Indução de léxicos bilíngües e regras para a tradução automática

Helena de Medeiros Caseli

Indução de léxicos bilíngües e regras para a tradução automática

Helena de Medeiros Caseli

Orientadora: ***Profa. Dra. Maria das Graças Volpe Nunes***

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências - Ciências de Computação e Matemática Computacional.

USP – São Carlos
Abril de 2007

Agradecimentos

Aos meus pais pelo grande amor que têm por mim e por serem meus grandes amigos.

Ao meu irmão, minha outra metade, pela amizade e o exemplo de vida.

Ao Leo, o amor da minha vida, por ser tudo aquilo o que sempre sonhei para um companheiro.

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

À Graça pelos anos de dedicação e orientação e por ter me guiado pela vida de pesquisadora. Ao Mikel pela orientação na Espanha.

Aos colegas do NILC presentes e distantes que nestes 6 anos me ajudaram profissionalmente, em especial a Mônica, Carmen e Élen.

Aos amigos do NILC e, principalmente, ao trio AniAni, Thiago e Lê.

Aos colegas da Espanha e às minhas companheiras de piso Susana e Maloles.

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

À FAPESP e à 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 quatro anos cruzou o meu caminho me apoiando, me incentivando ou simplesmente me ouvindo ...

muito obrigada!

Resumo

A Tradução Automática (TA) – tradução de uma língua natural (fonte) para outra (alvo) por meio de programas de computador – é uma tarefa árdua devido, principalmente, à necessidade de um conhecimento lingüístico aprofundado das duas (ou mais) línguas envolvidas para a construção de recursos, como gramáticas de tradução, dicionários bilíngües etc. A escassez de recursos lingüísticos, e mesmo a dificuldade em produzi-los, geralmente são fatores limitantes na atuação dos sistemas de TA, restringindo-os, por exemplo, quanto ao domínio de aplicação. Neste contexto, diversos métodos vêm sendo propostos com o intuito de gerar, automaticamente, conhecimento lingüístico a partir dos recursos multilíngües e, assim, tornar a construção de tradutores automáticos menos trabalhosa. O projeto ReTraTos, apresentado neste documento, é uma dessas propostas e visa à indução automática de léxicos bilíngües e de regras de tradução a partir de corpora paralelos etiquetados morfossintaticamente e alinhados lexicalmente para os pares de idiomas português–espanhol e português–inglês. O sistema proposto para a indução de regras de tradução apresenta uma abordagem inovadora na qual os exemplos de tradução são divididos em blocos de alinhamento e a indução é realizada para cada bloco, separadamente. Outro fator inovador do sistema de indução é uma filtragem mais elaborada das regras induzidas. Além dos sistemas de indução de léxicos bilíngües e de regras de tradução, implementou-se também um módulo de tradução automática para permitir a validação dos recursos induzidos. Os léxicos bilíngües foram avaliados intrinsecamente e os resultados obtidos estão de acordo com os relatados na literatura para essa área. As regras de tradução foram avaliadas direta e indiretamente por meio do módulo de TA e sua utilização trouxe um ganho na tradução palavra-a-palavra em todos os sentidos (fonte–alvo e alvo–fonte) para a tradução dos idiomas em estudo. As traduções geradas com os recursos induzidos no ReTraTos também foram comparadas às geradas por sistemas comerciais, apresentando melhores resultados para o par de línguas português–espanhol do que para o par português–inglês.

Abstract

Machine Translation (MT) – the translation of a natural (source) language into another (target) by means of computer programs – is a hard task, mainly due to the need of deep linguistic knowledge about the two (or more) languages required to build resources such as translation grammars, bilingual dictionaries, etc. The scarcity of linguistic resources or even the difficulty to build them often limits the use of MT systems, for example, to certain application domains. In this context, several methods have been proposed aiming at generating linguistic knowledge automatically from multilingual resources, so that building translation tools becomes less hard. The ReTraTos project presented in this document is one of these proposals and aims at inducing translation lexicons and transfer rules automatically from PoS-tagged and lexically aligned translation examples for Portuguese–Spanish and Portuguese–English language pairs. The rule induction system brings forth a new approach, in which translation examples are split into alignment blocks and induction is performed for each type of block separately. Another new feature of this system is a more elaborate strategy for filtering the induced rules. Besides the translation lexicon and the transfer rule induction systems, we also implemented a MT module for validating the induced resources. The induced translation lexicons were evaluated intrinsically and the results obtained agree with those reported on the literature. The induced translation rules were evaluated directly and indirectly by the MT module, and improved the word-by-word translation in both directions (source–target and target–source) for the languages under study. The target sentences obtained by the induced resources were also compared to those generated by commercial systems, showing better results for Portuguese–Spanish than for Portuguese–English.

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

TA – Tradução Automática, p. 1

SMT – *Statistical Machine Translation*, p. 1

EBMT – *Example-Based Machine Translation*, p. 1

RBMT – *Rule-Based Machine Translation*, p. 3

ReTraTos – **Re**cursos para a **Tra**dução automática induzidos de **T**extos paralel**os**, p. 4

`pt` – idioma português, p. 4

`en` – idioma inglês, p. 7

`es` – idioma espanhol, p. 7

AM – Aprendizado de Máquina, p. 8

TCR – *Translation Correspondence Ratio*, p. 16

SPM – *Sequential Pattern Mining*, p. 23

PoS – *Part-of-Speech*, p. 29

BS – *Bilingual Similarity*, p. 31

BLD – *Bilingual Lexical Distribution*, p. 31

SF – Sentença Fonte, p. 39

BLEU – *BiLingual Evaluation Understudy*, p. 42

BP – *Brevity Penalty*, p. 43

EM – *Expectation-Maximization*, p. 52

`SABLE` – *Scalable Architecture for Bilingual LExicography*, p. 53

LCSR – *Longest Common Subsequence Ratio*, p. 54

PLN – Processamento de Língua Natural, p. 61

PESA – *Portuguese-English Sentence Alignment*, p. 63

HMM – *Hidden Markov Model*, p. 68

UA – Universidade de Alicante, p. 68

`LIHLA` – *Language-Independent Heuristics Lexical Aligner*, p. 70

AER – *Alignment Error Rate*, p. 73

XML – *Extensible Marckup Language*, p. 83

DTD – *Document Type Definition*, p. 83

SA – sentença alvo, p. 116

Lista de Símbolos

$F_i{}^S$ – fragmentos da língua fonte em um padrão de tradução, p. 18

$F_j{}^T$ – fragmentos da língua alvo em um padrão de tradução, p. 18

C^S – conjunto de fragmentos fonte em um padrão de tradução, p. 18

C^T – conjunto de fragmentos alvo em um padrão de tradução, p. 18

A_f – conjunto de alinhamentos entre os fragmentos fonte e os fragmentos alvo de um padrão de tradução, p. 18

$V_k{}^S$ – variáveis fonte em um padrão de tradução, p. 18

$V_h{}^T$ – variáveis alvo em um padrão de tradução, p. 18

A_v – conjunto de alinhamentos entre as variáveis fonte e alvo em um padrão de tradução, p. 18

E_i – i-ésimo exemplo de tradução, p. 22

$E_i{}^S$ – parte (sentença) fonte do i-ésimo exemplo de tradução, p. 22

$E_i{}^T$ – parte (sentença) alvo do i-ésimo exemplo de tradução, p. 22

S – língua fonte, p. 22

T – língua alvo, p. 22

Q – conjunto de seqüências, p. 23

ϵ – suporte mínimo de uma seqüência para que esta seja considerada um padrão, p. 24

$P_i{}^S$ – um padrão fonte, p. 24

z_k – um item de uma seqüência de itens, p. 24

$P_j{}^T$ – um padrão alvo, p. 24

$P_i{}^S P_j{}^T$ – um padrão bilíngüe, p. 24

q – uma seqüência, p. 24

NP^S – sintagma fonte, p. 27

NP^T – sintagma alvo, p. 27

a_i – um alinhamento lexical, p. 27

s – lado esquerdo ou parte fonte de um exemplo, p. 27

t – lado direito ou parte alvo de um exemplo, p. 27

L_i – conjunto de correspondências lexicais, p. 27

l_i – uma correspondência lexical, p. 28

g_i – uma generalização, p. 33

R_k – conjunto de alinhamentos a_i e correspondências lexicais l_i a partir dos quais a generalização g_k foi gerada, p. 33

G_j – conjunto de generalizações g_i geradas a partir da correspondência lexical l_j, p. 33

R – uma regra de tradução, p. 37

f – freqüência de uma regra, p. 38

B – uma entrada de um léxico bilíngüe, p. 49

w^S – uma palavra na língua S, p. 50

w^T – uma palavra na língua T, p. 50

D – medida de similaridade entre duas palavras, p. 50

qid – identificador de seqüência, p. 101

`Xi_k` – variável que identifica o item fonte i e o valor de seu k-ésimo atributo, p. 109

`Yj_h` – variável que identifica o item alvo j e o valor de seu h-ésimo atributo, p. 109

Sumário

1 Introdução

A Tradução Automática (TA) – tradução de uma língua natural (fonte) para outra (alvo) por meio de programas de computador – é uma tarefa árdua devido, principalmente, à necessidade de um conhecimento lingüístico aprofundado das duas (ou mais) línguas envolvidas para a construção de recursos como gramáticas de tradução, dicionários bilíngües etc. A escassez de recursos lingüísticos, e mesmo a dificuldade em produzi-los, geralmente são fatores limitantes na atuação dos sistemas de TA restringindo-os, por exemplo, quanto ao domínio de aplicação. Por outro lado, com a quantidade cada vez maior de informação disponível em diversas línguas na *web*, faz-se necessária a criação de novas técnicas e recursos capazes de transformar essa abundância de informação multilíngüe em conhecimento lingüístico útil para sistemas de TA.

Para lidar com esse desequilíbrio entre escassez de conhecimento lingüístico e abundância de informação multilíngüe, diversos métodos vêm sendo propostos com o intuito de gerar, automaticamente, conhecimento lingüístico a partir dos recursos multilíngües existentes em abundância e, assim, tornar a construção de tradutores automáticos menos trabalhosa.

Mais especificamente, esses métodos tentam extrair o conhecimento de tradução contido em um *corpus* paralelo alinhado para utilizá-lo na criação de recursos úteis para sistemas de TA. Um *corpus* paralelo alinhado é um conjunto de exemplos (geralmente sentenças) escritos em uma língua fonte acompanhados de suas traduções na língua alvo. Esse conjunto de sentenças paralelas pode estar alinhado lexicalmente, ou seja, cada par de sentenças possui indicações de quais *tokens* (palavras, unidades multipalavras, símbolos de pontuação etc.) da sentença fonte são traduções de quais *tokens* da sentença alvo.

Esses métodos fazem parte do paradigma não-lingüístico de TA (ou TA baseada em *corpus*) o qual engloba as abordagens estatística (*Statistical Machine Translation* ou SMT) e baseada em exemplos (*Example-Based Machine Translation* ou EBMT).

Enquanto as técnicas de SMT usam medidas estatísticas para escolher as estruturas mais prováveis (na língua alvo) de formar a tradução da sentença na língua fonte (a probabilidade da tradução determina a tradução), as técnicas de EBMT empregam reconhecimento de padrões para traduzir partes da sentença fonte fornecida e, assim, determinar a tradução (GÜVENIR & CICEKLI, 1998). Ambas as abordagens possuem limitações, por exemplo, o conhecimento de tradução derivado por uma técnica de SMT, representado em um modelo estatístico, tem a deficiência de não modelar aspectos estruturais e sintáticos da língua; enquanto as técnicas de EBMT têm limitações relacionadas à seleção e ao processamento dos exemplos (KITAMURA, 2004). Até o momento, a comunidade científica não alcançou um consenso a respeito da superioridade de uma ou outra abordagem em cenários irrestritos – considerando-se qualquer par de línguas, tamanho de *corpus* ou outro fator relevante para o processo de TA – e começam a surgir propostas para mesclar "o melhor dos dois mundos" com o intuito de obter melhores sistemas de TA (GROVES & WAY, 2005).

Algumas dessas propostas que seguem a abordagem estatística utilizam os chamados *alignment templates* para melhorar a qualidade da TA. Tais *alignment templates* são generalizações de alinhamentos nos quais palavras são substituídas por classes de palavras geradas com base em estatística (OCH, 1999) ou em informações lingüísticas (SÁNCHEZ-MARTÍNEZ & NEY, 2006) – categorias de classe fechada (artigos, pronomes, conjunções etc.) e categorias dominantes (que propagam a informação de flexão para os itens vizinhos). Além desses, há ainda métodos estatísticos – como (GALLEY et al., 2004), (YAMADA & KNIGHT, 2001) e (GILDEA, 2003) – que utilizam informação sintática para extrair conhecimento aplicando cálculos estatísticos. O método de (GALLEY et al., 2004), por exemplo, diferentemente da maioria dos métodos estatísticos de TA, não gera um modelo estatístico do processo de tradução, mas, sim, regras simbólicas para expressar a relação entre uma árvore sintática na língua alvo e a sentença correspondente na língua fonte.

Dessas duas abordagens de TA baseada em *corpus*, a de maior relevância para o projeto apresentado neste documento é a EBMT, proposta em (NAGAO, 1984) como tradução por analogia. A tradução por analogia reproduz o modo como os humanos realizam a tradução automática desde a fase de aprendizagem – armazenando exemplos reais de sentenças fonte e suas traduções e buscando (inferindo) similaridades e diferenças nesses exemplos – até a fase da tradução propriamente dita – decompondo a sentença fonte em fragmentos menores, traduzindo esses fragmentos separadamente com base no que foi aprendido na fase de aprendizagem e formando a tradução final com a composição dos fragmentos traduzidos.

Algumas vantagens dessa abordagem, citadas em (SOMERS, 1999) e de especial im-

portância para este trabalho são:

- os exemplos são dados reais da língua e, portanto, o uso desses exemplos leva a sistemas que cobrem as construções que realmente ocorrem e ignoram as outras que não ocorrem, reduzindo, assim, a super-geração (geração de construções que não satisfazem a gramática da língua em questão);
- o conhecimento lingüístico do sistema pode ser mais facilmente enriquecido, simplesmente adicionando-se mais exemplos;
- os sistemas de EBMT são dirigidos aos dados e não à teoria e, uma vez que não há gramáticas complexas desenvolvidas por uma equipe de lingüistas, o problema de conflito de regra (no qual uma regra pode contradizer parcial ou totalmente uma outra) e a necessidade de se ter uma visão geral da teoria e de como as regras interagem são menores;
- dependendo do modo como os exemplos são usados é possível que um sistema de EBMT para um novo par de línguas seja rapidamente desenvolvido com base em (apenas) um novo *corpus* paralelo alinhado.

Embora a utilidade dos exemplos de tradução (sentenças paralelas) seja inegavelmente grande, informações sobre as estruturas desses exemplos e as correspondências existentes entre suas partes são, sem dúvida, muito mais relevantes para pesquisas em língua natural (MATSUMOTO et al., 1993). Essas informações – representadas por meio de regras de tradução (ou de transferência) e dicionários (ou léxicos) bilíngües – são utilizadas pelos sistemas de tradução automática baseada em regras (*Rule-Based Machine Translation* ou RBMT) para traduzir (transferir) a representação de uma sentença na língua fonte em uma representação correspondente na língua alvo. Segundo Hutchins (2005), RBMT era o paradigma de TA dominante até a década de 1980 quando a TA baseada em *corpus* ganhou força.

Nesse contexto, nos últimos anos, vários métodos têm sido propostos com o intituto de extrair, de forma automática, as correspondências estruturais, sintáticas ou lexicais dos exemplos alinhados e generalizá-las, quando possível, resultando em uma gramática de tradução (um conjunto de regras de tradução). Porém, de acordo com Hutchins (2005), mesmo que os sistemas de RBMT utilizem bases de dados bilíngües para derivar (total ou parcialmente) suas regras de tradução, isso não os converte em sistemas de EBMT.

Além disso, citando (MARUYAMA & WATANABE, 1992), Hutchins enfatiza que não há uma diferença clara entre exemplos e regras de tradução uma vez que ambos podem

ser processados do mesmo modo e um exemplo de tradução pode ser considerado um caso especial de regra de tradução no qual os nós são itens lexicais e não categorias gramaticais. Assim, Hutchins conclui que sistemas de TA baseados em regras extraídas automaticamente de exemplos de tradução são melhor referidos como sistemas "híbridos" de EBMT e RBMT.

O projeto apresentado neste documento – ReTraTos (**Re**cursos para a **Tra**dução automática induzidos de **T**extos paralel**os**) – propõe a indução de recursos úteis para a TA – regras de tradução e léxicos computacionais bilíngües[1] – a partir de *corpora* paralelos alinhados, por meio de métodos empíricos para minimizar os custos de desenvolvimento. De acordo com a classificação de Hutchins (2005) pode-se dizer que o sistema de TA derivado do ReTraTos é um híbrido de EBMT e RBMT. Além disso, a abordagem adotada para a indução das regras se mostrou inovadora no modo como as regras são buscadas e filtradas. Como resultado deste projeto, vários recursos lingüístico-computacionais foram gerados.

O restante deste capítulo apresenta a motivação (seção 1.1) e os objetivos (seção 1.2) do projeto ReTraTos. A última seção (1.3) apresenta a organização deste documento descrevendo, resumidamente, o que pode ser encontrado nos demais capítulos.

1.1 Motivação

O projeto ReTraTos surge como uma alternativa para o processo árduo de construção de tradutores, uma vez que propõe a indução de regras de tradução e de léxicos bilíngües a partir de *corpora* paralelos alinhados empregando métodos empíricos para minimizar os custos de desenvolvimento.

Além dessa, outra motivação deste projeto está relacionada aos avanços nos estudos de TA para o português (`pt`), incipientes no Brasil (e também em Portugal) frente à demanda enorme por sistemas desse tipo; contrapondo, assim, a escassez de trabalhos acadêmicos (e talvez comerciais) desenvolvidos exclusivamente para o português do Brasil. Vale ressaltar, aqui, que a TA envolvendo o português do Brasil tem ganhado força em pesquisa apenas recentemente, quando projetos mais ambiciosos como o da UNL[2] e o EPT-Web[3] – ambos sistemas que adotam a tradução por interlíngua – se propuseram a levar a cabo a tradução ao nível de um processo completo e robusto. Os resultados, restritos a estudos de caso, não

[1] A partir deste momento o termo léxico bilíngüe será usado para designar léxico computacional bilíngüe dado o contexto deste trabalho.

[2] Informações a respeito do projeto UNL para o português do Brasil podem ser obtidas em: `http://www.nilc.icmc.usp.br/nilc/projects/unl.htm`

[3] Informações a respeito do projeto EPT-Web podem ser obtidas em: `http://www.nilc.icmc.usp.br/nilc/projects/ept-web.htm`

garantem melhor desempenho quando comparado aos sistemas comerciais.

Com o intuito de se determinar o cenário da tradução automática envolvendo o português do Brasil e, assim, traçar os rumos desta pesquisa, fez-se, inicialmente, um levantamento dos trabalhos que apresentam análises dos sistemas de TA existentes para o `pt`. Em (OLIVEIRA Jr. et al., 2000), seis tradutores automáticos inglês-português-inglês[4] foram analisados na tradução de 20 passagens de texto (com uma ou mais sentenças) do jornal brasileiro "Folha de São Paulo" e do jornal norte-americano "*The New York Times*", constatando-se que menos de 50% das saídas geradas pelos sistemas poderiam ser consideradas inteligíveis. Além disso, percebeu-se que essas deficiências não motivaram os desenvolvedores das ferramentas a procurar estratégias alternativas para superá-las, uma vez que os níveis de desempenho se mantêm, quase sempre, os mesmos.

Além dessa análise geral do desempenho dos sistemas, foi realizado um levantamento dos principais problemas encontrados nos níveis lexical, sintático e semântico-pragmático. Desses, os dois primeiros fazem parte do escopo deste trabalho e, por isso, são apresentados em detalhes a seguir.

Os problemas identificados no nível lexical foram: (1) dicionarização das palavras, (2) homônimos, (3) conotações e (4) expressões idiomáticas. Em relação à dicionarização, constatou-se que os problemas mais freqüentes estavam relacionados a nomes próprios e palavras derivadas, como "*Hungary*" e "*Hungarian*". Quanto ao problema de palavras homônimas, bastante freqüente no português, constatou-se que apenas a dicionarização das palavras não foi suficiente para solucioná-lo sendo necessários recursos mais elaborados para desambigüação lexical de sentido. O terceiro problema refere-se ao uso conotativo de palavras em português que, por possuírem um contexto cultural muito específico, não podem ser transferidas de uma língua para outra de maneira direta. Um exemplo desse problema foi a tradução incorreta da expressão, em português, "pegar carona" (no sentido de "tirar proveito de") para "*to hitchhike*", em inglês, uma vez que o sentido, nesse caso, não é o literal. Por fim, os sistemas avaliados apresentaram muitos problemas na tradução de expressões idiomáticas (como "abrir mão de", "ao pé da letra" e os *phrasal verbs* do inglês) nas quais o significado da expressão como um todo não pode ser obtido por meio da composição dos significados das palavras que a formam.

