mercosur.org.uy.

pois, conforme constatação na avaliação anterior, estas palavras geram ruídos no processo de alinhamento e pioram sua performance. Desta forma, a LPA_ALCA possui, até o momento, cerca de 300 entradas.

O Quadro 14, da mesma forma que o Quadro 4, mostra a distribuição desses recursos lingüísticos em relação aos métodos e às fases nas quais eles são usados (alinhamento, teste ou avaliação).

Quadro 14: Recursos lingüísticos referentes ao corpusALCA utilizados pelos métodos no projeto PESA.

Método	Classificação	Alinhamento	Teste	Avaliação
GC	Empírico	-	CALCAT	CALCAR
GMA	Empírico	-	CALCAT	CALCAR
Lingüístico	Lingüístico	-	CALCATE	CALCAR
GSA+	Híbrido	LPA_ALCA	CALCAT	CALCAR
TCA	Híbrido	LPA_ALCA	CALCAT	CALCAR

Nessa avaliação, o único método para o qual novos parâmetros foram calculados foi o método lingüístico. Um novo modelo de regressão linear (vide Seção 5.1.1) foi gerado a partir de sentenças do CALCATE como mostrado em (14). A variância estimada neste caso foi 3,20.

$$Y = -0{,}55 + 1{,}45\,X_1 + 0{,}978\,X_2 + 1{,}03\,X_3 + 0{,}248\,X_4 \qquad (14)$$

Os cinco métodos do Quadro 14 foram avaliados com esses novos recursos lingüísticos e os valores para *precision, recall* e *F-measure* para todos eles são apresentados na Tabela 41.

Tabela 41: Métricas calculadas para o corpusALCA alinhado pelos cinco métodos.

	CorpusALCA				
Métricas	**GC**	**GMA**	**Lingüístico**	**GSA+**	**TCA**
precision	0,9917	0,9876	0,9833	0,9876	1,0000
recall	0,9890	0,8788	0,9725	0,8788	1,0000
F	0,9903	0,9300	0,9778	0,9300	1,0000

A análise das categorias de alinhamento mais freqüentes nestes métodos com relação às categorias do corpus de referência (CALCAR) também foi feita e seus resultados são mostrados na Tabela 42.

Tabela 42: Análise das categorias de alinhamentos do corpusALCA alinhado pelos cinco métodos.

Categoria	CALCAR	GC	GMA	Lingüístico	GSA+	TCA
0-1	1	-	1	-	1	1
1-1	362	361	322	355	322	362
1-2	-	1	-	1	-	-
2-2	-	-	-	3	-	-
Total	**363**	**362**	**323**	**359**	**323**	**363**

As taxas de erro em cada categoria, considerando-se os alinhamentos corretos – total e parcialmente – e errados no corpus alinhado pelos métodos, também foram verificadas e os resultados são apresentados, de acordo com a classe dos métodos, nas três tabelas que seguem. A Tabela 43 apresenta os valores dos métodos empíricos, a Tabela 44, os do método lingüístico e a Tabela 45, os dos métodos híbridos.

Tabela 43: Análise da taxa de erro dos métodos empíricos GC e GMA.

	CALCAT alinhado pelo GC			CALCAT alinhado pelo GMA		
Categoria	**Parcialmente**	**Corretos**	**Errados**	**Parcialmente**	**Corretos**	**Errados**
0-1	0	0	0	0	0	1
1-1	0	359	2	0	319	3
1-2	1	0	0	0	0	0
Total	**1**	**359**	**2**	**0**	**319**	**4**

Tabela 44: Análise da taxa de erro do método lingüístico.

	CALCAT alinhado pelo método lingüístico		
Categoria	**Parcialmente**	**Corretos**	**Errados**
1-1	0	353	2
1-2	1	0	0
2-2	3	0	0
Total	**4**	**353**	**2**

Tabela 45: Análise da taxa de erro dos métodos híbridos GSA+ e TCA.

	CALCAT alinhado pelo GSA+			CALCAT alinhado pelo TCA		
Categoria	**Parcialmente**	**Corretos**	**Errados**	**Parcialmente**	**Corretos**	**Errados**
0-1	0	0	1	0	1	0
1-1	0	319	3	0	362	0
1-2	0	0	0	0	0	0
Total	**0**	**319**	**4**	**0**	**363**	**0**

Por fim, uma análise comparativa das taxas de erro dos cinco métodos no alinhamento do corpusCALCA é apresentada na Tabela 46.

Tabela 46: Análise comparativa dos cinco métodos.