No nível sintático, foram identificados problemas como: (1) concordância (artigo-substantivo ou substantivo-verbo), (2) uso incorreto de tempos verbais, preposições, artigos,

[4]Os sistemas analisados em (OLIVEIRA Jr. et al., 2000) foram: `Translator Pro`, `Alta Vista`, `Intertran`, `GO Translator`, `Tradunet` e `Enterprise Translator Server`.

pronomes ou comparações e (3) ausência de algum componente (preposição, artigo, pronome reflexivo ou conjunção). Constatou-se, ainda, que alguns desses problemas poderiam ser solucionados com a existência de regras de geração para, por exemplo, garantir a concordância entre artigo e substantivo. Muitos outros problemas, no entanto, estão relacionados às diferenças sintáticas entre as duas línguas analisadas (português e inglês), diferenças para as quais as ferramentas de tradução não estão preparadas.

Por fim, os autores do estudo apontam três fatores principais para as deficiências encontradas: (1) a ausência ou a má qualidade dos recursos lingüísticos disponíveis; (2) a suposição errada de que há muita similaridade semântica (praticamente uma correspondência um-para-um) entre o português e o inglês, desconsiderando-se que, em muitos casos, as estruturas semânticas são dependentes de contexto ou de cultura; e (3) a dificuldade de geração de traduções naturais que preservem não apenas a informação da sentença, mas também a forma como essa informação é passada na língua alvo, uma vez que a forma é tão importante quanto o conteúdo propriamente dito.

Em uma outra análise do desempenho de sistemas de TA, apresentada em (FOSSEY et al., 2004), quatro sistemas[5] foram avaliados na tradução de 515 sentenças da primeira página do jornal "*The New York Times*" (em inglês) para o português. Nessa análise, as sentenças foram classificadas em três tipos: gramaticais corretas (sentenças que traduzem de uma forma aceitável o sentido da frase original), gramaticais incorretas (sentenças que obedecem regras gramaticais, mas não obedecem regras semânticas) e agramaticais (sentenças que não possuem nada que as identifique como uma sentença da língua portuguesa). Os resultados dessa análise mostraram que nenhum dos sistemas alcançou um número satisfatório de sentenças consideradas "gramaticais corretas", isso porque, nos quatro sistemas, a somatória das sentenças "gramaticais incorretas" e "agramaticais" sempre ultrapassa 50% do número total de sentenças do *corpus* de teste: `Linguatec e-translation Server` (66,8%), `Intertran` (85,9%), `Systram` (69,1%) e `FreeTranslation` (66%).

Com base nas duas avaliações de sistemas de TA apresentadas anteriormente, e com o intuito de analisar mais profundamente os tipos de erros encontrados na tradução de/para o português, realizou-se uma nova análise do desempenho de sistemas de TA português-inglês-português – `Systran`[6] (ST), `FreeTranslation`[7] (FT) e `TranslatorPro` (TP) – e, também,

[5] Em (Fossey et al., 2004), foram avaliados os sistemas de TA: `Linguatec e-translation Server`, `Intertran`, `Systran` e `FreeTranslation`.

[6] `http://www.systransoft.com`

[7] `http://www.freetranslation.com`

português-espanhol-português – `Universia`[8] e `AutomaticTrans`[9]. O propósito dessa nova análise, realizada no início deste projeto (em 16/08/2004), era apontar as classes de problemas que necessitam de maior atenção por parte dos desenvolvedores dos sistemas de TA para esses idiomas.

Nesse experimento, 20 sentenças em `pt` e suas respectivas traduções para o inglês (`en`) e o espanhol (`es`) foram submetidas aos tradutores, constatando-se que a maioria dos erros encontrados nas traduções, nos dois pares de línguas e nos dois sentidos, foi causada pela tradução incorreta (ou a não tradução) de palavras (erro lexical) ou pelo uso incorreto (ou ausência) de preposições, artigos e tempos verbais, como é apresentado na Tabela 1 (para o par `pt`–`en`) e na Tabela 2 (para o par `pt`–`es`).

Tabela 1: Resumo dos erros encontrados no experimento realizado com pares de sentenças `pt`–`en`

Sistema	`pt`→`en`			`en`→`pt`		
Erro (%)	**ST**	**FT**	**TP**	**ST**	**FT**	**TP**
Lexical	**27,0**	**32,8**	**23,6**	**51,5**	**32,6**	**32,5**
Uso incorreto	**51,1**	**52,3**	**54,7**	**29,1**	**18,1**	**19,1**
Preposições	14,6	23,5	18,2	11,2	8,7	9,6
Artigos	22,6	15,0	16,2	15,7	2,9	1,6
Tempos verbais	2,9	1,2	8,8	1,5	5,1	3,2
Outras categorias	11,0	12,6	11,5	0,7	1,4	4,7
Ausência	**8,0**	**3,4**	**10,2**	**11,2**	**32,6**	**31,7**
Preposições	2,9	1,7	2,7	6,7	13,0	9,5
Artigos	4,4	1,1	6,1	3,7	17,4	20,6
Outras categorias	0,7	0,6	1,4	0,8	2,2	1,6
Outros	**13,9**	**11,5**	**11,5**	**8,2**	**16,7**	**16,7**

De acordo com os dados da Tabela 1 é possível notar que todos os tradutores automáticos analisados no sentido `pt`→`en` apresentaram mais de 50% de erro no uso, principalmente, de preposições (ST = 14,6%, FT = 23,5% e TP = 18,2%) e artigos (ST = 22,6%, FT = 15,0% e TP = 16,2%). No sentido `en`→`pt`, a maior ocorrência de erro está na tradução incorreta (ou na não tradução) de palavras, ou seja, erro do tipo lexical (ST = 51,5%, FT = 32,6% e TP = 32,5%), porém os erros de uso incorreto ou ausência representam, juntos, mais de 40% do total, em todos os sistemas (ST = 40,3%, FT = 50,7% e TP = 50,8%).

Entre os outros tipos de erros encontrados (indicados na Tabela 1 e na Tabela 2 com a denominação "Outros") estão: ordem incorreta das palavras, concordância de gênero e número (entre substantivo e artigo, por exemplo) etc.

[8] `http://tradutor.universia.net/pt/`
[9] `https://www.automatictrans.es`

Tabela 2: Resumo dos erros encontrados no experimento realizado com pares de sentenças pt–es

Sistema	pt→es		es→pt	
Erro (%)	Universia	AutomaticTrans	Universia	AutomaticTrans
Lexical	**19,1**	**21,4**	**34,8**	**19,5**
Uso incorreto	**38,1**	**40,5**	**39,1**	**50,0**
Preposições	15,9	21,4	23,9	25,0
Artigos	4,8	4,8	15,2	25,0
Tempos verbais	7,9	11,9	0	0
Outras categorias	9,5	2,4	0	0
Ausência	**33,3**	**33,4**	**15,2**	**22,2**
Preposições	12,7	16,7	8,7	16,7
Artigos	20,6	16,7	6,5	5,5
Outros	**9,5**	**4,7**	**10,9**	**8,3**

Na análise dos tradutores para o par pt–es, constatou-se que, no sentido pt→es, mais de 38,0% dos erros estão relacionados, principalmente, ao uso incorreto de preposições (Universia = 15,9% e AutomaticTrans = 21,4%) e tempos verbais (Universia = 7,9% e AutomaticTrans = 11,9%). Além disso, os erros de ausência nesse sentido foram bastante freqüentes (mais de 33,0%), principalmente, no que diz respeito a preposições (Universia = 12,7% e AutomaticTrans = 16,7%) e artigos (Universia = 20,6% e AutomaticTrans = 16,7%).

No sentido es→pt, a porcentagem de erro de uso também é a maior (mais de 39,0%) em preposições (Universia = 23,9% e AutomaticTrans = 25,0%) e artigos (Universia = 15,2% e AutomaticTrans = 25,0%). Os erros de ausência, nesse sentido, são um pouco menores do que no sentido contrário, porém, ainda se mantêm altos, especialmente, com preposições (Universia = 8,7% e AutomaticTrans = 16,7%) e artigos (Universia = 6,5% e AutomaticTrans = 5,5%).

Comparando-se a quantidade de erros, por sentença, nos pares pt–en e pt–es é possível concluir que existem, aproximadamente e em média, 8 erros/sentença na tradução pt→en; 7 na tradução en→pt; 3 na tradução pt→es e 2 na tradução es→pt. Assim, o número de erros no par pt–en é maior (mais do que o dobro) do que no par pt–es; o que pode ser justificado pela maior proximidade do português com o espanhol do que com o inglês.

Com base em todas as análises apresentadas anteriormente é possível notar que há, ainda, muito a ser melhorado no que diz respeito à tradução envolvendo o pt. Sendo assim, o projeto ReTraTos propõe a aplicação de técnicas de Aprendizado de Máquina (AM) para induzir automaticamente (e em larga escala) os recursos necessários para tentar melhorar o

desempenho dos sistemas de TA; são eles: léxicos bilíngües e regras de tradução. Até onde se sabe este é o primeiro trabalho que utiliza técnicas de AM para a TA do `pt`.

1.2 Objetivos

Embora se saiba que a qualidade da TA comercial atual só foi atingida depois de anos de esforço na criação de regras de tradução codificadas à mão, e que essa qualidade não tem sido igualada nem superada pelos sistemas cuja fonte primária de conhecimento de tradução é derivada de uma base de exemplos criada automaticamente (RICHARDSON et al., 2001), é importante esclarecer que o objetivo do projeto apresentado aqui não é gerar um sistema completo de TA. O objetivo deste projeto é

> **Induzir automaticamente recursos lingüísticos úteis para a TA – na forma de léxicos bilíngües e regras de tradução – visando à diminuição do esforço na construção de tradutores automáticos e à tentativa de solucionar alguns dos problemas encontrados nos sistemas atuais.**

Com o intuito de alcançar esse objetivo principal, têm-se como metas intermediárias:

- Investigar as principais técnicas de indução de regras de tradução propostas na literatura;
- Implementar, adaptar e avaliar as técnicas que se mostrarem mais compatíveis para os pares `pt`–`en` e `pt`–`es` em *corpora* específicos;
- Combinar várias fontes de conhecimento – como alinhadores sentencial e lexical e etiquetadores morfossintáticos – de maneira automática, rápida e compatível;
- Construir um sistema de indução de regras de tradução que seja independente de língua e que tenha como entrada apenas um *corpus* de sentenças paralelas alinhadas lexicalmente e etiquetadas morfossintaticamente;
- Construir um sistema de indução de entradas para um léxico bilíngüe que seja independente de língua e que tenha como entrada apenas um *corpus* de sentenças paralelas alinhadas lexicalmente e etiquetadas morfossintaticamente;
- Induzir automaticamente regras de tradução que sejam legíveis por um humano e, portanto, passíveis de melhorias;

- Induzir automaticamente entradas para léxicos bilíngües;
- Produzir um sistema simples de TA baseado na recombinação das regras de tradução e na consulta aos léxicos bilíngües induzidos automaticamente que receba como entrada a representação de uma sentença na língua fonte e produza como saída a representação de uma sentença correspondente na língua alvo;
- Avaliar o custo e os benefícios da abordagem investigada.

1.3 Organização do texto

O restante deste documento está organizado como se segue.

O Capítulo 2 apresenta uma contextualização da área de indução de regras de tradução, na qual tem-se: (2.1) definição de regra de tradução; (2.2) descrição do processo de indução de regras de tradução e das principais técnicas empregadas em cada etapa do processo; (2.3) breve descrição do processo de tradução automática por meio das regras induzidas; e (2.4) apresentação das metodologias utilizadas na avaliação das regras induzidas bem como os valores levantados, na literatura, em algumas avaliações dos métodos citados.

O Capítulo 3, por sua vez, apresenta uma contextualização sobre a indução de léxicos bilíngües com: (3.1) uma definição de léxicos bilíngües, (3.2) uma breve apresentação de alguns métodos de indução de léxicos bilíngües e (3.3) a descrição das metodologias de avaliação e dos resultados apresentados na literatura.

No Capítulo 4 são apresentadas as tarefas de pré-processamento dos recursos lingüísticos (*corpora* paralelos) utilizados no projeto ReTraTos, bem como as ferramentas computacionais construídas ou adaptadas para desempenhar cada uma delas: (4.1) alinhamento sentencial, (4.2) etiquetação morfossintática e (4.3) alinhamento lexical.

O Capítulo 5, por sua vez, trata do processo de indução de regras de tradução e de léxicos bilíngües no projeto ReTraTos descrevendo (5.1) os formalismos de representação adotados no projeto para: (5.1.1) os exemplos de tradução, (5.1.2) os léxicos bilíngües e (5.1.3) as regras de tradução. Em seguida, são descritos os processos de indução de léxicos bilíngües (5.2), de regras de tradução (5.3) e de TA realizada por meio da recombinação das regras induzidas (5.4).

A avaliação dos recursos induzidos no projeto ReTraTos é o tema do Capítulo 6, no qual descreve-se a metodologia empregada e os resultados obtidos na avaliação: (6.1) dos

léxicos bilíngües e (6.2) das regras de tradução. Por fim, o Capítulo 7 apresenta as conclusões deste trabalho e as várias propostas de trabalhos futuros.

O Apêndice A apresenta os símbolos gramaticais utilizados no projeto ReTraTos para representar as categorias e os traços morfossintáticos dos *tokens* presentes nos exemplos de tradução usados na indução dos recursos lingüísticos.

2 Indução de regras de tradução

Um sistema de TA requer uma grande quantidade de conhecimento de tradução – geralmente armazenado em dicionários bilíngües, bases de exemplos ou modelos estatísticos (MENEZES & RICHARDSON, 2001) – de difícil construção ou manutenção. Contudo, na última década, diversas pesquisas têm se concentrado na aquisição automática desse conhecimento, induzindo-o de *corpora* bilíngües. Nesse contexto, e mesclando estratégias de EBMT e RBMT, estão inseridos os sistemas de indução de regras de tradução.

De acordo com (BROWN, 2001), os sistemas de EBMT lexicalizada, como os de (VEALE & WAY, 1997) e (BROWN, 1999) têm a vantagem de que requerem pouco ou nenhum conhecimento adicional além do texto paralelo que forma a base de exemplos. A desvantagem é que a base de exemplos deve ser relativamente grande para se garantir boa cobertura e permitir a aplicação do sistema de tradução a textos irrestritos. Como grandes bases de exemplos são difíceis ou, para línguas menos usadas, impossíveis de serem obtidas; vários esforços estão sendo empregados com o intuito de reduzir a necessidade de grande quantidade de dados por meio da generalização dos exemplos de tradução transformando-os em padrões ou regras de tradução.[1]

Também na abordagem estatística têm surgido propostas – como (OCH & NEY, 2004) e (SÁNCHEZ-MARTÍNEZ & NEY, 2006) – com o intuito de generalizar o conhecimento presente nos exemplos de tradução transformando-os em *alignment templates*. Nesses métodos, os alinhamentos entre palavras e unidades multipalavras gerados com base em modelos estatísticos são generalizados substituindo-se as palavras por classes mais gerais. Embora haja semelhanças entre *alignment templates* e regras de tradução, estas últimas são conhecidas por possuirem informações lingüísticas não encontradas nos primeiros.

Segundo (KAJI et al., 1992) e (LIU & ZONG, 2004), a utilização de regras de tradução em lugar de exemplos de tradução possui algumas vantagens:

[1] Se o leitor não estiver familiarizado com os termos padrões e regras de tradução, veja a seção 2.1.

- **Maior transparência na tradução** – a tradução é obtida transferindo-se diretamente a parte fonte presente na seqüência de entrada que casa com uma regra de tradução para a parte alvo correspondente;
- **Menor computação** – enquanto os sistemas de EBMT tradicionais requerem uma grande quantidade de computação para processar os exemplos, em um sistema baseado em regras, os exemplos são transformados em estruturas de representação mais simples e apenas a informação útil para a tradução é processada, reduzindo, portanto, a quantidade de computação;
- **Maior unificação do conhecimento** – vários tipos de conhecimento de tradução são extraídos e representados por meio de uma única estrutura de regra de tradução, unificando, assim, o conhecimento;
- **Menor *corpus*** – o tamanho do *corpus* necessário em sistemas de EBMT pode ser reduzido usando regras já que uma regra de tradução pode ser entendida como a generalização de vários exemplos de tradução;
- **Maior probabilidade de "casamento"** – a probabilidade de casamento (do inglês, *matching*) para a sentença fonte com as regras pode ser aumentada quando comparada à probabilidade de casamento com os exemplos.

De modo geral, os sistemas de indução de regras de tradução e de TA baseada nas regras induzidas possuem a arquitetura mostrada na Figura 1, na qual a linha pontilhada indica que a utilização de recursos lingüístico-computacionais (como *parsers*, dicionários bilíngües, etiquetadores etc.) é opcional.

Nessa arquitetura, um *corpus* bilíngüe alinhado, geralmente no nível sentencial, é fornecido como entrada para o módulo de indução. As regras de tradução geradas como saída são posteriormente utilizadas na geração das sentenças alvo correspondentes às sentenças fonte por meio de um módulo de TA/recombinação (aplicação) das regras induzidas.

A parte variável dessa arquitetura está no módulo de indução de regras de tradução. Os sistemas de indução de regras de tradução propostos na literatura variam de acordo com diversos critérios. Um desses critérios é a utilização (ou não) de recursos lingüístico-computacionais no processo de extração das gramáticas de tradução (como indicado pela linha pontilhada da Figura 1). Nos sistemas de EBMT "puros", a única fonte de conhecimento disponível para a indução das regras é o par de textos paralelos alinhados; enquanto que em sistemas mais refinados outros recursos lingüísticos são utilizados em menor ou maior

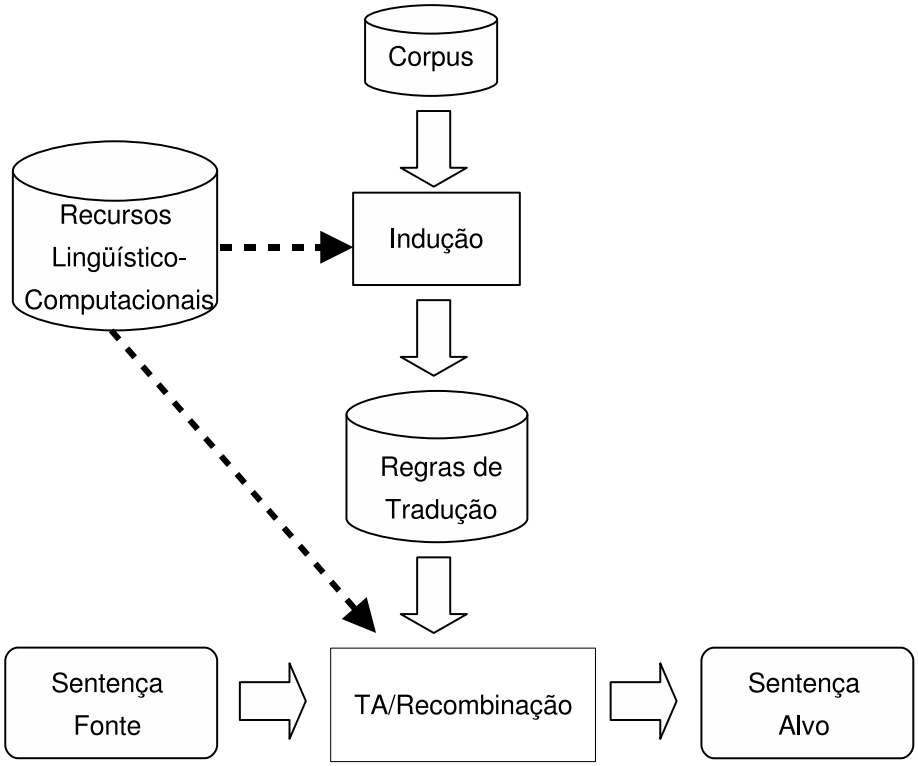


Figura 1: Arquitetura do sistema de indução de regras de tradução e TA/Recombinação (MCTAIT, 2003)

grau. Além disso, os sistemas também variam quanto ao número e à qualidade dos recursos utilizados e quanto ao modo como esse conhecimento é representado, armazenado e usado para tradução (CARL, 2001).

Por fim, tais sistemas variam no modo como tratam os exemplos de tradução: como seqüências não-estruturadas de palavras ou como estruturas resultantes de uma análise sintática (*parsing*) em um ou ambos os lados (fonte e alvo), realizada como um passo prévio à aquisição das regras de tradução (MEYERS et al., 1998).

Embora existam diversos métodos de indução de regras de tradução propostos na literatura com diferentes abordagens, três etapas comuns do processo de indução podem ser identificadas na maioria dos métodos, sendo que apenas a primeira delas varia de acordo com a realização ou não de análise sintática em uma ou ambas as línguas. De modo geral, o processo de indução de regras de tradução a partir de textos paralelos alinhados pode ser dividido em: (1) identificação de padrões (em sistemas que não realizam a análise sintática) ou alinhamento de árvores sintáticas (em sistemas que analisam sintaticamente as sentenças paralelas), (2) geração das regras de tradução e (3) filtragem ou ordenação das regras geradas.

A identificação dos padrões pode ser realizada, por exemplo, por meio de reconheci-

mento de seqüências repetidas de palavras em dois pares de exemplos (com técnicas de reconhecimento de padrões) ou por meio de correspondências lexicais existentes em um léxico bilíngüe ou alinhamento lexical. O alinhamento das árvores sintáticas engloba o alinhamento dos nós folha com base em alinhamentos lexicais que foram extraídos de um léxico bilíngüe, gerados previamente (manual ou automaticamente) ou determinados estatisticamente durante o processo de alinhamento. Em seguida, os nós restantes são alinhados com base em regras pré-definidas, probabilidades de casamento de um nó fonte com um nó alvo, programação dinâmica etc.

A segunda etapa – geração das regras de tradução – baseia-se na generalização dos padrões ou alinhamentos definidos na etapa anterior. No caso dos padrões, estes são agrupados e generalizados (partes do padrão são substituídas por variáveis) considerando-se apenas a existência de similaridades, de diferenças ou de ambas (similaridades e diferenças). No caso dos alinhamentos, as regras podem ser geradas de diversas maneiras que variam de acordo com o método e vão desde a extração de padrões correspondentes a subárvores das árvores alinhadas até uma estratégia oposta de expansão dos nós alinhados para a inserção de contexto no padrão que engloba este nó.

A terceira e última etapa, presente em apenas alguns dos métodos estudados, engloba a filtragem das regras, por exemplo, para a eliminação de ambigüidades; ou a ordenação dessas regras de acordo com algum critério como freqüência de ocorrência, especificidade etc. Além da filtragem das regras propriamente ditas, em alguns métodos, como (IMAMURA et al., 2004), um passo prévio à indução garante a filtragem dos exemplos a serem utilizados no processo. Esse método usa uma medida denominada *Translation Correspondence Ratio* (ou TCR)[2] para filtrar os exemplos bilíngües calculando sua literalidade (do inglês, *literalness*) e, assim, determinar um conjunto apropriado de sentenças a partir do qual as regras serão extraídas.

Embora seja grande a variedade de estratégias empregadas pelos métodos de indução de regras de tradução, Menezes & Richardson (2001) apontam algumas características desejáveis para todos esses métodos:

- as regras devem ser induzidas com uma alta precisão;
- o método deve ser robusto em relação a erros introduzidos por recursos computaci-

[2] A TCR de um par de sentenças paralelas é calculada como $TCR = \frac{2L}{T_s+T_t}$ em que T_s é o número de palavras fonte e T_t o número de palavras alvo dessas sentenças encontradas em um léxico bilíngüe, e L o número de ligações entre as palavras fonte e alvo. Assim, TCR denota a porção de palavras traduzidas diretamente entre as palavras que deveriam ser traduzidas (IMAMURA et al., 2004).

onais de análise sintática e de alinhamento sentencial/lexical, e a erros (ortográficos, gramaticais ou de tradução) intrínsecos do *corpus*;

- as regras produzidas devem oferecer contexto suficiente para permitir que o sistema de TA que as utiliza escolha a melhor opção de tradução em um determinado momento.

A seguir, na seção 2.1, a definição de uma regra de tradução é apresentada com base nos diferentes tipos de exemplos de tradução – exemplos literais, padrões de tradução e regras de tradução – especificados na literatura, acompanhados dos formalismos utilizados para representá-los. A seção 2.2 apresenta as técnicas empregadas pelos principais métodos de indução de regras de tradução em cada uma das etapas do processo de indução: (2.2.1) identificação de padrões, (2.2.2) alinhamento de árvores sintáticas, (2.2.3) geração das regras de tradução e (2.2.4) filtragem e ordenação das regras de tradução.

A seção 2.3 apresenta uma breve descrição sobre o processo de TA a partir das regras de tradução induzidas automaticamente. Por fim, na seção 2.4, tem-se uma visão geral das metodologias de avaliação empregadas atualmente para verificar a qualidade das regras induzidas.

2.1 Regras de tradução

Segundo Furuse & Iida (1992), os exemplos de tradução podem ser classificados em três níveis diferentes: (1) exemplos literais (*string-level*), (2) exemplos de padrões (*pattern-level*), nos quais algumas palavras são substituídas por variáveis (V)[3], e (3) exemplos gramaticais (*grammar-level*) expressos em termos de categorias gramaticais. Nesse último nível, por exemplo, as variáveis V_1, V_2 e V_3 no exemplo apresentado a seguir correspondem a substantivos e só poderão ser substituídas por palavras dessa categoria gramatical.

A seguir, são apresentados exemplos de cada um desses três níveis de exemplos de tradução:

(1) I'm hungry ↔ Eu estou com fome

(2) May I speak to V ↔ Poderia falar com V

[3]Padrões de tradução (ou *translation template*), segundo (LIU & ZONG, 2004), são a generalização de exemplos bilíngües alinhados e são pares bilíngües de tradução nos quais os lemas, morfemas, palavras, seqüências de palavras ou sintagmas correspondentes são substituídos por variáveis.

(3) V_1 V_2 for V_3 $\leftrightarrow$ V_2 de V_1 para V_3
V_1=application/inscrição, V_2=form/formulário, V_3 = participation/participação

Como já mencionado no capítulo anterior, embora a utilidade de exemplos literais de sentenças paralelas (tipo de exemplo de tradução apresentado em (1)) seja inegável, informações sobre as estruturas das sentenças e as correspondências (alinhamentos) existentes entre elas são, sem dúvida, muito mais relevantes para as pesquisas em língua natural (MATSUMOTO et al., 1993). Por isso, diversos sistemas foram propostos, nos últimos anos, para a indução de padrões ou regras de tradução (tipos (2) e (3) apresentados anteriormente).

Um padrão de tradução, segundo McTait (2003), pode ser definido formalmente como uma quádrupla $<C^S, C^T, A_f, A_v>$, onde os fragmentos na língua fonte (${F_i}^S$) e alvo (${F_j}^T$) são armazenados, respectivamente, em C^S e C^T, com os alinhamentos entre eles definidos em A_f. Os fragmentos fonte e alvo são separados por variáveis (${V_k}^S$ ou ${V_h}^T$) cujos alinhamentos estão indicados em A_v. Em (2.1) tem-se um exemplo genérico de um padrão de tradução com esse formato.

$$ {F_1}^S\ {V_1}^S\ {F_2}^S\ {V_2}^S\ \ldots\ {F_n}^S\ {V_n}^S \leftrightarrow {F_1}^T\ {V_1}^T\ {F_2}^T\ {V_2}^T\ \ldots\ {F_m}^T\ {V_m}^T \tag{2.1} $$

Por exemplo, um padrão de tradução extraído para os exemplos inglês–espanhol em (4), é apresentado em (5) (MCTAIT & TRUJILLO, 1999). Nesse caso, *gave* e *up* são fragmentos na língua fonte que correspondem ao fragmento na língua alvo *abandonó*, ou seja, esses fragmentos estão alinhados e o alinhamento entre eles é especificado em A_f. As variáveis também se alinham entre si, como especificado em A_v.

(4) The Commission gave the plan up $\leftrightarrow$ La Comisión abandonó el plan
Our Government gave all laws up $\leftrightarrow$ Nuestro Gobierno abandonó todas las leyes

(5) ${V_1}^S\ {F_1}^S\ {V_2}^S\ {F_2}^S \leftrightarrow {V_1}^T\ {F_1}^T\ {V_2}^T$
${F_1}^S$ = {gave}, ${F_2}^S$ = {up} e ${F_1}^T$ = {abandonó}
A_f ={(${F_1}^S$,${F_2}^S$):${F_1}^T$}
${V_1}^S$ = {The Commision,Our Government}, ${V_2}^S$ = {the plan,all laws}, ${V_1}^T$ = {La Comisión,Nuestro Gobierno} e ${V_2}^T$ = {el plan,todas las leyes}
$A_v = \{{V_1}^S : {V_1}^T, {V_2}^S : {V_2}^T\}$

Os padrões de tradução podem, ainda, conter informações morfossintáticas como os padrões apresentados em (7) gerados a partir dos pares de sentenças inglês–turco em (6)

(GÜVENIR & CICEKLI, 1998).

(6) I give+PAST the book ↔ kitap+ACC ver+PAST+1SG
You give+PAST the pencil ↔ kurşun kalem+ACC ver+PAST+2SG

(7) $V_1{}^S\ F_1{}^S\ V_2{}^S \leftrightarrow V_1{}^T\ F_1{}^T\ V_2{}^T$
$F_1{}^S$ = {give+PAST the} e $F_1{}^T$ = {+ACC ver+PAST}
A_f ={$F_1{}^S$:$F_1{}^T$}
$V_1{}^S$ = {I,You}, $V_2{}^S$ = {book,pencil}, $V_1{}^T$ = {kitap,kurşun kalem} e $V_2{}^T$ = {+1SG,+2SG}
$A_v = \{V_1{}^S : V_2{}^T, V_2{}^S : V_1{}^T\}$

Como se pode perceber, conforme se caminha do primeiro nível de exemplos de tradução – os exemplos literais – para o último – as regras de tradução – cresce a quantidade e a complexidade das informações representadas. Assim, as regras de tradução podem ser compostas por informações mais complexas, como as especificadas no formalismo utilizado em (LAVIE et al., 2004) para um método de indução de regras de tradução que realiza análise sintática. Uma regra de tradução, segundo esse formalismo, possui as seguintes informações (veja exemplo na Figura 2 para o par de línguas inglês–hindi[4]):

- **Informação de tipo** – identifica o tipo de uma regra e, na maioria dos casos, corresponde ao tipo de um constituinte sintático – regras de sentença são do tipo `S`, regras de sintagmas nominais (*noun phrases*), do tipo `NP` e assim por diante. O formalismo também permite que a informação de tipo seja diferente nas línguas fonte e alvo;

- **Informação morfossintática** – lista os componentes de uma regra (categorias lexicais, itens lexicais etc.) tanto para a língua fonte quanto para a língua alvo;

- **Alinhamentos** – especificam como o conjunto de componentes na língua fonte se alinha com (transfere para) o conjunto de componentes na língua alvo. Além do tradicional alinhamento $1:1$, alinhamentos do tipo $n:0$ (omissões) e $n:m$, com $n, m > 1$ (alinhamentos de multipalavras), também são possíveis;

- **Restrições do lado fonte** – fornecem informações sobre os atributos e seus respectivos valores na sentença da língua fonte. Essas restrições são usadas para restringir a aplicação de uma regra de tradução a uma dada sentença fonte de entrada;

[4]Um dos idiomas da Índia.

- **Restrições do lado alvo** – são similares às restrições do lado fonte, mas se referem à língua alvo. Essas restrições são utilizadas para guiar e restringir a geração da sentença alvo correspondente à sentença fonte fornecida;
- **Restrições de ambos os lados** – informam quais valores deverão ser inseridos, na geração da sentença alvo, para substituir os valores presentes na sentença fonte.

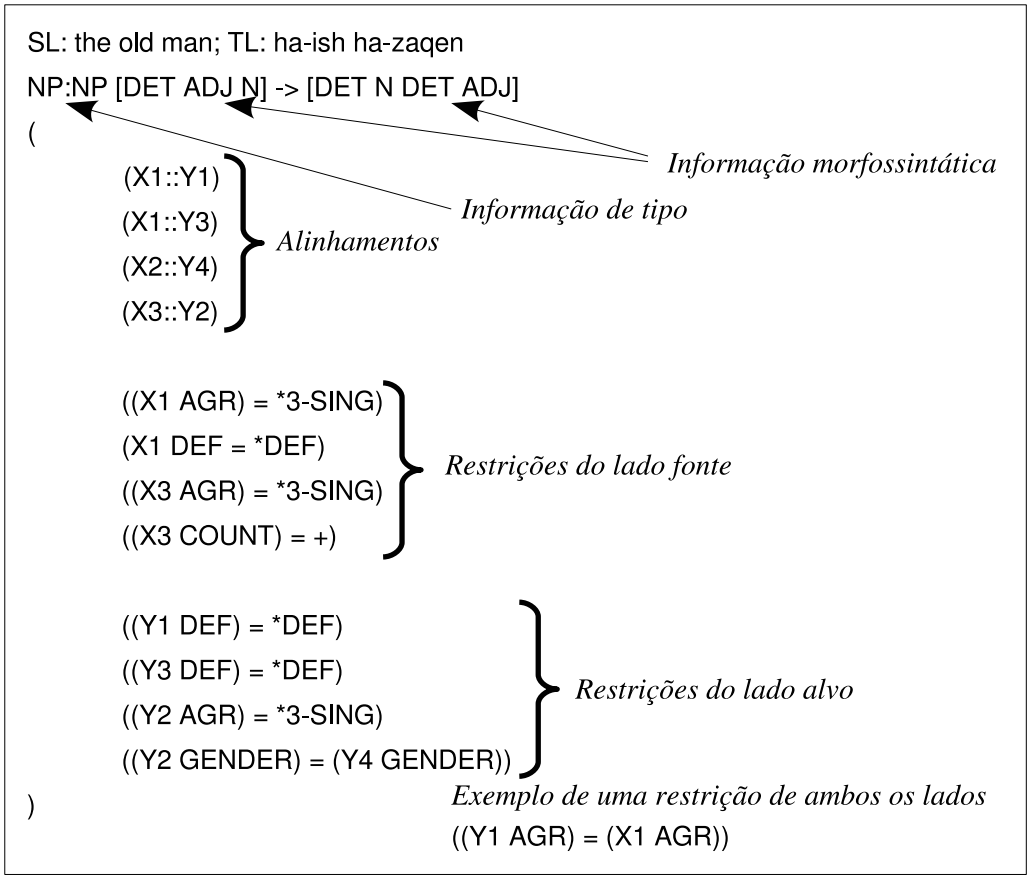


Figura 2: Exemplo de um formalismo de representação de regras de tradução inglês–hindi (LAVIE et al., 2004)

Tal formalismo é capaz de lidar com uma variedade de divergências de tradução como: mudanças nas relações gramaticais em que, por exemplo, um objeto na língua fonte é expresso como sujeito na língua alvo; mudanças estruturais em que, por exemplo, um sintagma nominal se transforma em um sintagma preposicional em outra língua; etc. (CARBONELL et al., 2002).

Outro formalismo de representação de uma regra de tradução (agora para o par coreano–inglês), utilizado também por um método que realiza análise sintática, é apresentado na Figura 3. Esse formalismo engloba a noção de dependência sintática e identifica as variáveis pelo uso do caractere "$" prefixado. Além disso, cada regra é acompanhada de

uma pontuação baseada em *log-likelihood* (MANNING & SCHUTZE, 1999) e calculada com referência às sentenças do *corpus* de treinamento.

```
(a)
@KOREAN:
        {po} [class=vbma] (
          s1 $X [ppca={reul}]
        )
@ENGLISH:
        look [class=verb] (
          attr at [class=preposition] (
            ii $X
          )
        )
@-2xLOG_LIKELIHOOD: 12.77
```

```
(b)
@KOREAN:
        $X [class=vbma ente={ra}]
@ENGLISH:
        $X [class=verb mod=imp]
@-2xLOG_LIKELIHOOD: 33.37
```

Figura 3: Outro exemplo de formalismo de representação de regras de tradução coreano–inglês (LAVOIE et al., 2001)

As regras (a e b) da Figura 3 podem ser usadas para transferir a representação sintática da sentença em coreano *ci-to-reul po-ra* para a representação sintática da sentença em inglês *look at the map*, sendo que a primeira (a) lexicaliza o predicado em inglês e insere a preposição correspondente, enquanto a segunda (b) insere o atributo de imperativo inglês.

Com base em tudo no que foi apresentado nesta seção e considerando-se que as regras de tradução são padrões de tradução com mais informações, de agora em diante o termo "regra de tradução" será usado, neste documento, para se referir tanto a regras quanto a padrões de tradução. Sendo assim, no contexto deste projeto, uma regra de tradução pode ser entendida como a generalização de sentenças que são traduções umas das outras, possuindo o seguinte formato:

$$A \rightarrow B \tag{2.2}$$

em que A é um conjunto de *tokens* ou variáveis derivadas do texto fonte (podendo conter todas as informações apresentadas na Figura 2 e até mesmo outras que se julgarem necessárias) e B, um conjunto semelhante derivado do texto alvo.

O símbolo $\rightarrow$ em (2.2) indica que as regras são unidirecionais no sentindo fonte para alvo, ou seja, as correspondências entre um conjunto de palavras ou variáveis na língua fonte

e um conjunto semelhante na língua alvo não são sempre válidas no sentido inverso (da língua alvo para a língua fonte). A bidirecionalidade (↔) das regras de tradução é uma característica desejada, porém não encontrada em muitos métodos de indução.

Contudo, considerando-se que os exemplos de tradução são bidirecionais, o processo de indução pode ser aplicado nos dois sentidos (fonte→alvo e alvo→fonte) obtendo-se regras de tradução uni ou bidirecionais (resultado da intersecção entre os dois sentidos) que formariam a gramática de tradução final.

2.2 Etapas do processo de indução de regras de tradução

Como mencionado no início deste capítulo, a maioria dos métodos de indução de regras de tradução possuem algumas etapas comuns apresentadas no diagrama da Figura 4 (correspondente ao módulo de indução apresentado na Figura 1, agora, em detalhes). Dado um *corpus* com exemplos de tradução, o processo de indução tem início com o alinhamento de árvores sintáticas ou a identificação de padrões – de acordo com a realização ou não da análise sintática dos exemplos –, em seguida as regras são geradas e, por fim, opcionalmente (por isso esta etapa está representada com linha pontilhada na Figura 4), essas regras são filtradas ou ordenadas resultando em um conjunto de regras de tradução induzidas automaticamente.

As próximas subseções apresentam, resumidamente, algumas das técnicas empregadas em cada uma das etapas da Figura 4: (2.2.1) identificação de padrões, (2.2.2) alinhamento de árvores sintáticas, (2.2.3) geração das regras de tradução e (2.2.4) filtragem ou ordenação das regras geradas.

2.2.1 Identificação de padrões

Os métodos que empregam técnicas de identificação de padrões partem de um princípio de analogia, o qual estabelece que: dados dois pares de exemplos de tradução, se os exemplos fonte apresentam alguma similaridade, então suas traduções na língua alvo também devem possuir partes similares que correspondam às traduções das similaridades fonte; além disso, as partes diferentes restantes nos exemplos fonte também devem corresponder às diferenças nos exemplos alvo.

Considere que um exemplo de tradução E_i: ${E_i}^S \leftrightarrow {E_i}^T$ é composto por um par de sentenças ${E_i}^S$ e ${E_i}^T$ que são traduções mútuas e estão escritas nas línguas fonte (S) e alvo (T), respectivamente. Dado um par de exemplos de tradução $E_1 : {E_1}^S \leftrightarrow {E_1}^T$ e $E_2 : {E_2}^S \leftrightarrow {E_2}^T$,

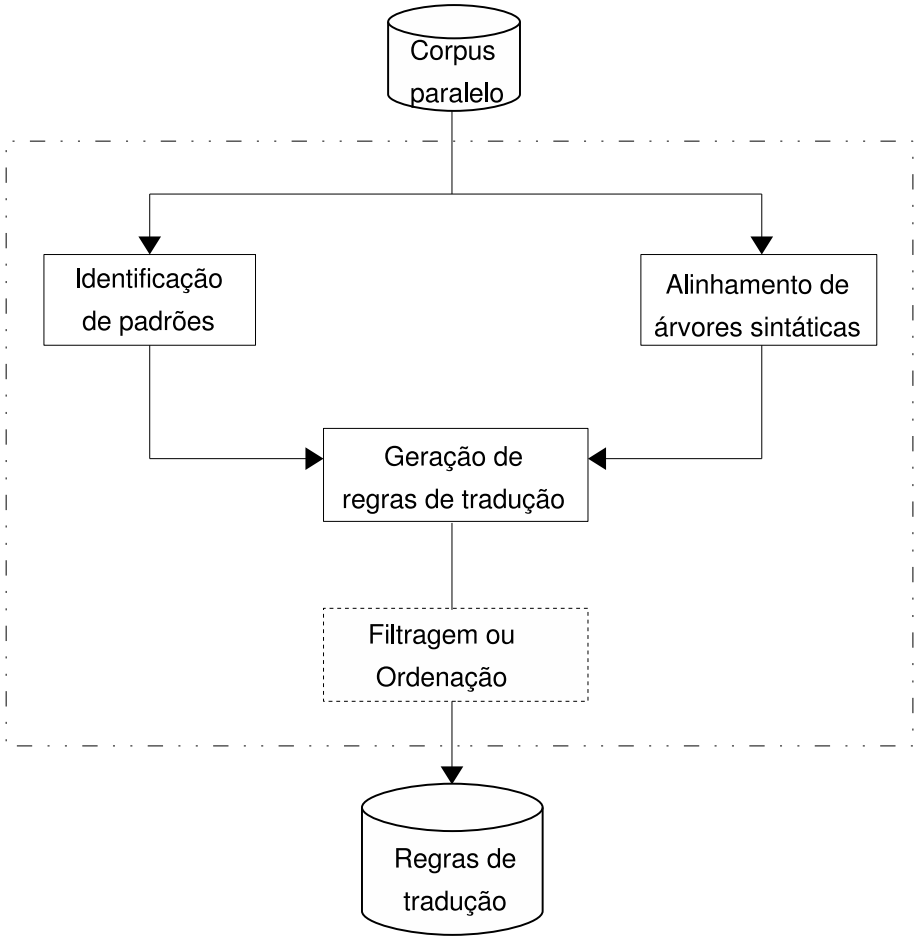


Figura 4: Fluxo de etapas de um método de indução de regras de tradução (visão detalhada do módulo de indução apresentado na Figura 1)

na etapa de identificação de padrões, os métodos tentam identificar seqüências de palavras comuns (padrões) em E_1 e E_2 buscando similaridades entre esses exemplos – (YAMAMOTO et al., 2003), (GÜVENIR & CICEKLI, 1998; CICEKLI & GÜVENIR, 2001) e (MCTAIT, 2003) – ou utilizando regras de transferência lexical geradas previamente com base em um alinhamento lexical ou em um léxico bilíngüe – (BROWN, 2001).