Alinhamentos Propostos	GC	GMA e GSA+	Lingüístico	TCA
Parcialmente Corretos	1 (0,28%)	0	4 (1,11%)	0
Totalmente Corretos	359 (99,17%)	319 (98,76%)	353 (98,33%)	363 (100%)
Errados	2 (0,55%)	4 (1,24%)	2 (0,56%)	0
Total	**362**	**323**	**359**	**363**

Os valores da Tabela 46 evidenciam que, no alinhamento do corpusALCA, o método TCA foi o que obteve melhor desempenho, ao contrário do que havia sido verificado no alinhamento dos corpora CAT e CPT, no qual os métodos GMA e GSA+ obtiveram os melhores resultados (vide Seção 7.4). Este fato pode ser explicado pelas constatações apresentadas na Seção 7.3.2 de que a menor taxa de erro do método TCA está em alinhamentos 1-1 e omissões, ou seja, as únicas existentes no corpusALCA.

Com relação ao tempo de processamento, os métodos GMA e GSA+ levaram três vezes mais tempo para alinhar o corpusALCA do que para alinhar os outros corpora. Este fato indica que o desempenho destes métodos pode estar relacionado ao tamanho do corpus que eles processam, enquanto que os outros métodos (inclusive o TCA) mantiveram seu tempo de processamento constante, ou seja, levaram praticamente o mesmo tempo para alinhar o corpusALCA e os demais corpora (CAT e CPT).

Nesta avaliação não foi possível isolar a influência do domínio e do tamanho dos textos paralelos no desempenho (precisão e tempo de processamento) dos métodos de alinhamento. Sabe-se, porém, que um ou ambos influenciam o processo de alinhamento dos métodos, pois estes apresentaram desempenhos diferentes no alinhamento dos corpora jurídico (corpusALCA) e científico (CAT e CPT).

Assim, os valores apresentados com a avaliação do corpusALCA despertam o interesse para novos testes para se verificar em que medida o tamanho do corpus influencia o desempenho dos métodos. Da mesma forma, a influência dos domínios aos quais os textos pertencem também deverá ser analisada.

Referências Bibliográficas

AIRES, R.V.X. (2000). *Implementação, Adaptação, Combinação e Avaliação de Etiquetadores para o português do Brasil*. Dissertação (mestrado) – Instituto de Ciências Matemáticas e de Computação (ICMC), Universidade de São Paulo, São Carlos, SP, outubro.

AIRES, R.V.X.; ALUÍSIO, S.M. (2001). Criação de um corpus com 1.000.000 de palavras etiquetado morfossintaticamente, *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional*, NILC-TR-01-8, outubro.

BRILL, E. (1995). Transformation-based error-driven learning of natural language: A case study in part of speech tagging. *Computational Linguistics*, v.21, n.4, p.543-565, dezembro. Disponível em 18/02/2003 (http://www.cs.jhu.edu/~brill/papers.html).

BROWN, P.F.; LAI, J.C.; MERCER, R.L. (1991). Aligning sentences in parallel corpora. In: *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, Berkley. p.169-176.

CAMPBELL, J.A.; CHATTERJEE, N.; DAWKINS, N. (1998). Experiments in Automated Alignment of Text over Several Languages. In: *Proceedings of the International Conference on Computational Linguistics*, *Speech and Document Processing*, Indian Statistical Institute, Calcutta. p.C-47-C-54.

CASELI, H.M.; NUNES, M.G.V. (2002). A construção dos recursos lingüísticos do projeto PESA. *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional,* NILC-TR-02-07, junho.

CASELI, H.M.; FELTRIM, V.D.; NUNES, M.G.V. (2002). TagAlign: Uma ferramenta de pré-processamento de textos. *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional,* NILC-TR-02-09, junho.

CHUANG, T.C; YOU, GN; CHANG, J.S. (2002). Adaptive Bilingual Sentence Alignment. In: *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas (AMTA 2002)*, Tiburon, CA, USA, outubro. p. 21-30.

FELTRIM, V.D.; NUNES, M.G.V.; ALUÍSIO, S.M. (2001). Um corpus de textos científicos em português para a análise da Estrutura Esquemática. *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional,* NILC-TR-01-4, julho.

GALE, W.A.; CHURCH, K.W. (1991). A program for aligning sentences in bilingual corpora. In: *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics (ACL)*, Berkley. p.177-184.

GALE, W.A.; CHURCH, K.W. (1993). A program for aligning sentences in bilingual corpora. *Computational Linguistics*, v.19, n.3, p.75-102.

GAUSSIER, E.; HULL, D.; AÏT-MOKTHAR, S. (2000). Term alignment in use: Machine-aided human translation. In: VÉRONIS, J. (ed.). *Parallel text processing: Alignment and use of translation corpora.* s.l.: Kluwer Academic Publishers. p.253-274.