No processo de identificação de padrões realizado em (YAMAMOTO et al., 2003) para inglês–japonês, inicialmente, cada par de sentenças paralelas passa por etapas de pré-processamento separadas em cada língua, como segmentação de palavras e etiquetação morfossintática. Em seguida, as seqüências monolíngues são concatenadas em uma única seqüência bilíngüe e a coleção de seqüências bilíngües se torna uma base de dados de seqüências Q. Uma única execução da técnica *Sequential Pattern Mining* (SPM) identifica e conta os padrões candidatos a tradução – rígidos e com lacunas (*gaps*), alguns sobrepostos – na base de dados de seqüências bilíngües.

Todas as subseqüências fonte que satisfazem o suporte[5] mínimo ϵ são geradas e indicadas por ${P_i}^S$. Por exemplo, considerando-se a seqüência $z_1 z_2 \ldots z_n$ em que z_k é um item e k é a posição que ele ocupa na seqüência, exemplos de subseqüências geradas a partir dessa seqüência são: z_1, $z_1 z_2$, $z_1 z_3, \ldots$. De maneira semelhante, todas as subseqüências alvo e bilíngües com suporte maior ou igual a ϵ são geradas e indicadas por ${P_j}^T$ e ${P_i}^S {P_j}^T$, respectivamente. É importante citar que, para todo padrão bilíngüe ${P_i}^S {P_j}^T$, os padrões fonte e alvo correspondentes (${P_i}^S$ e ${P_j}^T$) que o constituem são sempre reconhecidos e contados.

Para realizar SPM, Yamamoto et al. (2003) utilizam o algoritmo `PrefixSpan` (PEI et al., 2001, 2004) cuja idéia geral é dividir a base de dados de seqüências por prefixo freqüente e aumentar os padrões priorizando a profundidade (*depth-first*).

No método proposto por Cicekli & Güvenir (1998, 2001), as partes iguais e diferentes em um par de exemplos de tradução (E_1, E_2) são identificadas e heurísticas são aplicadas para determinar as correspondências entre as diferenças (GÜVENIR & CICEKLI, 1998) ou entre as similaridades e as diferenças (CICEKLI & GÜVENIR, 2001). Nesse processo, apenas pares de exemplos de tradução que possuem similaridades são processados.

Por exemplo, considere os exemplos de tradução inglês–turco apresentados em (8) cujas similaridades estão sublinhadas. Nesse caso, para determinar as correspondências entre as partes diferentes, aplica-se a seguinte heurística: quando houver apenas uma diferença em cada um dos lados, a correspondência é direta, mas quando o número de diferenças (n) em ambos os lados for maior que 1 (como é o caso em (8), em que $n = 2$) o algoritmo identifica novos padrões apenas se pelo menos $n - 1$ diferenças já foram aprendidas. Além disso, se o número de diferenças nos dois lados for distinto, tenta-se igualá-los separando as diferenças em vários morfemas (se houver mais de um).

Assim, considerando-se que em passos prévios do algoritmo os padrões em (9) já foram aprendidos; a partir das similaridades identificadas em (8) os dois novos padrões apresentados em (10) poderiam ser aprendidos. O processo de identificação se repete até que nenhum novo padrão possa ser aprendido.

(8) E_1 : I give+PAST the book ↔ kitap +ACC ver+PAST+1SG
E_2 : You give+PAST the pencil ↔ kurşun kalem +ACC ver+PAST+2SG

(9) I ↔ +1SG
You ↔ +2SG

[5] O suporte de uma seqüência q em uma base de dados de seqüências Q é a freqüência de ocorrência de q em Q (YAMAMOTO et al., 2003).

(10) book ↔ kitap
pencil ↔ kurşun kalem

Seguindo o mesmo princípio de analogia, o algoritmo de aprendizado de máquina apresentado em (MCTAIT, 2003) se baseia no princípio de distribuições similares de palavras (co-ocorrência e limites de freqüência). Esse princípio pressupõe que palavras na língua fonte e na língua alvo que co-ocorrem em pelo menos dois pares de sentenças de um *corpus* bilíngüe são prováveis de serem traduções umas das outras. Dessa maneira, os padrões são identificados em duas fases: fase monolíngüe e fase bilíngüe.

A fase monolíngüe é aplicada, independentemente, nas sentenças fonte e alvo dos exemplos de tradução. Nela, os itens lexicais (*tokens*) que ocorrem em pelo menos duas sentenças são armazenados juntamente com uma identificação das sentenças nas quais eles ocorrem. Uma estrutura de árvore (colocação) é formada com as possíveis combinações dos itens lexicais com um tamanho crescente e uma freqüência decrescente garantindo, assim, que a maior quantidade de informação esteja presente nas folhas as quais são coletadas no final dessa fase (as folhas oferecem mais contexto e, conseqüentemente, menor possibilidade de ambigüidade).

Por exemplo, considerando-se o par de sentenças inglês–espanhol apresentado em (11), os itens lexicais recuperados e as colocações geradas para esses itens lexicais são apresentados, respectivamente, em (12) e (13). As colocações são formadas quando há intersecção de pelo menos dois identificadores de sentenças nos itens recuperados, como é o caso de *gave up* em (13).

(11) E_1 : The Commission gave the plan up ↔ La Comisión abandonó el plan
E_2 : Our Government gave all laws up ↔ Nuestro Gobierno abandonó todas las leyes

(12) (gave) [1,2], (up) [1,2]
(abandonó) [1,2]

(13) (gave)(up) [1,2]
(abandonó) [1,2]

Na fase bilíngüe, os padrões bilíngües são obtidos usando o critério simples de co-ocorrência no qual colocações fonte e alvo com exatamente os mesmos identificadores de sentenças são consideradas traduções mútuas, ou seja, padrões bilíngües (por exemplo, as colocações fonte e alvo em (13)).

De maneira similar aos métodos citados anteriormente, o método de indução proposto em (BROWN, 2001) também se baseia no fato de que quando dois pares de sentenças no *corpus* têm alguma parte em comum, mas diferem em alguma outra, as partes similares e diferentes correspondem a algum constituinte (sintagma nominal ou preposicional) coerente. Porém, diferentemente dos métodos apresentados até então, o algoritmo utiliza um léxico bilíngüe para determinar as correspondências entre as palavras em cada par de exemplos.

Assim, são processados os pares de exemplos de tradução com apenas uma diferença no lado fonte.

$E_2{}^S : I_1 D_2 I_2$

Para cada um desses pares de exemplos de tradução é criado um mapeamento bilíngüe com base em um léxico bilíngüe com o intuito de descartar aquelas diferenças que não parecem casar entre as sentenças fonte e alvo. Em seguida, buscam-se todos os pares de exemplos que compartilham as n primeiras palavras na língua fonte e, para cada seqüência encontrada, cria-se um subcorpus. Com base nos exemplos em cada subcorpus inicia-se uma busca pelos pares de exemplos que compartilham as mesmas m últimas palavras na língua fonte. Por fim, realiza-se uma comparação par-a-par com os exemplos desse último subcorpus adicionando as partes diferentes a uma nova classe de equivalência e, se forem suficientemente grandes, também ao conjunto de exemplos de tradução como novos (porém menores) exemplos. Os prefixos e sufixos comuns em cada par de exemplos de tradução também são adicionados ao *corpus* como dois exemplos adicionais, se forem suficientemente grandes (pelo menos duas palavras cada).

Na próxima etapa do processo de indução, as classes de equivalência que agrupam as diferenças serão usadas para generalizar os exemplos de tradução e alguns desses exemplos generalizados serão inseridos na base de exemplos. A etapa de identificação de padrões, então, se repetirá buscando padrões na base de exemplos atualizada e criando novas classes de equivalência. Esse ciclo identificação-generalização se repete até que nenhuma outra classe de equivalência possa ser gerada ou um número máximo de iterações seja alcançado.

2.2.2 Alinhamento de árvores sintáticas

Muitos dos métodos de indução de regras de tradução propostos na literatura realizam a análise sintática das sentenças nas línguas fonte e alvo ou, às vezes, em apenas uma delas. Essa análise é efetuada de maneira automática por *parsers* específicos para os idiomas envolvidos (com ou sem treinamento prévio no domínio em questão) e pode ser seguida de

uma verificação manual para a correção de possíveis erros. Com essa análise, os métodos de indução de regras de tradução podem obter, além das correspondências lexicais, regras estruturais.

A primeira etapa dos métodos de indução de regras de tradução que realizam análise sintática é a de alinhamento das árvores geradas. Essa etapa, na verdade, pode ser subdividida em dois passos nos quais, primeiro, é realizado um alinhamento dos nós folhas das árvores com base em alinhamentos lexicais extraídos de um léxico bilíngüe, gerados previamente (manual ou automaticamente) ou calculados com base em estatística. Em seguida, os nós restantes são alinhados considerando-se, por exemplo, regras de composição dos nós definidas previamente, probabilidades de casamento de um nó fonte com um nó alvo, programação dinâmica etc.

Em (KAJI et al., 1992), após montar as tabelas de análise sintática das sentenças fonte e alvo, os pares de palavras lexicais (apenas *content words*) correspondentes presentes nessas tabelas são identificados de acordo com um léxico bilíngüe. Em seguida, as tabelas de análise sintática das duas sentenças são percorridas de baixo para cima (*bottom-up*) em busca de sintagmas (*phrases*) correspondentes. Para cada sintagma NP^S na tabela de análise fonte, busca-se o sintagma NP^T na tabela de análise da sentença alvo, que inclua uma contra-parte para cada palavra em NP^S, mas nenhuma para palavras fora de NP^S. Se apenas um NP^T for encontrado, NP^S e NP^T são associados; porém, se houver mais de um NP^T candidato o desempate é resolvido considerando-se a correspondência de tamanhos: sintagmas menores com menores, sintagmas maiores com maiores.

O algoritmo para a extração de gramáticas de tradução probabilísticas proposto em (CARL, 2001) também tem como entrada um texto bilíngüe com n pares de sentenças alinhadas $a_1...a_n$, onde cada alinhamento a_i possui um lado esquerdo (s) e um direito (t) analisados, separadamente, por um *shallow parser* como apresentado em (2.3). Nesse exemplo, a, b, c, d, e são lemas de s e a', b', c', d', e' são lemas de t anotados com informação morfossintática.

$$a_1 : (a)b(c(d(e))) \leftrightarrow (((a')b')c')d'(e') \tag{2.3}$$

Para cada alinhamento a_i pode-se extrair $n \times m$ correspondências (alinhamentos) lexicais L_i: $\{l_1...l_{n\times m}\}$, em que n é o número de nós em s, ou seja, o número de parênteses no lado esquerdo do alinhamento em (2.3) e m, o equivalente no lado direito. Por exemplo, para o alinhamento em (2.3) podem ser geradas 16 ($n = m = 4, n \times m = 16$) correspondências lexicais apresentadas na Figura 5. Tanto os alinhamentos a_i, quanto as correspondências

lexicais l_i são generalizados na próxima etapa do processo de indução.

$l_1 : (a) \leftrightarrow (e')$	$l_5 : (e) \leftrightarrow (e')$	$l_9 : (de) \leftrightarrow (e')$	$l_{13} : (cde) \leftrightarrow (e')$
$l_2 : (a) \leftrightarrow (a')$	$l_6 : (e) \leftrightarrow (a')$	$l_{10} : (de) \leftrightarrow (a')$	$l_{14} : (cde) \leftrightarrow (a')$
$l_3 : (a) \leftrightarrow (a'b')$	$l_7 : (e) \leftrightarrow (a'b')$	$l_{11} : (de) \leftrightarrow (a'b')$	$l_{15} : (cde) \leftrightarrow (a'b')$
$l_4 : (a) \leftrightarrow (a'b'c')$	$l_8 : (e) \leftrightarrow (a'b'c')$	$l_{12} : (de) \leftrightarrow (a'b'c')$	$l_{16} : (cde) \leftrightarrow (a'b'c')$

Figura 5: Conjunto L_1 com regras de transferência lexical extraídas de a_1 (CARL, 2001)

No método proposto em (MENEZES & RICHARDSON, 2001), o algoritmo de alinhamento primeiro tenta encontrar correspondências lexicais entre nós fonte e alvo buscando pares de tradução em um léxico bilíngüe. Em seguida, considerando-se como ponto de partida os alinhamentos lexicais encontrados e seguindo uma estratégia *best-first* (considera, primeiro, os nós com melhores correspondências lexicais), o algoritmo alinha os nós restantes utilizando uma gramática de alinhamento com 18 regras de composição bilíngües codificadas manualmente. Por exemplo, nessa gramática há uma regra que especifica o alinhamento de um nó fonte com um nó alvo se eles possuem uma correspondência lexical e seus nós filhos já estão alinhados entre si. O propósito dessa gramática é garantir que apenas alinhamentos lingüisticamente significativos sejam gerados.

As regras da gramática de alinhamento são aplicadas em ordem e recursivamente até que nenhum novo alinhamento possa ser gerado. A Figura 6 apresenta um exemplo de árvores sintáticas alinhadas por esse método no qual as linhas pontilhadas indicam os alinhamentos entre os nós fonte e alvo. Neste exemplo, as correspondências lexicais que estão presentes no léxico bilíngüe usado para consultas foram identificados com a letra L.

A regra R1 especifica o alinhamento entre traduções bidirecionais únicas como é o caso de *dirección* e *address*, *usted* e *you* e *clic* e *click*. A regra R3 alinha os filhos de pais alinhados que possuem correspondência lexical como é o caso de *hipervínculo* e *hyperlink*. Com a resolução da ambigüidade que a palavra *hipervínculo* (possível tradução de *Hyperlink_Information* e *hyperlink*), a regra R1 é novamente aplicada para determinar o alinhamento entre *información* e *hipervínculo* com *Hyperlink_Information*. Por fim, a regra R4 é aplicada para criar o alinhamento entre *hacer* e *click* já que ela especifica, a grosso modo, que um nó verbo (*hacer*) cujo filho não-verbo (*clic*) está alinhado com nó verbo (*click*) deve se juntar ao filho no alinhamento com nó verbo na sentença alvo.

Em (MEYERS et al., 1996, 1998, 2000), no alinhamento dos nós utiliza-se a técnica de programação dinâmica para calcular (de modo *bottom-up*) a pontuação do casamento de cada nó na árvore fonte com cada nó na árvore alvo, gerando uma matriz |árvore fonte| × |árvore alvo| a partir da qual as correspondências serão recuperadas para geração das regras

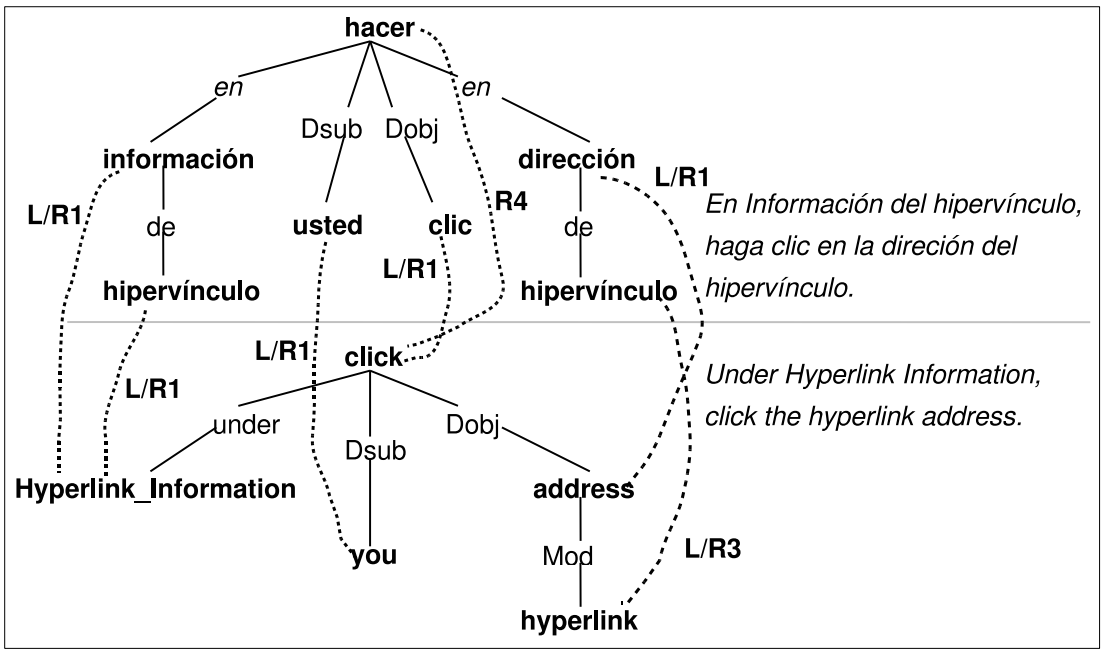


Figura 6: Árvore sintática com alinhamentos entre nós fonte e alvo (MENEZES & RICHARDSON, 2001)

de tradução.

De maneira semelhante a (MEYERS et al., 1998), o método apresentado em (LAVOIE et al., 2001) alinha os nós das árvores fonte e alvo utilizando a técnica de programação dinâmica, porém, nesse caso, a busca pelo mapeamento menos custoso entre os nós é realizada de maneira *top-down* e bi-direcional. Nessa busca são considerados os custos de alinhar dois nós cujos lemas não estão em um dicionário de transferência (com regras de transferência lexical extraídas de léxicos bilíngües ou dos próprios bitextos usando métodos estatísticos ou regras léxico-estruturais geradas manualmente) ou que possuem *part-of-speech* (PoS) ou posições relativas diferentes e o custo de remover ou inserir um nó em uma das árvores. Os custos de PoS foram determinados a partir de uma parte do *corpus* (com alinhamentos confiáveis) com base na co-ocorrência das etiquetas; os outros custos foram determinados manualmente.

Em (LAVIE et al., 2004; PROBST, 2005), o sistema infere regras hierárquicas de transferência sintática baseando-se, inicialmente, nos constituintes das duas línguas alinhados lexicalmente (por um processo manual). Para isso, as sentenças de treinamento escritas na língua com mais recursos (o inglês, nesse caso) são analisadas sintaticamente e desambiguadas. Todos os componentes da regra descritos na seção 2.1 e apresentados na Figura 2 (com exceção das restrições de ambos os lados) são gerados considerando-se a estrutura sintática

da língua com mais recursos, os alinhamentos lexicais e os dicionários das línguas fonte e alvo.

Embora o alinhamento estrutural (ou alinhamento de árvores sintáticas) seja muito útil na aquisição das regras de tradução, a construção automática ou manual de um *corpus* alinhado estruturalmente é uma tarefa muito complexa, além de estar sujeita a erros. *Parsers* robustos, para ambas as línguas, são muito difíceis de serem encontrados e a construção manual de um *corpus* alinhado estruturalmente é uma tarefa muito árdua; além disso, as gramáticas usadas para a análise sintática das duas línguas devem ser compatíveis (LIU & ZONG, 2004).

2.2.3 Geração das regras de tradução

Após a identificação dos padrões ou o alinhamento das árvores sintáticas, as regras são geradas aplicando-se diversas técnicas. Nesta seção são apresentadas algumas das técnicas de geração de regras de tradução utilizadas pelos métodos que identificam padrões e os que realizam a análise sintática, nessa ordem.

Em métodos que identificam os padrões nos exemplos de tradução, as regras são geradas por meio do agrupamento de padrões similares ou diferentes seguido da generalização desses padrões, ou seja, da substituição de algumas de suas partes por variáveis. Com relação a essa substituição, ela pode ser realizada com as similaridades – (BROWN, 2001) –, com as diferenças – (GÜVENIR & CICEKLI, 1998) – ou ambas – (MCTAIT, 2003) e (CICEKLI & GÜVENIR, 2001).

Como apresentado na subseção 2.2.1, durante o processo de identificação dos padrões de (BROWN, 2001), classes de equivalência são geradas para agrupar os padrões similares. Os padrões em cada classe de equivalência são, então, aplicados ao conjunto de exemplos substituindo-se cada instância de um membro de uma classe pelo nome da classe. As únicas exceções nesse processo de aplicação são aqueles exemplos nos quais essa substituição resultaria em uma seqüência composta apenas pelo nome da classe.

Com essa generalização, a similaridade dos exemplos de tradução é aumentada possibilitando novos casamentos de padrões na próxima iteração da etapa de identificação de padrões; uma vez que dois exemplos que, anteriormente, tinham segmentos iniciais diferentes podem, após a generalização, ter os mesmos segmentos iniciais se esses pertencerem à mesma classe de equivalência. Ao final, o conjunto de exemplos de tradução generalizados com as indicações de classes de equivalência e as próprias classes formam o conjunto de regras de

tradução.

Algo semelhante ocorre em (GÜVENIR & CICEKLI, 1998) com as diferenças e em (CICEKLI & GÜVENIR, 2001) com as diferenças e também com as similaridades, as quais são substituídas por variáveis. Por exemplo, considerando-se os exemplos de tradução em (15) com as similaridades sublinhadas e os padrões (16) e (17) aprendidos com base nestas similaridades, a generalização em (18) é gerada utilizando-se a heurística de diferenças aplicada à única similaridade existente nos pares em questão. Nesse caso, para acessar as correspondências entre ${V_1}^S$ e ${V_2}^T$ e entre ${V_2}^S$ e ${V_1}^T$ deve-se fazer referência aos padrões definidos em (16) e (17), respectivamente.

(15) E_1 : I give+PAST the book ↔ kitap +ACC ver+PAST+1SG
E_2 : You give+PAST the pencil ↔ kurşun kalem +ACC ver+PAST+2SG

(16) I ↔ +1SG
You ↔ +2SG

(17) book ↔ kitap
pencil ↔ kurşun kalem

(18) ${V_1}^S$ give+PAST the ${V_2}^S$ ↔ ${V_1}^T$+ACC ver+PAST ${V_2}^T$

Em (MCTAIT, 2003), após a identificação dos padrões executada em duas fases – monolíngüe e bilíngüe – (veja subseção 2.2.1) os fragmentos de texto correspodentes a esses padrões, bem como as variáveis que os rodeiam, são alinhados baseando-se na comparação de seqüências e em uma métrica de similaridade (distância) bilíngüe (veja equação (2.4)).

A medida de similaridade bilíngüe BS é neutra em relação à língua e é uma pontuação combinada baseada na distribuição lexical bilíngüe dos fragmentos de texto (BLD) e o número de cognatos que os fragmentos de texto compartilham. A pontuação de BLD (um valor real entre 0 e 1) é computada com coeficiente de Dice (DICE, 1945 apud MCTAIT, 2003) enquanto os cognatos são determinados usando a distância de Levenshtein (LEVENSHTEIN, 1966 apud MCTAIT, 2003).[6]

$$BS = \frac{BLD + |\text{cognatos}|}{1 + |\text{cognatos}|} \tag{2.4}$$

[6]A distância de Levenshtein é normalizada em relação à distância máxima entre 2 *strings* retornando uma pontuação de similaridade ou probabilidade de que 2 *strings* sejam cognatas.

O alinhamento é realizado em dois passos: o primeiro busca alinhamentos entre fragmentos/variáveis adjacentes e é realizado por uma programação dinâmica tradicional; o segundo busca alinhamentos 1 : 1 não adjacentes com altas probabilidades que, possivelmente, alteram os alinhamentos encontrados previamente. Por exemplo, dado o padrão de tradução em (20) identificado a partir dos exemplos de tradução em (19), ao final do processo de alinhamento têm-se as correspondências apresentadas em (21) para fragmentos (A_f) e variáveis (A_v).

(19) E_1 : The Commission gave the plan up ↔ La Comisión abandonó el plan
E_2 : Our Government gave all laws up ↔ Nuestro Gobierno abandonó todas las leyes

(20) ${V_1}^S\ {F_1}^S\ {V_2}^S\ {F_2}^S \leftrightarrow {V_1}^T\ {F_1}^T\ {V_2}^T$
${F_1}^S = \{\text{gave}\}$, ${F_2}^S = \{\text{up}\}$ e ${F_1}^T = \{\text{abandonó}\}$
${V_1}^S = \{\text{The Commision,Our Government}\}$, ${V_2}^S = \{\text{the plan,all laws}\}$, ${V_1}^T = \{\text{La Comisión,Nuestro Gobierno}\}$ e ${V_2}^T = \{\text{el plan,todas las leyes}\}$

(21) $A_f = \{({F_1}^S, {F_2}^S) : {F_1}^T\}$
$A_v = \{{V_1}^S : {V_1}^T, {V_2}^S : {V_2}^T\}$

Em métodos que realizam o alinhamento das árvores sintáticas, as regras são geradas aplicando-se técnicas que variam desde simples cálculos estatísticos (probabilidade) ou recuperação dos alinhamentos lexicais, até processos mais complexos de expansão dos nós alinhados ou o processo inverso de extração de subpadrões nas árvores alinhadas.

Em (KAJI et al., 1992), cada par de unidades (palavras ou sintagmas) associadas é um candidato a ser substituído por uma variável. Uma regra é obtida escolhendo-se um subconjunto da unidade associada e atribuindo-se uma variável única a cada par no subconjunto. As categorias sintáticas são anexadas à variável. Esse procedimento pode ser aplicado a qualquer subcojunto de unidades associadas, contanto que se escolha unidades que não se sobrepõem. Regras de tradução que representam apenas parte do par de sentenças podem ser embutidas no resultado da tradução de outra regra.

Em (CARL, 2001), com base nos alinhamentos e possíveis correspondências lexicais determinados no passo anterior (veja subseção 2.2.2), são criadas generalizações. Uma generalização possui pelo menos uma variável em cada lado da língua onde uma regra de menor nível casa com subseqüências nos lados esquerdo e direito. Portanto, uma generalização tem o mesmo número de variáveis nos lados esquerdo e direito e cada variável no lado esquerdo

está ligada a exatamente uma variável no lado direito. Porém, antes de gerar as generalizações, os alinhamentos (a_i) e as correspondências lexicais (l_i) recebem pesos calculados com base nas suas probabilidades, para que apenas as generalizações (g_i) de maiores pesos sejam geradas.

Para cada alinhamento a_i, as $n \times m$ correspondências lexicais (veja Figura 5) são ordenadas pelo tamanho da menor seqüência de palavras no lado esquerdo ou no lado direito do alinhamento. As correspondências $l_j \in L_i$ são, então, generalizadas começando com a menor regra. Por exemplo, a Figura 7 apresenta as generalizações induzidas a partir das correspondências lexicais da Figura 5. Nesse exemplo, considerando-se que há 16 correspondências lexicais extraídas do alinhamento a_1, a cada uma das generalizações é atribuída uma probabilidade igual a 1/4 (de acordo com a fórmula $p(l_j) = \frac{1}{x} \sum_{l_j \in L_i} \frac{1}{\sqrt{\#L_i}}$ e considerando $x = 1$ tem-se $p(l_j) = \frac{1}{1} \times \frac{1}{\sqrt{16}} = \frac{1}{4}$).

A generalização g_1 é gerada substituindo-se uma subseqüência em $l_{11} : (de) \leftrightarrow (a'b')$ pela regra $l_6 : (e) \leftrightarrow (a')$. O peso de g_1 é dado pela fórmula $w(g_k) = \sum_{r \in R_k} p(r) + \sum_{l \in L_k} w(l)$ – na qual R_k é o conjunto de alinhamentos (a_i) e correspondências lexicais (l_j) a partir dos quais g_k foi gerada –, assim $w(g_1) = p(l_6) + p(l_{11}) = \frac{1}{4} + \frac{1}{4} = \frac{2}{4}$. Esse peso também é atribuído à regra l_{11}, já que $w(l_j) = \max\{w(g \in G_j)\}$, ou seja, o peso de uma correspondência lexical l_j é dado pelo peso máximo das generalizações geradas por l_j, armazenadas em G_j. O novo peso de l_{11} também afeta o peso da generalização g_5, obtida com a substituição de l_{11}.

G_i	Generalização induzida	$p(g_k)$	$w(g_k)$
G_{11}	$g_1 : (d*) \leftrightarrow (*b')$	1/4	2/4
	$g_2 : (cd*) \leftrightarrow (*b'c')$	1/4	2/4
G_{16}	$g_3 : (cd*) \leftrightarrow (*c')$	1/4	2/4
	$g_4 : (c*) \leftrightarrow (*b'c')$	1/4	2/4
	$g_5 : (c*) \leftrightarrow (*c')$	1/4	3/4

Figura 7: Conjuntos G_{11} e G_{16} de generalizações induzidas a partir das correspondências l_{11} e l_{16} apresentadas na Figura 5 (CARL, 2001)

Assim, a gramática de tradução resultante é composta por alinhamentos a_i, correspondências lexicais l_j e generalizações g_k e pode, ainda, ser filtrada para a eliminação de ambigüidades.

Em (MENEZES & RICHARDSON, 2001), o processo é um pouco mais complexo e envolve a expansão dos alinhamentos gerados na etapa anterior com tipos e quantidades variadas de contexto utilizando construtores lingüísticos, como sintagmas nominais e verbais, para determinar as fronteiras dos contextos a serem inseridos. Assim, algumas das regras de

tradução obtidas para os alinhamentos das formas lógicas apresentados na Figura 6, são apresentadas na Figura 8.

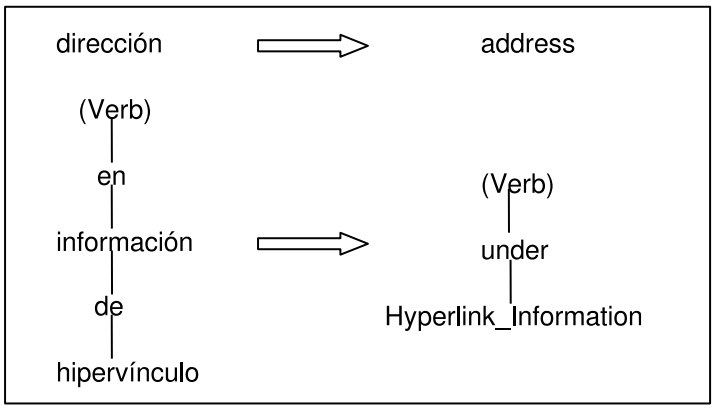


Figura 8: Regras de tradução obtidas para os alinhamentos das FLs apresentados na Figura 6 (MENEZES & RICHARDSON, 2001)

De maneira inversa, em (LAVOIE et al., 2001), após o alinhamento dos nós, as regras de tradução candidatas são geradas extraindo-se subpadrões dos pares alinhados usando restrições de alinhamento e de atributo geradas manualmente. As restrições de alinhamento definem a maioria das divergências sintáticas possíveis entre as línguas e identificam as subárvores de interesse nas árvores alinhadas. Por sua vez, as restrições de atributo limitam o espaço de regras de tradução que podem ser geradas a partir das subárvores que satisfazem as restrições de alinhamento e determinam que partes dessas subárvores devem ser incluídas nas regras de tradução candidatas.

Em (LAVIE et al., 2004), as regras são geradas por meio da composição e da generalização dos alinhamentos das árvores sintáticas (veja subseção 2.2.2). Primeiro, o método tenta fazer a composição verificando se cada subárvore pode ser obtida por uma regra de menor nível (mais simples). Se essa regra existir, a composição é realizada inserindo a regra de menor nível na regra de maior nível e atualizando as restrições. A maioria das restrições fonte são mantidas (com os índices ajustados para as novas seqüências), contudo, são eliminadas as restrições que pertencem ao trecho tratado pela regra de nível mais baixo (CARBONELL et al., 2002). As restrições alvo são determinadas comparando-se as partes traduzidas correta e incorretamente pela regra de nível mais baixo: para cada restrição na tradução correta o sistema checa se essa restrição aparece em todas as outras traduções, senão, uma nova restrição é construída e inserida na regra composta, com os índices atualizados.

Por fim, as regras são generalizadas por meio de um algoritmo denominado *Seeded Version Space Learning* que, primeiro, agrupa as regras similares, ou seja, com as mesmas

PoS, alinhamentos e tipos – isso significa que as regras pertencentes a cada grupo diferem, apenas, em suas restrições. Em seguida, o algoritmo analisa cada grupo (*version space*) separadamente tentando, repetidamente, unir duas regras por meio da remoção ou união de suas restrições[7]: (1) remoção de uma restrição de valor, (2) remoção de uma restrição de concordância e (3) união de duas restrições de valor formando uma restrição de concordância – duas restrições de valor podem ser unidas se elas possuem, por exemplo, o seguinte formato: se $((X_i \text{ feature}_k) = \text{value}_h)$ e $((X_j \text{ feature}_k) = \text{value}_h)$ então $((X_i \text{ feature}_k) = (X_j \text{ feature}_k))$.

A regra resultante só é aceita se conseguir cobrir todos os casos cobertos pelas duas regras que ela substitui e, além disso, sua aplicação não insere nenhum exemplo incorreto. A Figura 9 apresenta uma regra generalizada (terceira coluna) para as Regras 1 e 2 (primeira e segunda colunas, respectivamente). Nesse exemplo, uma nova restrição de concordância (`((y2 gender = (y1 gender))`) foi criada a partir de duas restrições de valor nas Regras 1 (`((y1 gender = *m)` e `(y2 gender = *m)`) e 2 (`((y1 gender = *f)` e `(y2 gender = *f)`).

Regra 1	Regra 2	Regra Generalizada
;;SL: the man	;;SL: the woman	;;SL:
;;TL: der Mann	;;TL: die Frau	;;TL:
NP::NP	NP::NP	NP::NP
[DET N] → [DET N]	[DET N] → [DET N]	[DET N] → [DET N]
(;;alignments:	(;;alignments:	(;;alignments:
(x1::y1)	(x1::y1)	(x1::y1)
(x2::y2)	(x2::y2)	(x2::y2)
;;x-side constraints:	;;x-side constraints:	;;x-side constraints:
((x1 agr) = *3-sing	((x1 agr) = *3-sing)	((x1 agr) = *3-sing)
((x1 def) = *def)	((x1 def) = *def)	((x1 def) = *def)
((x2 agr) = *3-sing)	((x2 agr) = *3-sing)	((x2 agr) = *3-sing)
((x2 count) = +)	((x2 count) = +)	((x2 count) = +)
;;y-side constraints	;;y-side constraints	;;y-side constraints
((y1 agr) = *3-sing)	((y1 agr) = *3-sing)	((y1 agr) = *3-sing)
((y1 case) = *nom)	((y1 case) = (*not* *gen *dat))	
((y1 def) = *def)	((y1 def) = *def)	((y1 def) = *def)
((y1 gender) = *m)	((y1 gender) = *f)	
((y2 agr) = *3-sing)	((y2 agr) = *3-sing)	((y2 agr) = *3-sing)
((y2 case) = *nom)		
((y2 gender) = *m)	((y2 gender) = *f)	((y2 gender) = (y1 gender))

Figura 9: Regras simples e generalizada (CARBONELL et al., 2002)

[7]Há dois tipos de restrições definidos em (LAVIE et al., 2004): restrição de valor e restrição de concordância. Uma restrição de valor é do tipo $((X_i \text{ feature}_k) = \text{value}_h)$ indicando que o atributo feature_k do item X_i possui o valor value_h. Uma restrição de concordância, por sua vez, é do tipo $((X_i \text{ feature}_k) = (X_j \text{ feature}_k))$ indicando que os itens X_i e X_j possuem o mesmo valor para o atributo feature_k.

Segundo (CARBONELL et al., 2002), essa abordagem é uma abordagem gulosa (do inglês, *greedy*) para a generalização e não oferece garantias de que as regras de tradução ótimas (mais gerais) sejam obtidas. Por outro lado, o método trata de maneira adequada as possíveis generalizações e é computacionalmente tratável.

2.2.4 Filtragem e ordenação das regras de tradução

Após a geração das regras de tradução, duas tarefas podem ser realizadas pelos métodos de indução: filtragem e ordenação. Alguns métodos filtram as regras de tradução, por exemplo, para eliminar ambigüidades. Há também os métodos que ordenam as regras com base em algum critério estatístico (probabilidade, freqüência etc.) ou de especificidade (ou generalização) preparando-as, assim, para serem usadas no processo de TA.

Em (KAJI et al., 1992), após gerar as regras de tradução, essas são refinadas para solucionar conflitos. Todas as regras de tradução obtidas a partir de um *corpus* bilíngüe são agrupadas por suas partes fonte e, depois, subagrupadas por suas partes alvo. Quando há um grupo de regras com mesma parte fonte, mas diferentes partes alvo, essas são consideradas conflitantes (ambíguas), uma vez que podem produzir diferentes traduções para a mesma sentença. As regras conflitantes são refinadas examinando-se os exemplos de tradução a partir dos quais elas foram geradas com o intuito de identificar categorias semânticas que façam a distinção de cada regra. Se for possível identificar tais categorias, essas são adicionadas às variáveis da regra resolvendo o conflito.

Uma maneira mais simples de filtrar as regras de tradução, aplicada em (MENEZES & RICHARDSON, 2001), baseia-se na freqüência das regras: quando há mais de uma regra com a mesma parte fonte, seleciona-se a regra de maior freqüência. Além disso, os autores também filtram a gramática de tradução permitindo que apenas as regras que ocorrem no mínimo N vezes (por exemplo, $N = 2$) estejam presentes. Esse processo de filtragem, segundo os autores, melhora consideravelmente o tempo de processamento do sistema de TA que utiliza as regras induzidas.

Em (CARL, 2001), a gramática gerada é filtrada para eliminar ocorrências ambíguas, ou seja, regras de tradução que possuem mesmo lado esquerdo ou lado direito. Nesse processo, apenas a regra de maior peso (veja subseção 2.2.3) é mantida para cada conjunto ambíguo. A Figura 10 apresenta um exemplo de gramática de tradução gerada após o processo de filtragem, em que $p(.)$ e $w(.)$ indicam, respectivamente, a probabilidade e o peso calculados para cada alinhamento (a_i), generalização (g_k) e correspondência lexical (l_j). Por

exemplo, como as generalizações g_1 e g_2 são ambíguas, apenas a de maior peso (g_1) é mantida na gramática filtrada.

Gramática induzida	$p(.)$	$w(.)$
$a_1 : (dx) \leftrightarrow (m'n')$	1/4	2/4
$g_1 : (d*) \leftrightarrow (m'*)$	1/4	2/4
$l_1 : (x) \leftrightarrow (n')$	1/4	1/4
$a_2 : (de) \leftrightarrow (a'b')$	1/8	1/4
$g_2 : (d*) \leftrightarrow (*b')$	1/8	1/4
$l_2 : (e) \leftrightarrow (a')$	1/8	1/8

Gramática filtrada	$p(.)$	$w(.)$
$g_1 : (d*) \leftrightarrow (m'*)$	1/4	2/4
$l_1 : (x) \leftrightarrow (n')$	1/4	1/4
$a_2 : (de) \leftrightarrow (a'b')$	1/8	1/4

Figura 10: Gramáticas induzida e filtrada (CARL, 2001)

Em (LAVOIE et al., 2001), primeiro, as regras são ordenadas decrescentemente de acordo com suas pontuações de *log likelihood* e, em caso de empate, priorizam-se as regras mais gerais (com base no número de atributos de relacionamento de dependência que as regras representam). Após serem ordenadas, as regras são filtradas, seguindo a ordem estabelecida previamente e uma de cada vez, removendo aquelas candidatas que não melhoram a precisão geral das árvores alvo produzidas. A cada iteração do processo de filtragem, uma regra candidata é adicionada provisoriamente ao conjunto de regras aceitas e o conjunto atualizado é aplicado a todas as estruturas fonte. As estruturas transferidas e as árvores alvo são comparadas e se o erro for menor do que o erro atual, a candidata permanece no conjunto e o erro é atualizado; caso contrário, a candidata é rejeitada e removida do conjunto de regras aceitas.

Um critério usado, freqüentemente, para a ordenação das regras é a especificidade (ou generalização) das mesmas. Em (CICEKLI & GÜVENIR, 2001), as regras de tradução são ordenadas por especificidade: a regra com maior número de terminais (palavras e não variáveis) na língua fonte é a mais específica. Porém, essa ordenação baseada no número de símbolos terminais, segundo Öz & Cicekli (1998), não é eficiente para grandes sistemas e, por isso, os autores propuseram um modelo estatístico para ordenar as regras de acordo com um fator de confiança.

Nesse modelo, são atribuídos pesos (fatores de confiança) às regras e a algumas combinações de regras. Na fase de aprendizado, a cada regra atribui-se um número (identificador) e uma vez que a tradução é bidirecional, dois pesos são atribuídos a cada regra/combinação, um para cada sentido (fonte $\rightarrow$ alvo e alvo $\rightarrow$ fonte). Assim, dada uma regra (R) do tipo $R : X \leftrightarrow Y$ e um conjunto de exemplos de tradução na forma $E_i : {E_i}^S \leftrightarrow {E_i}^T$, o peso dessa regra será calculado de acordo com a equação (2.5).

$$\text{peso}_1 = \frac{N_1}{N_1 + N_2} \tag{2.5}$$

em que N_1 é o número de exemplos de tradução nos quais X é uma *substring* de ${E_i}^S$ e Y é uma *substring* de ${E_i}^T$ e N_2 será o número de exemplos de tradução nos quais X é uma *substring* de ${E_i}^S$ e Y não é uma *substring* de ${E_i}^T$ se estivermos calculando o peso no sentido fonte → alvo; e o número de exemplos de tradução nos quais Y é uma *substring* de ${E_i}^T$ e X não é uma *substring* de ${E_i}^S$ se estivermos calculando o peso no sentido alvo → fonte. É possível que os valores do peso sejam os mesmos para a tradução em ambos os sentidos, mas é mais provável que esses valores sejam diferentes. De acordo com esse modelo, a melhor regra a ser aplicada será aquela cujo peso se aproxime mais do valor 1,0.