HOFLAND, K. (1996). A program for aligning English and Norwegian sentences. In: HOCKEY, S.; IDE, N.; PERISSINOTTO, G. (eds.). *Research in Humanities Computing*. Oxford: Oxford University Press. p.165-178.

KAY, M.; RÖSCHEISEN, M. (1988). Text-translation alignment. *Technical Report*, Xerox Palo Alto Research Center.

KAY, M.; RÖSCHEISEN, M. (1993). Text-translation alignment. *Computational Linguistics*, v.19, n. 1, p.121-42.

MARTINS, M.S.; CASELI, H.M.; NUNES, M.G.V. (2001). A construção de um corpus de textos paralelos inglês-português. *Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional,* NILC-TR-01-05, setembro.

MELAMED, I.D. (1996). Porting SIMR to New Language Pairs. *Institute of Research in Cognitive Science Technical Report,* 96-26, Philadelphia, PA: University of Pennsylvania.

MELAMED, I.D. (2000). Pattern recognition for mapping bitext correspondence. In: VÉRONIS, J. (ed.). *Parallel text processing: Alignment and use of translation corpora.* s.l.: Kluwer Academic Publishers. p.25-47.

MELBY, A.K. (2000). Sharing of translation memory databases derived from aligned parallel text. In: VÉRONIS, J. (ed.). *Parallel text processing: Alignment and use of translation corpora.* s.l.: Kluwer Academic Publishers. p.347-368.

MOORE, R.C. (2002). Fast and Accurate Sentence Alignment of Bilingual Corpora. In: *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas (AMTA 2002)*, Tiburon, CA, USA, outubro. p.135-144.

OLIVEIRA JR., O.N.; MARCHI, A.R.; MARTINS, M.S.; MARTINS, R.T. (2000). A Critical Analysis of the Performance of English-Portuguese-English MT Systems. In: *V Encontro para o processamento computacional da Língua Portuguesa Escrita e Falada (PROPOR'2000)*, Atibaia, SP. p.85-92.

PAPAGEORGIOU, H.; CRANIAS, L.; PIPERIDIS, S. (1994). Automatic alignment in parallel corpora. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL 94)*, Las Cruses, New Mexico, junho. p.334-336.

PIPERIDIS, S.; PAPAGEORGIOU, H.; BOUTSIS, S. (2000). From sentences to words and clauses. In: VÉRONIS, J. (ed.). *Parallel text processing: Alignment and use of translation corpora.* s.l.: Kluwer Academic Publishers. p.117-138.

RATNAPARKHI, A. (1996). A Maximum Entropy Part-of-Speech Tagger. In: *Proceedings of the First Empirical Methods in Natural Language Processing Conference*.

RENOUF, A. (1987). Corpus development. In: SINCLAIR, J.M. (org.). *Looking up: An account of the COBUILD Project in lexical computing*. Londres/Glasgow: Collins. p.1-22.

RIBEIRO, A.; LOPES, G.; MEXIA, J.(2000a). Using confidence bands for parallel texts alignment. In: *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*, Hong Kong, China. p.432-439.

RIBEIRO, A.; LOPES, G.; MEXIA, J. (2000b). Linear regression based alignment of parallel texts using homograph words. In: *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI 2000)*, Berlin, Alemanha. p.446-450.

SANTOS, D.; OKSEFJELL, S. (2000). An evaluation of the Translation Corpus Aligner, with special reference to the language pair English-Portuguese. In: *Proceedings of the 12th "Nordisk datalingvistikkdager". Trondheim, Departmento de Lingüística*, NTNU. p.191-205.

SCHMID, H. (1995). Probabilistic Part-of-Speech Tagging Using Decision Trees. In: *Proceedings of the Conference on New Methods in Language Processing*, Manchester, UK.

SILVA, M.H.B. (1999). *A abordagem de críticas para a construção de sistemas de aprendizado da escrita técnica*. Dissertação (mestrado) – Instituto de Ciências Matemáticas e de Computação (ICMC), Universidade de São Paulo, São Carlos, SP.

SINCLAIR, J.M. (1991). *Corpus, concordance, collocation*. Oxford: Oxford University Press. p.13-26.

VÉRONIS, J. (2000). From the Rosetta stone to the information society: A survey of parallel text processing. In: VÉRONIS, J. (ed.). *Parallel text processing: Alignment and use of translation corpora.* s.l.: Kluwer Academic Publishers. p.1-24.

VÉRONIS, J.; LANGLAIS, P. (2000). Evaluation of parallel text alignment systems: The ARCADE project. In: VÉRONIS, J. (ed.). *Parallel text processing: Alignment and use of translation corpora.* s.l.: Kluwer Academic Publishers. p.369-388.

VIDAL, R.V.V. (1993) (Ed). *Applied Simulated Annealing*. Heidelberg: Springer-Verlag.