Outra maneira de atribuir pesos às regras, apresentada em (MEYERS et al., 2000), é usando probabilidades. O peso de uma regra R passível de ser aplicada a um nó N pertencente à árvore sintática da sentença fonte é calculado de acordo com a equação (2.6).

$$\text{peso}_2 = \log_2 \left(\frac{f(R)}{f(\text{todas as regras que se aplicam a } N)} \right) \tag{2.6}$$

$$\text{peso}_3 = \text{peso}_2 - \text{normalização} \tag{2.7}$$

A freqüência (f) de uma regra R é o número de vezes que essa regra casa com um exemplo no *corpus* de treinamento durante o processo de indução. O denominador é uma freqüência combinada de todas as regras que se aplicam a N. Segundo os autores, esse peso é dependente da maneira como as regras de tradução foram derivadas e a normalização aplicada em (2.7) garante que o conjunto mais provável de regras de tradução seja considerado o quanto antes.

2.3 Tradução automática por meio das regras induzidas

Essa seção descreve como as regras de tradução induzidas automaticamente por alguns dos métodos mencionados na seção 2.2 são usadas para traduzir uma sentença fonte em uma sentença alvo. De maneira geral, a TA é realizada em 3 etapas, como ilustrado na Figura 11: (1) pré-processamento da sentença fonte de entrada, (2) transferência e (3) geração da sentença alvo.

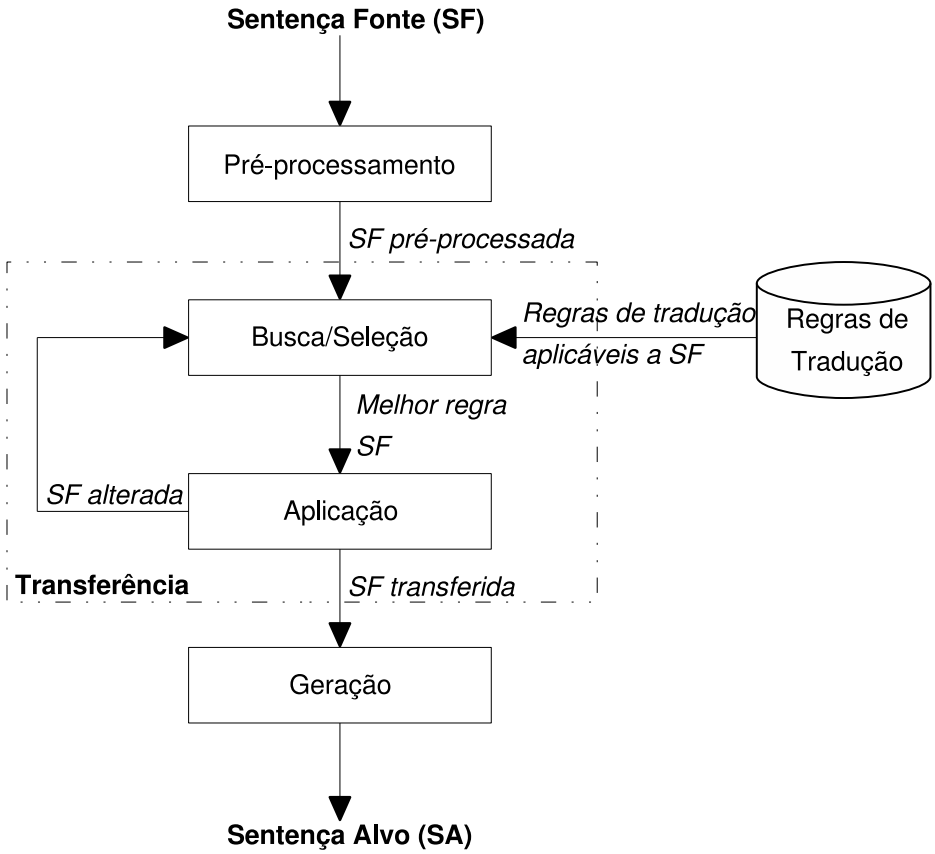


Figura 11: Etapas do processo de TA com base nas regras de tradução induzidas automaticamente

O pré-processamento da sentença fonte (SF) pode englobar lematização, análise morfológica, análise sintática ou qualquer outro processamento relevante para o tratamento em questão. A transferência, parte principal da TA, é realizada por meio de um processo recursivo que busca as regras de tradução aplicáveis à sentença fonte de entrada, seleciona a melhor regra a ser aplicada em um dado momento e aplica a regra selecionada; até que não seja possível aplicar mais nenhuma regra ou a sentença fonte já tenha sido completamente transferida. Por fim, a geração transforma o resultado do processo de transferência em uma sentença alvo aplicando as transformações necessárias e específicas da língua alvo (como inflexão, conjugação etc.).

Analisando-se mais especificamente o processo de transferência, é possível perceber que ele está dividido em 2 passos realizados recursivamente: (1) busca/seleção e (2) aplicação das regras de tradução. No primeiro passo, as regras de tradução passíveis de serem aplicadas à SF são buscadas com base no casamento dos padrões (do inglês, *pattern matching*) existentes em SF e nas regras do repositório de regras de tradução; em seguida, a melhor regra é selecionada com base em vários critérios como: tamanho (MENEZES & RICHARDSON, 2001);

especificidade (GÜVENIR & CICEKLI, 1998); técnicas de aprendizado de máquina (MENEZES, 2002); ou pesos baseados na freqüência (MENEZES & RICHARDSON, 2001; ÖZ & CICEKLI, 1998) ou na probabilidade (MEYERS et al., 2000) das regras candidatas. Os pesos podem ser calculados durante a indução das regras de tradução (na etapa de filtragem/ordenação) ou durante o processo de TA (na etapa de busca/seleção). Há, ainda, sistemas de TA que optam por aplicar todas as regras possíveis e selecionar não a melhor regra, mas, sim, a melhor sentença alvo gerada (MCTAIT, 2003).

Por fim, no último passo da transferência, a regra selecionada é aplicada, ou seja, um paralelo é estabelecido entre seus itens no lado esquerdo e os valores na SF e as transformações especificadas no lado direito da regra são realizadas resultando em uma seqüência de itens na língua alvo (KAJI et al., 1992).

Por exemplo, as regras induzidas em (MENEZES & RICHARDSON, 2001) são usadas no sistema de TA apresentado em (RICHARDSON et al., 2001) da seguinte maneira. Na etapa de pré-processamento, realiza-se a análise sintática da sentença fonte. Em seguida, durante a etapa de transferência, as regras de tradução adquiridas automaticamente são consultadas para verificar quais casam com porções da árvore sintática fonte. A seleção da melhor regra é feita com base em tamanho e freqüência: regras maiores (mais específicas) são priorizadas e, se houver mais de uma regra de mesmo tamanho, a de maior freqüência é selecionada.

Na aplicação da melhor regra, são utilizados um dicionário bilíngüe e algumas regras de tradução criadas manualmente para tratar os casos em que a aplicação das regras induzidas automaticamente não tem sucesso. Por fim, na geração, a representação alvo resultante da transferência é transformada na sentença alvo de saída utilizando-se regras de geração e um dicionário da língua alvo.

Em um trabalho mais recente, Menezes (2002) mostra que técnicas de aprendizado de máquina podem ser usadas na seleção da melhor regra a ser aplicada, de acordo com um determinado contexto. Essa estratégia melhorou a qualidade da tradução em 66,8% dos casos testados, quando comparada à estratégia de selecionar, sempre, a regra de maior tamanho e de maior freqüência.

2.4 Avaliação das regras de tradução

As regras de tradução resultantes do processo de indução podem ser avaliadas diretamente ou indiretamente. No primeiro caso, avaliam-se as regras de tradução resultantes do processo

de indução (veja repositório de regras de tradução da Figura 1), enquanto que, no segundo caso, as regras são usadas (recombinadas) para traduzir sentenças fonte em sentenças alvo e a avaliação é feita com base nas sentenças alvo produzidas (veja processo de TA/recombinação ilustrado na Figura 1).

Tradicionalmente, em ambos os casos, o processo de avaliação é trabalhoso e necessita da ajuda de especialistas para se determinar, por exemplo, a precisão ou a cobertura das regras de tradução ou a aceitabilidade das sentenças alvo geradas. Uma alternativa para tornar o processo de avaliação menos trabalhoso é realizá-lo automaticamente por meio de alguma métrica capaz de julgar a qualidade de uma regra ou de uma sentença alvo com base em uma ou mais sentenças de referência (consideradas corretas). Assim, nas próximas subseções são apresentadas as diferentes metodologias de avaliação das regras de tradução – direta não-automática (subseção 2.4.1), direta automática (subseção 2.4.2), indireta não-automática (subseção 2.4.3) e indireta automática (subseção 2.4.4) – seguidas por um breve relato de algumas avaliações dos métodos citados neste capítulo (subseção 2.4.5).

2.4.1 Avaliação direta não-automática

Na avaliação direta das regras de tradução, realizada de modo não-automático, um tradutor humano especialista nas duas línguas envolvidas é responsável por analisar as regras induzidas e julgá-las segundo sua cobertura, relevância, precisão ou qualquer outro critério de interesse. A avaliação direta não-automática é a mais trabalhosa das metodologias de avaliação pois, além da necessidade do tradutor humano ser especialista nas duas línguas, ele deve, também, estar familiarizado com o formalismo de representação das regras.

2.4.2 Avaliação direta automática

Uma alternativa para a avaliação direta não-automática apresentada na seção 2.4.1, é a avaliação direta automática a qual dispensa a necessidade de um tradutor humano especialista nas duas línguas envolvidas desde que haja uma maneira de avaliar as regras automaticamente. Por exemplo, a avaliação direta automática pode ser desempenhada por um sistema capaz de calcular automaticamente medidas como cobertura, relevância, precisão etc., para cada uma das regras de tradução. Em (CARL, 2001), o autor considerou como língua alvo a mesma língua fonte e verificou, nesse caso, a porcentagem de regras de tradução geradas com lado esquerdo (fonte) igual ao lado direito (alvo). Os resultados da avaliação do método de (CARL, 2001) são apresentados na subseção 2.4.5.

2.4.3 Avaliação indireta não-automática

Na avaliação indireta não-automática, o processo é um pouco menos trabalhoso já que, nesse caso, o especialista humano não precisa estudar o formalismo de representação das regras de tradução para analisá-las. Além disso, também não é necessário (em muitos casos) que ele tenha conhecimento das duas línguas, bastando que seja especialista apenas na língua alvo.

Assim, as regras de tradução são utilizadas em um sistema de TA e o especialista humano deve julgar, por exemplo, se a sentença alvo gerada para uma dada entrada na língua fonte é adequada (ou não) – como em (ÖZ & CICEKLI, 1998) – ou, ainda, se é melhor do que outra gerada por um sistema em comparação (desenvolvido com outra tecnologia) – como o Babelfish[8], por exemplo, em (MENEZES & RICHARDSON, 2001) e (LAVOIE et al., 2002). Os resultados das avaliações desses métodos são apresentados na subseção 2.4.5.

2.4.4 Avaliação indireta automática

Na avaliação indireta automática, a sentença alvo (candidata) gerada pelo sistema de TA com regras induzidas é comparada com uma ou mais sentenças de referência (consideradas corretas) por meio de uma métrica. Essa metodologia é a que tem sido mais aplicada, atualmente, para avaliar os sistemas de TA.

Alguns estudos – (DODDINGTON, 2002; TURIAN et al., 2003; FINCH et al., 2004) – sobre o número de sentenças de referências que devem ser utilizadas em uma avaliação indireta automática constataram que quanto maior o número de referências, melhor a performance da avaliação. Porém, em experimentos realizados com a métrica NIST (apresentada em detalhes a seguir) constatou-se que a avaliação melhora gradualmente quando até 4 referências são usadas, mas com mais de 4 referências sua performance começa a cair (FINCH et al., 2004).

Os trabalhos mais recentes em avaliação de sistemas de TA utilizam métricas que estão se tornando padrão, como BLEU (PAPINENI et al., 2002) e NIST (DODDINGTON, 2002); além das tradicionais precisão (*precision*), cobertura (*recall*) e medida-F (*F-measure*) (MELAMED et al., 2003). Uma breve descrição de cada uma dessas métricas é apresentada a seguir.

BLEU

A métrica BLEU (PAPINENI et al., 2002) – cujo nome provém de *BiLingual Evaluation Understudy* – avalia a saída de um sistema de TA medindo a precisão dos n-gramas (n

[8] `http://world.altavista.com`.

variando de 1 a 4, nesse caso) das sentenças alvo geradas automaticamente, em relação a um conjunto de traduções de referência. A idéia por trás dessa métrica é que uma boa tradução tem mais n-gramas em comum com as sentenças de referência do que uma tradução ruim (FINCH et al., 2004).

A BLEU é calculada como a média geométrica da precisão de n-grama, multiplicada pela penalidade de brevidade (*brevity penalty*, ou BP) que penaliza sentenças muito menores do que a(s) referência(s). Dessa maneira, a melhor candidata deve ser similar à(s) referência(s) em tamanho, escolha e ordem das palavras. A BLEU é calculada como mostra (2.8), em que o valor de N proposto pelos autores é 4 e p_n e BP são calculados como em (2.9) e (2.10), respectivamente.

$$\text{BLEU} = \text{BP} \times \exp\left(\sum_{n=1}^{N} \frac{1}{N} \ln\ p_n\right) \tag{2.8}$$

$$p_n = \frac{\sum_{w_1...w_n \in C} \text{count}_{\text{clip}}(w_1...w_n)}{\sum_{w_1...w_n \in C} \text{count}(w_1...w_n)} \tag{2.9}$$

em que C é a candidata a tradução, $\text{count}(w_1...w_n)$ é o número de vezes que o n-grama $w_1...w_n$ ocorre na candidata a tradução C e $\text{count}_{\text{clip}}(w_1...w_n)$ é o número de vezes que o n-grama $w_1...w_n$ casa com um n-grama de referência, limitado pelo número máximo de vezes que ele ocorre em qualquer uma das referências.

$$\text{BP} = \begin{cases} 1 & \text{se c} > \text{r} \\ \exp\left(1 - \frac{r}{c}\right) & \text{se c} \leq \text{r} \end{cases} \tag{2.10}$$

em que c é o tamanho da candidata C e r é o tamanho médio das referências para essa candidata.

A medida de precisão (p_n) captura dois aspectos da tradução: adequação e fluência. Uma tradução que utiliza as mesmas palavras (1-grama) que a(s) referência(s) tende a satisfazer a adequação enquanto que a existência de seqüências maiores de n-gramas em comum está relacionada à fluência (PAPINENI et al., 2002).

O valor da métrica BLEU, para uma candidata, varia entre 0 e 1, sendo que quanto mais próximo do 1, melhor é a sentença candidata em relação às sentenças de referência.

Em dois dos métodos de indução de regras de tradução apresentados anteriormente – (LAVOIE et al., 2002) e (LAVIE et al., 2004) – essa métrica foi utilizada para avaliar o

desempenho, como apresentado em detalhes na subseção 2.4.5.

NIST

A métrica NIST (DODDINGTON, 2002), assim como a BLEU, também se baseia em precisão de n-gramas (n variando de 1 a 5, nesse caso), porém ela emprega a média aritmética das quantidades de n-gramas ao invés da média geométrica como faz a BLEU. Outra diferença entre essas duas métricas é que, na NIST, os n-gramas são ponderados por pesos de acordo com a contribuição de informação que fornecem ao invés de simplesmente serem contados como acontece na BLEU (FINCH et al., 2004).

A NIST representa a informação média, por palavra, dada pelos n-gramas na candidata que casam com um n-grama de uma das referências no conjunto de referências. A penalidade de brevidade (BP') da NIST, em relação à BP da BLEU, penaliza mais seriamente as candidatas muito pequenas e menos as candidatas mais próximas das referências, em tamanho. Assim, a NIST é calculada como mostra (2.11) em que C, c, r e , $count(w_1...w_n)$ são os mesmos definidos para a BLEU e $N = 5$. $Info$[9] e BP$'$ são mostradas em (2.12) e (2.13), respectivamente.

$$\text{NIST} = \text{BP}' \times \sum_{n=1}^{N} \sum_{w_1...w_n \in C} \frac{\text{info}(w_1...w_n)}{\text{count}(w_1...w_n)} \tag{2.11}$$

$$\text{info}(w_1...w_n) = \log_2 \left[\frac{\text{número de ocorrências de } w_1...w_{n-1}}{\text{número de ocorrências de } w_1...w_n} \right] \tag{2.12}$$

$$\text{BP}' = \begin{cases} 1 & \text{se c} > \text{r} \\ \exp\left(\beta \ln^2\left(\frac{c}{r}\right)\right) & \text{se c} \leq \text{r} \end{cases} \tag{2.13}$$

em que β é selecionado de tal forma que quando $c = \frac{2r}{3}$, BP$' = 0,5$.

O valor da NIST é sempre positivo e quanto maior ele for, melhor é a candidata em relação às referências; porém não há um limite fixo para o valor máximo dessa métrica.

Em (LAVIE et al., 2004) o sistema de TA com as regras induzidas, além de ser avaliado com a métrica BLEU, também foi avaliado usando a métrica NIST, como apresentado em detalhes na subseção 2.4.5.

Precisão, cobertura e medida-F

[9] As quantidades de n-gramas usadas para calcular os pesos de informação são derivadas do conjunto de referência.

Embora as métricas apresentadas anteriormente sejam úteis na comparação da qualidade das sentenças alvo geradas por diferentes sistemas de TA, é difícil entender o que elas significam, ou seja, o que significa, por exemplo, um valor de 0,112 para BLEU ou 5,32 para NIST? Nesse sentido, em (MELAMED et al., 2003), os autores demonstram como sistemas de TA podem ser avaliados em termos das métricas bem conhecidas: precisão e cobertura. Os autores sustentam que essas métricas podem ser interpretadas graficamente de maneira intuitiva, o que torna mais fácil o entendimento dos problemas dos sistemas de TA avaliados e de como esses problemas podem ser solucionados.

Precisão, cobertura e medida-F são utilizadas há muitos anos para avaliar diversos sistemas de PLN em áreas como recuperação de informação e alinhamento de textos paralelos. Precisão e cobertura são calculadas comparando-se os itens candidatos com os itens de referência como mostram as equações (2.14) e (2.15), e a medida-F (2.16) é a combinação das duas métricas anteriores. Assim, a precisão demonstra o número de itens candidatos corretos ($|\text{candidatos} \cap \text{referência}|$) em relação à quantidade total de itens candidatos ($|\text{candidatos}|$), enquanto a cobertura indica o número de itens candidatos corretos ($|\text{candidatos} \cap \text{referência}|$) em relação à quantidade total de itens de referência ($|\text{referência}|$).

$$\text{precisão}(\text{candidatos}|\text{referência}) = \frac{|\text{candidatos} \cap \text{referência}|}{|\text{candidatos}|} \tag{2.14}$$

$$\text{cobertura}(\text{candidatos}|\text{referência}) = \frac{|\text{candidatos} \cap \text{referência}|}{|\text{referência}|} \tag{2.15}$$

$$\text{medida-}F = 2\frac{\text{cobertura} \times \text{precisão}}{\text{cobertura} + \text{precisão}} \tag{2.16}$$

No contexto da TA, a precisão verifica a capacidade do sistema em traduzir corretamente as sentenças, enquanto a cobertura indica a capacidade do sistema em traduzir corretamente o maior número possível de sentenças do conjunto de teste/referência. A medida-F, por sua vez, representa a combinação das duas métricas anteriores. Os valores para essas três métricas variam entre 0 e 1, sendo que um valor próximo do 1 significa uma boa qualidade do sistema avaliado.

Além de ser uma métrica bem conhecida e mais fácil de compreender, a medida-F mostrou-se, em alguns casos, mais confiável do que a BLEU e a NIST para avaliar os sistemas de TA nos experimentos apresentados em (TURIAN et al., 2003). Em outra avaliação apresentada em (FINCH et al., 2004), constatou-se, também, que a medida-F é a melhor

métrica quando são usadas quatro referências ou mais.

Dentre a métricas utilizadas para avaliar o sistema de TA em (MCTAIT, 2003), a cobertura foi a mais explorada. Em (BROWN, 2001), os autores também utilizaram cobertura para avaliar seu sistema, porém de uma maneira diferente da apresentada em (2.15); e em (MEYERS et al., 2000), os autores utilizam a medida-F para avaliar o sistema, mas com outra denominação (*accuracy*). Os resultados das avaliações desses métodos, com essas métricas, são apresentados na subseção 2.4.5.

2.4.5 Avaliação dos métodos de indução de regras de tradução

As seções anteriores apresentaram as diferentes metodologias de avaliação dos métodos de indução de regras de tradução. Nesta seção são apresentados os resultados das avaliações dos métodos citados na seção 2.2 de acordo com a metodologia de avaliação empregada.

Embora os métodos de indução de regras de tradução citados na seção 2.2 tenham sido avaliados utilizando diversas metodologias (avaliação direta ou indireta, não-automática ou automaticamente) e métricas (precisão, cobertura, BLEU, NIST etc.), em *corpora* de idiomas, gêneros e tamanhos muito variados, é possível identificar alguns pontos importantes nas avaliações apresentadas nesta seção. A Tabela 3 resume os resultados obtidos nas avaliações dos métodos agrupando-os de acordo com a metodologia de avaliação empregada (DA – direta automática, I – indireta não-automática e IA – indireta automática); e a Tabela 4 apresenta o tamanho dos *corpora* de treinamento e teste, os idiomas testados e o número de regras geradas.

Antes de comentar os valores apresentados na Tabela 3, são necessárias algumas considerações. Com relação aos valores relatados em (BROWN, 2001), além da métrica cobertura utilizada nessa avaliação[10] ser mais tolerável do que a cobertura tradicional (equação (2.15), subseção 2.4.4), o tamanho do *corpus* usado no processo de indução foi muito maior do que o utilizado nos outros métodos, por exemplo, 1.107.000 exemplos inglês–francês para se atingir a cobertura de 92,34% e 107.000 exemplos nos mesmos idiomas para se alcançar 77,70% de cobertura (veja Tabela 4).

Com base nos valores da Tabela 3 é possível constatar que a maioria dos métodos que realizam análise sintática – (MENEZES & RICHARDSON, 2001), (LAVOIE et al., 2002), (LAVIE et al., 2004) e (MEYERS et al., 2000) – foram avaliados com metodologias e métricas

[10] A cobertura, em (BROWN, 2001), foi calculada como a porcentagem do total de palavras na sentença fonte de entrada para as quais o sistema gera, pelo menos, uma palavra alvo como tradução.

Tabela 3: Resumo das avaliações de alguns dos métodos de indução de regras de tradução apresentados neste capítulo (parte 1)

Método	**Metodologia**	**Métrica**	**Resultados**
(CARL, 2001)	DA	Precisão	82% a 96,6%
(ÖZ & CICEKLI, 1998)	I	Precisão	60% das 5 primeiras sentenças alvo estavam corretas
(MENEZES & RICHARDSON, 2001)	I	Sistema indução (SI) X Babelfish	SI melhor em 46,5% dos casos e igual em 17%
(LAVOIE et al., 2002)	I	Sistema indução (SI) X Babelfish	SI (regras+léxico) melhor em 46% dos casos e igual em 27%
	IA	BLEU	0,0950 (regras+léxico) X 0,0802 (Babelfish)
(LAVIE et al., 2004)	IA	BLEU	0,112 (regras) X 0,102 (SMT) X 0,058 (EBMT)
		NIST	5,32 (regras) X 4,70 (SMT) X 4,22 (EBMT)
(MCTAIT, 2003)	IA	Cobertura	27,2% a 33,9%
(BROWN, 2001)	IA	Cobertura	72,23% a 89,44% (espanhol–inglês) 77,70% a 92,34% (francês–inglês)
(MEYERS et al., 2000)	IA	Medida-F	62,6% a 70,9%

Tabela 4: Resumo das avaliações de alguns dos métodos de indução de regras de tradução apresentados neste capítulo (parte 2)

Método	**Idiomas**	**# Treinamento**	**# Teste**	**# Regras**
(CARL, 2001)	fonte = alvo	4.997 sentenças	–	4.506
(ÖZ & CICEKLI, 1998)	inglês–turco	488 sentenças	–	4.723
(MENEZES & RICHARDSON, 2001)	espanhol–inglês	161.606 sentenças	200–500 sentenças	58.314
(LAVOIE et al., 2002)	coreano–inglês	1.433 sentenças	50 sentenças	2.133
(LAVIE et al., 2004)	hindi–inglês	17.589 sentenças ou sintagmas	258 sentenças	16
(MCTAIT, 2003)	inglês–francês	2.500 sentenças	1.000 sentenças	7.237–9.610
(BROWN, 2001)	francês–inglês	107.000–1.107.000 palavras	45.320 palavras	–
	espanhol–inglês	104.000–1.000.000 palavras	9.059 palavras	–
(MEYERS et al., 2000)	espanhol–inglês	1.039–2.355 sentenças	116–262 sentenças	1.109–2.191

que permitem compará-los com outros sistemas disponíveis comercialmente, como é o caso do Babelfish. Talvez, por esse motivo, os métodos com análise sintática, aparentemente, possuem melhor desempenho do que os métodos que não realizam essa análise. Como já mencionado anteriormente, os valores de BLEU e NIST não são de fácil compreensão, mas pode-se dizer que nas avaliações com essas métricas os sistemas de TA que utilizavam as regras induzidas se saíram melhor do que o Babelfish (LAVOIE et al., 2002) e sistemas estatístico (SMT) e baseado em exemplos (EBMT) (LAVIE et al., 2004).

Com base nos valores da Tabela 4, pode-se perceber que o tamanho do *corpus* de treinamento varia de 488 a 161.606 sentenças e o do *corpus* de teste, de 50 a 1.000 sentenças. Em avaliações do mesmo método com tamanhos de corpora variados (como em (BROWN, 2001)), constatou-se melhor desempenho em *corpora* maiores, porém não se pode afirmar que isso é verdade para todos os métodos. Além disso, o número de regras geradas pelo processo de indução de regras de tradução varia muito entre os métodos estudados: de 16 a 58.314.

Assim, não se pode afirmar qual é o melhor método de indução de regras de tradução existente hoje nem mesmo dizer qual é o estado da arte em termos de precisão, cobertura ou alguma outra métrica, nessa área.

3 Indução de léxicos bilíngües

Os léxicos bilíngües são recursos lingüísticos de grande importância para diversas áreas de PLN já que especificam as correspondências entre palavras e, às vezes, multipalavras em dois idiomas. Tais recursos são fundamentais em qualquer sistema de tradução automática e têm papel vital em outras aplicações multilíngües, como na tradução assistida por computador (MELAMED, 1996c; LANGLAIS et al., 2001), no alinhamento de *corpora* paralelos (DAGAN et al., 1993; FUNG & CHURCH, 1994; MELAMED, 1996a), nos concordanciadores para lexicografia bilíngüe (GALE & CHURCH, 1991), na recuperação multilíngüe de documentos (RESNIK & MELAMED, 1997), entre outros; e mesmo em aplicações monolíngües, por exemplo, na desambiguação lexical de sentido (DAGAN & ITAI, 1994).

Nesse contexto, têm sido desenvolvidas várias pesquisas relacionadas à construção automática de léxicos bilíngües como produto final (WU & XIA, 1994; MELAMED, 1996b; RESNIK & MELAMED, 1997) ou como passo intermediário, por exemplo, na tradução automática (BROWN et al., 1993) e no alinhamento de *corpora* bilíngües (BROWN et al., 1991; DAGAN et al., 1993).

A próxima seção (3.1) apresenta uma descrição formal de léxico bilíngüe. Em seguida, alguns dos métodos de indução de léxicos bilíngües propostos na literatura são descritos brevemente (seção 3.2). Por fim, apresentam-se as metodologias de avaliação e alguns dos principais resultados relatados na literatura (seção 3.3).

3.1 Léxicos bilíngües

Segundo Melamed (1996b), há várias maneiras de se organizar um léxico bilíngüe, contudo a representação mais usual talvez seja a de um conjunto de pares ordenados de palavras. Um léxico bilíngüe (ou *translation lexicon*, tradução mais comum em inglês) para as línguas S e T pode ser formalmente definido como um subconjunto do produto cartesiano (*crossproduct*) das palavras de S e as palavras de T. Cada entrada B do léxico bilíngüe é um par ordenado

(w^S, w^T), onde $w^S \in S$ e $w^T \in T$.

Alguns métodos de indução de léxicos bilíngües adicionam uma pontuação de associação – por exemplo, probabilidade ou alguma medida de confiança – a cada entrada. Uma alta pontuação de associação indica que duas palavras estão fortemente associadas, ou seja, são "boas" traduções mútuas. Um léxico bilíngüe com uma pontuação de associação adicionada a cada entrada é denominado léxico bilíngüe pontuado (tradução do termo em inglês *graded translation lexicon*) (MELAMED, 1996b).

A Tabela 5 traz um exemplo de entradas bilíngües (alemão–inglês) geradas pelo método apresentado em (KOEHN & KNIGHT, 2002), com base na similaridade ortográfica, acompanhadas de suas respectivas pontuações de associação.

Tabela 5: Entradas alemão–inglês com suas respectivas pontuações de associação geradas pelo método apresentado em (KOEHN & KNIGHT, 2002)

Alemão	**Inglês**	**Pontuação**	
organisation	organization	0,92	correta
präsident	president	0,90	correta
industrie	industries	0,90	correta
parlament	parliament	0,90	correta
interesse	interest	0,89	correta
	...		
experte	expert	0,86	correta
investition	investigation	0,85	errada
mutter	matter	0,83	errada
bruder	border	0,83	errada
nummer	number	0,83	correta

3.2 Métodos de indução de léxicos bilíngües

De acordo com Melamed (1996b), a maioria dos algoritmos estatísticos projetados para produzir léxicos bilíngües para o par de línguas S e T – por exemplo, (GALE & CHURCH, 1991), (FUNG, 1995), (MELAMED, 1995) e (WU & XIA, 1994) – são variações do algoritmo guloso (do inglês, *greedy*) apresentado a seguir:

1. Escolhe-se a medida que será usada para calcular a similaridade D entre as palavras de S e as palavras de T, ou seja, a pontuação de associação. A medida de similaridade geralmente especifica quão freqüentemente as palavras co-ocorrem em regiões correspondentes de um *corpus* de textos paralelos, embora medidas diferentes também tenham sido propostas, por exemplo em (FUNG, 1995).

2. Calculam-se as pontuações de associação $D(w^S, w^T)$ para cada par de palavras $(w^S, w^T) \in (S \times T)$.

3. Ordenam-se os pares de palavras de acordo com a ordem decrescente de suas pontuações de associação.

4. Escolhe-se um limite para o qual os pares de palavras com pontuação de associação maior do que tal limite se tornam as entradas do léxico bilíngüe.

Esse algoritmo apresenta bom desempenho apesar de sua simplicidade, porém ele possui um problema: geralmente os algoritmos calculam as pontuações de associação (passo 2) independentemente umas das outras, o que não permite diferenciar uma associação direta (traduções mútuas de fato) de uma associação indireta (palavras que sempre aparecem no mesmo contexto, porém não são traduções mútuas). Não surpreendentemente, esses algoritmos produzem léxicos bilíngües cheios de associações indiretas (e incorretas). A Figura 12 ilustra casos de associações diretas e indiretas.

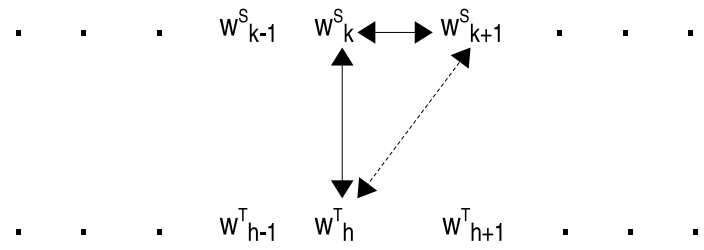


Figura 12: A associação direta entre as palavras $w^S{}_k$ e $w^T{}_h$ e entre as palavras $w^S{}_k$ e $w^S{}_{k+1}$ dá origem a uma associação indireta entre $w^S{}_{k+1}$ e $w^T{}_h$ (MELAMED, 1996b)

As irregularidades (ruído) no texto e na tradução amenizam esse problema já que esse ruído enfraquece uma associação direta e, conseqüentemente, uma associação indireta baseada na associação direta enfraquecida. Por outro lado, o ruído pode enfraquecer uma associação indireta sem afetar nenhuma associação direta. Sendo assim, em média, as associações diretas são mais fortes do que as indiretas.

Gale & Church (1991) demonstraram que, se todas as entradas em um léxico bilíngüe forem ordenadas por suas pontuações de associação, mais de 98% das entradas no topo da lista estão corretas. Esses autores apresentam um método estatístico para encontrar as correspondências bilíngües em um *corpus* inglês–francês. Neste método é aplicada uma estratégia de profundidade progressiva (*progressive deepening strategy*): a busca pelos melhores pontos de correspondência é feita, incialmente, em uma parte pequena do *corpus* e o escopo

da busca é aumentado a cada passo subseqüente. A cada iteração, os pares de palavras já selecionados em iterações anteriores são removidos do *corpus* de treinamento para que outras alternativas possam ser identificadas.

Uma estratégia similar é adotada em (WU & XIA, 1994) e em (FUNG, 1995). O método de Wu & Xia (1994) induz automaticamente um léxico bilíngüe inglês–chinês por meio do treinamento estatístico realizado com um grande *corpus* paralelo (com mais de 3 milhões de palavras). O processo de treinamento bilíngüe emprega uma variação do modelo de Brown et al. (1993) e está baseado em um procedimento iterativo de *expectation-maximization* (EM) para maximizar a probabilidade de geração de um *corpus* chinês dada a versão em inglês. A saída do processo de treinamento é um conjunto de possíveis traduções, em chinês, para cada palavra em inglês, juntamente com a probabilidade estimada para cada tradução.

Em (FUNG, 1995), o autor propõe um método para a indução de entradas bilíngües envolvendo apenas substantivos, nomes próprios e sintagmas nominais a partir de um *corpus* paralelo não alinhado inglês–chinês. A motivação para a indução de entradas dos tipos citados está no fato de que termos de domínios específicos são difíceis de serem traduzidos já que, freqüentemente, não aparecem nos dicionários bilíngües de domínio geral.

Fung (1995) considera o problema de compilação de léxicos bilíngües como um problema de casamento de padrão: cada palavra compartilha algumas características comuns com sua contra-parte no texto traduzido. O método tenta encontrar as melhores representações dessas características e o melhor modo de casá-las. Para os autores, as características compartilhadas entre as palavras fonte e alvo são: suas posições no *corpus*, a tendência de se agruparem na diagonal quando suas posições são plotadas em um gráfico (com as posições fonte em um eixo e as posições alvo em outro) e a tendência de formarem segmentos alinhados. Com base nessas características, um léxico inicial é criado com os pares de palavras (pontos âncoras) que dividem o *corpus* em segmentos alinhados.

Em seguida, os substantivos e os nomes próprios restantes em inglês e todas as palavras em chinês são representados na forma de vetores binários de segmentos não-lineares a partir de suas posições no texto. Por fim, os vetores binários em inglês são casados com suas contra-partes em chinês usando uma pontuação de informação mútua, e são filtrados com base em um fator de confiança. Os pontos resultantes após o filtro dão origem ao segundo léxico bilíngüe.

Em (RESNIK & MELAMED, 1997) os autores aplicam o sistema `SABLE` (MELAMED, 1997b) em um *corpus* de domínio técnico com aproximadamente 400.000 palavras com o

intuito de induzir um léxico bilíngüe de termos. O sistema SABLE (*Scalable Architecture for Bilingual LExicography*) produz léxicos bilíngües a partir de textos paralelos (bitextos) não-alinhados. Esse sistema foi desenvolvido para trabalhar com qualquer gênero de texto em qualquer par de línguas e não usa nenhum recurso específico para as línguas envolvidas, apenas os tokenizadores e algumas heurísticas para a identificação de pares de palavras que são traduções mútuas.

Depois de tokenizar as duas partes do bitexto, SABLE chama o algoritmo SIMR (MELAMED, 1996a) e seus componentes relacionados para produzir o mapeamento do bitexto. Um mapeamento de bitexto é uma função injectiva parcial entre as posições dos caracteres nas duas partes do bitexto – similar ao mapeamento realizado em (FUNG, 1995) ao plotar as posições das palavras fonte e alvo em eixos perpendiculares. Cada ponto de correspondência (x, y) no mapeamento do bitexto indica que a palavra cujo caractere mediano está na posição x do texto fonte é uma tradução da palavra cujo caractere mediano está na posição y do texto alvo.

O algoritmo SIMR possui duas fases – geração e filtragem dos pontos de correspondência – as quais são executadas alternadamente. Na fase de geração, os pontos de correspondência são gerados usando um subconjunto de heurísticas aplicadas a palavras – baseadas em cognatos (SIMARD et al., 1992; MELAMED, 1995, 1996a) ou léxicos bilíngües iniciais (MELAMED, 1997a) – selecionado de acordo com a língua e os recursos disponíveis. Na fase de filtragem, o SIMR filtra os pontos de correspondência candidatos usando um algoritmo de reconhecimento de padrão geométrico.

Após a determinação dos pontos de correspondência realizada por SIMR, o SABLE considera que dois *tokens* co-ocorrem se seus pontos de correspondência estão há uma distância pequena d do mapeamento do bitexto interpolado no espaço do bitexto como apresentado na Figura 13.

SABLE usa a estatística de co-ocorrência dos tokens para induzir um léxico bilíngüe inicial, usando o método descrito em (MELAMED, 1995). O módulo de filtro iterativo alterna entre estimação das traduções mais prováveis entre *tokens* no bitexto e estimação das traduções mais prováveis entre *types*. Por fim, SABLE constrói automaticamente um léxico bilíngüe composto de pares de palavras que não foram removidas durante o ciclo de filtro iterativo (MELAMED, 1996b).

A cobertura do léxico bilíngüe pode ser computada automaticamente em relação ao bitexto de entrada (MELAMED, 1996b), assim os usuários do SABLE têm a opção de

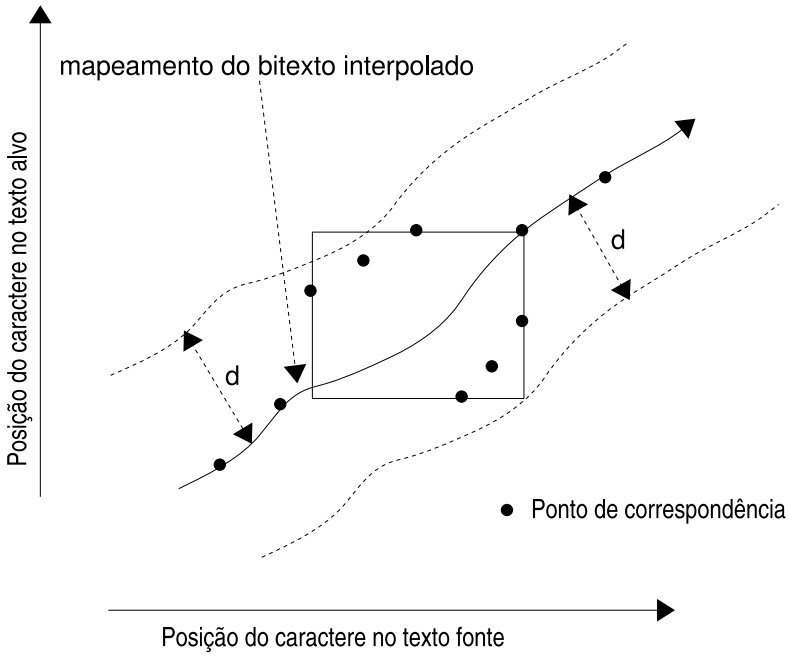


Figura 13: Pares de palavras cujas coordenadas estão entre as linhas pontilhadas são considerados co-ocorrentes (RESNIK & MELAMED, 1997)

especificar a cobertura que eles desejam na saída. Por padrão, `SABLE` seleciona um limite que provavelmente produzirá uma boa precisão.

Além dos métodos que buscam correspondências bilíngües com base em estatísticas, cognatos e outras métricas de similaridade em textos paralelos como os métodos apresentados até o momento, existem outros que utilizam, por exemplo, *corpora* monolíngües não-relacionados (KOEHN & KNIGHT, 2002) ou uma língua ponte (*bridge language*) (SCHAFER & YAROWSKY, 2002) para induzir os léxicos bilíngües.

O método de Koehn & Knight (2002) constrói um léxico bilíngüe alemão–inglês para substantivos a partir de *corpora* monolíngües não-relacionados combinando várias heurísticas. Para tanto, dois *corpora* monolíngües com textos em domínios comparáveis – textos jornalísticos, no caso dos experimentos apresentados pelos autores – são utilizados. A partir desses *corpora*, os pares de palavras que são traduções mútuas são determinados com base em 5 heurísticas: (1) palavras idênticas ou que diferem em apenas uma letra, (2) ortografia similar (calculada por meio da *longest common subsequence ratio* (LCSR)) (MELAMED, 1995), (3) contexto de ocorrência similar, (4) similaridade e (5) freqüência de palavras (medida como a razão da freqüência da palavra normalizada pelo tamanho do *corpus*).

A heurística de contexto de ocorrência similar assume que se os *corpora* monolíngües são comparáveis, uma palavra fonte que ocorre em um certo contexto deve ter a tradução

ocorrendo em um contexto similar. Assim, vetores de contexto são criados e traduzidos com base no conjunto inicial de correspondências obtido aplicando-se a primeira heurística. O vetor de contexto que melhor casa é usado para construir um mapeamento de palavra.

A similaridade de palavras, por sua vez, parte do pressuposto de que pares de palavras similares em uma língua provavelmente possuem traduções similares na outra língua (como ocorre entre as palavras que designam dias da semana). Assim, para uma nova palavra, calcula-se sua pontuação de similaridade em relação às palavras no conjunto inicial de correspondências (gerado com base na primeira heurística), criando um vetor de similaridades. Essa pontuação de similaridade é calculada com base nos vetores de contexto gerados anteriormente (terceira heurística). O vetor de similaridade com melhor casamento adiciona as palavras correspondentes ao léxico bilíngüe.

Por fim, o método de Schafer & Yarowsky (2002), também usa algumas heurísticas para induzir léxicos bilíngües porém sem utilizar *corpora* bilíngües paralelos nem um léxico bilíngüe inicial. Os autores propõem um método cujo objetivo é aprender léxicos bilíngües usando recursos disponíveis na *web* por meio do uso de uma língua ponte, ou seja, esse método não utiliza nenhum léxico entre o inglês e a língua de interesse (sérvio ou gujarati[1]), mas sim um entre o inglês e a língua ponte. Assim, os dicionários usados nos experimentos foram: checo–inglês (com 171K entradas) e hindi–inglês (com 74K entradas).

Os vocabulários de sérvio e gujarati foram obtidos extraindo-se dos *corpora* as palavras únicas (*word types*) e excluindo-se as palavras pouco freqüentes e as muito pequenas (com menos de 5 caracteres). Assim como em (KOEHN & KNIGHT, 2002), o método de (SCHAFER & YAROWSKY, 2002) baseia-se na combinação de 4 modelos de similaridade – similaridade de *string*, similaridade de contexto, similaridade de distribuição de datas e similaridade de freqüência de palavras. Além disso, outras características dos pares de palavras candidatos são consideradas na geração do léxico bilíngüe como a consistência de PoS: se as palavras diferem na PoS uma penalidade é atribuída a essa correspondência para ranqueá-la abaixo das candidatas com PoS compatíveis, mas não excluí-la.

Para cada medida de similaridade, as candidatas em inglês são ordenadas decrescentemente pelo valor dessa medida. A pontuação de cada palavra em inglês é calculada com base na classificação normalizada (obtida com base no valor da medida de similaridade e no peso do modelo de similaridade).

[1]Um dos idiomas da Índia.

3.3 Avaliação dos léxicos bilíngües

De modo geral, os léxicos bilíngües induzidos automaticamente podem ser avaliados seguindo duas metodologias distintas: avaliação intrínseca ou avaliação extrínseca. Na avaliação intrínseca, as entradas do léxico são avaliadas em termos do conteúdo que representam, de acordo com alguma métrica de interesse. Na avaliação extrínseca, o léxico bilíngüe é utilizado em alguma tarefa de PLN e avalia-se o resultado da aplicação desse recurso verificando-se o desempenho final na tarefa escolhida. Nesse sentido, é possível criar paralelos entre a avaliação direta das regras de tradução e a avaliação intríseca dos léxicos bilíngües; e entre a avaliação indireta das regras de tradução e a avaliação extrínseca dos léxicos bilíngües.

De modo semelhante ao que ocorre na avaliação das regras, tanto a avaliação intríseca como a avaliação extrínseca dos léxicos bilíngües podem ser realizadas de maneira automática ou manual. Assim, nas próximas subseções descrevem-se brevemente as diferentes metodologias de avaliação dos léxicos bilíngües – intríseca manual (subseção 3.3.1), intrínseca automática (subseção 3.3.2), extrínseca manual (subseção 3.3.3) e extrínseca automática (subseção 3.3.4) – e algumas avaliações dos métodos citados neste capítulo (subseção 3.3.5).

3.3.1 Avaliação intrínseca manual

A avaliação intrísenca manual das entradas de um léxico bilíngüe é realizada com o auxílio de juízes humanos. Tais juízes são responsáveis por julgar as entradas em um léxico bilíngüe como válidas (corretas), úteis (por exemplo, na geração de glossários de termos técnicos, onde as entradas de uso geral são consideradas inúteis) ou de acordo com outro critério de interesse.

Geralmente a precisão é calculada como a porcentagem de entradas válidas (V) embora, às vezes, as entradas classificadas como parcialmente válidas (PV) também sejam consideradas no cálculo da precisão.

Essa parece ser a metodologia mais comumente empregada na avaliação dos léxicos bilíngües já que foi utilizada em quatro dos seis métodos apresentados neste capítulo: (WU & XIA, 1994), (FUNG, 1995), (RESNIK & MELAMED, 1997) e (SCHAFER & YAROWSKY, 2002).

3.3.2 Avaliação intrínseca automática

A avaliação intrínseca automática das entradas de um léxico bilíngüe é realizada por meio da comparação automática das entradas do léxico induzido automaticamente com as entradas existentes em um léxico bilíngüe de referência. Nesta comparação, as entradas do léxico induzido que estão presentes no léxico de referência são consideradas corretas. Dentre os métodos apresentados neste capítulo, dois foram avaliados com essa metodologia: (SCHAFER & YAROWSKY, 2002) e (KOEHN & KNIGHT, 2002).

3.3.3 Avaliação extrínseca manual

Na avaliação extrínseca manual, o léxico bilíngüe é utilizado em alguma tarefa de PLN – tradução automática, recuperação de informação multilíngüe etc. – e juízes humanos avaliam a saída desta tarefa verificando se a utilização do léxico resultou em alguma melhora no desempenho. Dentre os métodos estudados, nenhum foi avaliado empregando-se esta metodologia.

3.3.4 Avaliação extrínseca automática

Na avaliação extrínseca automática, assim como na manual, o léxico bilíngüe é utilizado em alguma tarefa de PLN só que, desta vez, a saída de tal tarefa é avaliada automaticamente. O método de (GALE & CHURCH, 1991), por exemplo, foi avaliado extrínseca e automaticamente utilizando-se o léxico induzido no contexto de um concordanciador bilíngüe, enquanto o método de (KOEHN & KNIGHT, 2002) foi avaliado no contexto da TA.

3.3.5 Avaliação dos métodos de indução de léxicos bilíngües

As seções anteriores apresentaram as diferentes metodologias de avaliação dos léxicos bilíngües induzidos automaticamente. Como mencionado em (LANGLAIS et al., 2001), os métodos de indução de léxicos são, por natureza, difíceis de serem comparados entre si. Mesmo assim, embora os métodos de indução de léxicos bilíngües citados na seção 3.2 tenham sido avaliados utilizando diversas metodologias (avaliação intrínseca ou extrínseca, manual ou automática), métricas (precisão, cobertura e utilidade) e contexto (apenas a melhor candidata, as n melhores candidatas etc.), em *corpora* de idiomas, gêneros e tamanhos muito variados, é possível identificar alguns pontos importantes nas avaliações apresentadas nesta seção.

A Tabela 6 resume os resultados obtidos nas avaliações dos métodos agrupando-os de acordo com a metodologia de avaliação empregada (IM – intrínseca manual, IA – intrínseca automática e EA – extrínseca automática); e a Tabela 7 apresenta o tamanho dos *corpora* a partir dos quais os léxicos foram induzidos (treinamento) e testados (teste), os idiomas envolvidos e o número de entradas bilíngües geradas.

Tabela 6: Resumo das avaliações dos métodos de indução de léxicos bilíngües apresentados neste capítulo (parte 1)

Método	**Metodologia**	**Métrica**	**Resultados**
(GALE & CHURCH, 1991)	EA	precisão cobertura	95% 61%
(WU & XIA, 1994)	IM	precisão	86% (1a. melhor tradução) a 91% (em média, 2,33 traduções)
(FUNG, 1995)	IM	precisão cobertura	73,1% (1a. melhor tradução e 3 melhores) 23,78%
(RESNIK & MELAMED, 1997)	IM	precisão cobertura	81% a 89% V/PV 55% a 56% V 30,4% a 37,0%
(KOEHN & KNIGHT, 2002)	IA e EA	precisão	38,6% (palavras mais freqüentes)
(SCHAFER & YAROWSKY, 2002)	IA e IM	precisão	43% a 58% (inglês–sérvio) 30% a 46% (inglês–gujarati)

Tabela 7: Resumo das avaliações dos métodos de indução de léxicos bilíngües apresentados neste capítulo (parte 2)

Método	**Idiomas**	**# Treinamento**	**# Teste**	**# Entradas**
(GALE & CHURCH, 1991)	inglês–francês	890.000 sentenças	800 sentenças	13.466
(WU & XIA, 1994)	inglês–chinês	18.329 sentenças	200 palavras	6.517
(FUNG, 1995)	chinês–inglês	5.760 palavras	661 entradas	661
(RESNIK & MELAMED, 1997)	francês–inglês	410.320 palavras	100 entradas	3.135–4.071
(KOEHN & KNIGHT, 2002)	alemão–inglês	–	5.000 sentenças	185
(SCHAFER & YAROWSKY, 2002)	inglês–sérvio inglês–gujarati	204M *tokens* 194M *tokens*	– –	– –

Em apenas dois dos métodos avaliados intrínseca e manualmente foram citados os números de juízes: em (FUNG, 1995), o léxico induzido foi avaliado por três juízes humanos enquanto em (RESNIK & MELAMED, 1997) seis juízes avaliaram o léxico gerado.

Neste último trabalho, os juízes classificaram as entradas como: inválidas, válidas (V) ou parcialmente válidas (PV) – as quais necessitam uma mudança de PoS na tradução ou estão incompletas (quando deveriam envolver mais de uma palavra para a tradução ser válida). Com base nas classificações dos 6 juízes, uma classificação do grupo foi gerada considerando-se as entradas nas quais pelo menos 3 juízes atribuíram a mesma classificação.

A concordância entre os juízes foi verificada por meio da medida Kappa (CARLETTA, 1996), cujo valor foi obtido comparando-se a avaliação de cada juiz com a avaliação do grupo, resultando em valores de kappa (k) que variam de 0,55 a 0,74. De acordo com Carletta

(1996), um valor de $k > 0,8$ indica uma boa replicabilidade enquanto valores entre 0,67 e 0,8 permitem que conclusões sejam tiradas. Porém, de acordo com Craggs & Wood (2005), assumir estes valores indiscriminadamente para qualquer estudo é um erro comum já que, devido à diversidade de fenômenos sendo codificados e das aplicações de seus resultados, é impossível estabelecer os limites com base nos quais todas as codificações podem ser julgadas. Por fim, esses autores concluem que cada um deve decidir, com base no uso pretendido para o esquema de codificação, se os níveis de concordância observados são suficientes e, assim, realizar a análise dos resultados.

Tanto em (FUNG, 1995) quanto em (RESNIK & MELAMED, 1997) os autores concluem que os erros de etiquetação foram responsáveis por boa parte dos erros de tradução. Por fim, alguns dos métodos apresentados restringem o escopo de indução a substantivos (KOEHN & KNIGHT, 2002) e nomes próprios (FUNG, 1995) ou palavras lexicais (*content words*) (RESNIK & MELAMED, 1997)

4 Pré-processamento dos corpora

Como apresentado no Capítulo 2, um recurso lingüístico indispensável para a extração de regras de tradução usando a abordagem de EBMT e técnicas de Aprendizado de Máquina são os *corpora* paralelos alinhados sentencialmente. Além disso, quando alinhados lexicalmente, esses recursos representam toda a informação necessária para a indução automática de léxicos bilíngües. Portanto, antes de iniciar a implementação das técnicas de indução propriamente ditas, é necessário preparar os recursos lingüístico-computacionais utilizados por elas.

O uso de *corpora* paralelos como fonte de conhecimento lingüístico é uma prática comum em diversas áreas de PLN; contudo, para que um *corpus* paralelo seja realmente útil para a tarefa em questão, alguns cuidados devem ser tomados na sua construção. Primeiro, deve-se delimitar gênero e domínio dos textos que formarão o *corpus* paralelo e, em seguida, coletar os textos paralelos que satisfiçam essas condições. Nesse sentido, no projeto ReTraTos, optou-se por utilizar textos de gênero jornalístico, de boa procedência (diminuindo, assim, a possibilidade de traduções ou textos originais de má qualidade) e provenientes de um domínio acadêmico-científico (no qual as traduções tendem a ser mais literais), porém, não restritos a uma determinada área.

Com relação aos idiomas dos textos paralelos, no projeto ReTraTos, decidiu-se lidar com três idiomas – português do Brasil (`pt`), inglês (`en`) e espanhol (`es`) – combinados em dois pares de tradução envolvendo o `pt`: um par com línguas mais próximas (`pt`–`es`) e outro com línguas mais distantes (`pt`–`en`).

Assim, o *corpus* resultante dessa compilação inicial está composto por artigos da revista científica *Pesquisa FAPESP*[1] escritos originalmente em `pt` e traduzidos para `en` e `es`. Esse conjunto de textos paralelos recebeu a denominação de `CorpusFAPESP`.

O `CorpusFAPESP`, na verdade, é composto por dois *corpora* paralelos: um para o par `pt`–`es`, com 645 pares de textos paralelos totalizando 1.050.924 *tokens* (504.130 em `pt` e

[1]URL da versão online da revista *Pesquisa FAPESP*: `http://revistapesquisa.fapesp.br`.

546.794 em es); e outro para o par pt–en, com 646 pares de textos paralelos e 1.038.638 *tokens* (504.387 em pt e 534.251 em en).[2] As Tabelas 8 e 9 apresentam os números de *tokens*, *types* e sentenças nos *corpora* pt–es e pt–en, respectivamente.[3]

Tabela 8: Quantidade de *tokens*, *types* e sentenças no CorpusFAPESP pt–es original

Idioma	*tokens*	*types*	sentenças
pt	504.130	31.331	18.305
es	546.794	32.568	18.480
Total	**1.050.924**	**63.899**	**36.785**

Tabela 9: Quantidade de *tokens*, palavras e sentenças no CorpusFAPESP pt–en original

Idioma	*tokens*	*types*	sentenças
pt	504.387	31.345	18.313
en	534.251	23.520	17.583
Total	**1.038.638**	**54.865**	**35.896**

Mais especificamente, o CorpusFAPESP conta com artigos de 9 seções: ciência (205), editorial (11), estratégias (136/137)[4], humanidades (40), linha de produção (111), memória (11), opinião (4), política (54) e tecnologia (73); escritos em estilos que vão desde relato de projetos (o estilo mais freqüente) até entrevistas com pesquisadores, todos dissertando sobre diversas áreas de pesquisa.

Com relação ao tamanho dos textos, embora o número médio de *tokens* no CorpusFAPESP pt–es seja 781 para textos em pt e de 847 para textos em es, constatou-se uma grande variedade de tamanho nos textos em pt que variam de 50 a 4.520 *tokens* e, em es, de 60 a 4.823 *tokens*. O *token* mais freqüente em pt é , (34.126 ocorrências) e o menos freqüente é *gaviões* (1 ocorrência); em es o *token* mais freqüente é *de* (40.845 ocorrências) e o menos freqüente é *expelió* (1 ocorrência).

A versão pt–en do CorpusFAPESP possui, em média, 780 *tokens* em pt e 827 *tokens* em en e a mesma variedade de tamanho dos textos em pt encontrada no *corpus* pt–es e de 54 a 4.927 *tokens* nos textos em en. Os *tokens* mais e menos freqüentes em pt também são os mesmos da versão pt–es enquanto o *token* mais freqüente em en é *the* (45.478 ocorrências) e o menos freqüente é *stereoscopic* (1 ocorrência).

[2] A diferença no número de textos nos dois *corpora* paralelos que compõem o CorpusFAPESP se deve ao fato de que um dos textos em português não foi traduzido para o espanhol como o esperado, uma vez que o conteúdo da versão em espanhol é o mesmo do original em português.

[3] Neste trabalho, o conceito *token* é usado para designar qualquer seqüência de caracteres delimitada por espaços (ou começo ou fim de sentença) enquanto *type* é usado para se referir a um *token* independentemente do número de vezes em que ele ocorre no *corpus*. Por exemplo, na seqüência "a um o , um , do a" há 8 *tokens* e 5 *types*.

[4] O CorpusFAPESP pt–en possui um texto a mais da seção estratégia do que o CorpusFAPESP pt–en, num total de 257 *tokens*.

Outra constatação interessante a respeito do tamanho dos textos nos dois *corpora* paralelos foi a de que em ambos, geralmente, os textos em `pt` são menores do que suas versões em `es` e em `en`, o que pode ser constatado pelas quantidades de *tokens* nos originais e nas traduções apresentadas nas Tabelas 8 e 9. Devido às características dos idiomas estudados no projeto ReTraTos, já era esperado que os textos em `pt` fossem menores do que suas versões em `es`, mas não se esperava que o mesmo ocorresse para os textos em `en`. Contudo, verificou-se que nesses textos, em muitos casos, os termos são apresentados no idioma original (`pt`) e, em seguida, traduzidos para `en`, ou há um maior número de palavras inseridas pelo tradutor para explicar conceitos conhecidos pelo público brasileiro mas, talvez, desconhecidos por outros públicos.

Após a coleta dos textos paralelos (em estado "bruto"), estes foram processados com o intuito de adicionar informações úteis para a indução das regras de tradução e dos léxicos bilíngües. Esse processo de "enriquecimento" dos textos pode ser realizado durante o processo de indução, porém, no projeto ReTraTos, optou-se por efetuá-lo como um passo prévio à indução de regras de tradução e de léxicos bilíngües, como um pré-processamento.

As próximas seções apresentam as tarefas de pré-processamento realizadas com o `CorpusFAPESP` – alinhamento sentencial (seção 4.1), etiquetação morfossintática (seção 4.2) e alinhamento lexical (seção 4.3) –, juntamente com as ferramentas computacionais implementadas ou adaptadas para desempenhá-las.

4.1 Alinhamento sentencial

O alinhamento sentecial de dois textos paralelos é o processo no qual são estabelecidas as correspondências entre as sentenças do texto fonte e as sentenças do texto alvo. O alinhamento sentencial dos textos paralelos que compõem o `CorpusFAPESP` foi realizado por meio do alinhador automático `TCAalign` implementado durante o projeto PESA (*Portuguese-English Sentence Alignment*) com base no *Translation* corpus *Aligner* (HOFLAND, 1996).[5] Esse alinhador emprega vários critérios de alinhamento para encontrar as correspondências entre as sentenças fonte e alvo, como listas de palavras âncoras (opcional), palavras com iniciais maiúsculas (candidatas a nomes próprios), caracteres especiais (por exemplo, ! e ?), palavras

[5]Informações a respeito da ferramenta de alinhamento sentencial de textos paralelos utilizada no projeto ReTraTos podem ser obtidas em: `http://www.nilc.icmc.usp.br/projects/aligners.htm`.

cognatas (calculadas por meio de coeficiente de Dice[6] ou LCSR[7]) e tamanho das sentenças (em palavras).

No `TCAalign`, uma estrutura de programação dinâmica é usada para determinar o melhor alinhamento entre as sentenças fonte e alvo com base nos critérios mencionados anteriormente. Os textos alinhados são mantidos em arquivos separados nos quais são inseridas etiquetas e atributos com indicações de alinhamento.

O alinhamento sentencial das sentenças que compõem os *corpora* paralelos `pt`–`es` e `pt`–`en` foi realizado separadamente para cada *corpus*, uma vez que o alinhamento de uma sentença em `pt` e sua tradução para `es` pode não ser o mesmo alinhamento da sentença em `pt` com sua tradução para `en`. A Tabela 10 apresenta um exemplo de três sentenças, uma em cada um dos idiomas estudados no ReTraTos, após o processo de alinhamento sentencial, no qual a correspondência entre elas está indicada pelo mesmo valor do atributo `snum` nas etiquetas de início de sentenças <s>. Neste caso, o mesmo alinhamento sentencial para a sentença em `pt` e sua tradução para `es` foi encontrado para esta sentença em `pt` e sua tradução para `en`.

Tabela 10: Exemplo de uma sentença em `pt` e suas correspondentes em `es` e `en` após alinhamento sentencial

`pt`	<s snum=87>Embora o piquiá não esteja sob risco de ser extinto , a exploração descontrolada pode levar ao desaparecimento dessa árvore em algumas regiões . </s>
`es`	<s snum=87>Pese a que el piquiá no se encuentra bajo riesgo de extinción , la explotación desmesurada puede ocasionar su desaparición en algunas regiones . </s>
`en`	<s snum=87>Although pekea is not under any risk of becoming extinct , its uncontrolled exploitation may lead to the disappearance of this tree in some regions . </s>

É importante citar que, após o alinhamento sentencial, as sentenças foram tokenizadas por meio da inserção de espaços antes e depois de caracteres de pontuação (.,;!? etc.), com tratamento especial para alguns caracteres como "." e "," em representações numéricas.

Os 645 textos paralelos do `CorpusFAPESP` `pt`–`es` foram alinhados automaticamente por `TCAalign` sem a utilização de uma lista de palavras âncoras; já os 646 textos paralelos do `CorpusFAPESP` `pt`–`en` foram alinhados automaticamente por `TCAalign` utilizando a lista

[6] O coeficiente de Dice de duas palavras é computado, nesse caso, dividindo-se a quantidade de bigramas em comum nas duas palavras multiplicado por 2, pela soma das quantidades de bigramas nas duas palavras. Por exemplo, o coeficiente de Dice da palavra em `pt` *alinhamento* e da palavra em `es` *alineamiento* é $\frac{2\times7}{(10+11)} \simeq 0,67$ uma vez que os bigramas são *al-li-in-nh-ha-am-me-en-nt-to* e *al-li-in-ne-ea-am-mi-ie-en-nt-to*, respectivamente (os bigramas em comum nas duas palavras aparecem sublinhados).

[7] A LCSR (*Longest Common Subsequence Ratio*) de duas palavras é computada dividindo-se o tamanho da maior subseqüência em comum pelo tamanho da maior palavra. Por exemplo, a LCSR da palavra em `pt` *alinhamento* e da palavra em `es` *alineamiento* é $\frac{10}{12} \simeq 0,83$ uma vez que a maior subseqüência comum é *a-l-i-n-a-m-e-n-t-o*.

de palavras âncoras pt–en gerada no projeto PESA. Ambos os *corpora* paralelos foram alinhados usando LCSR como medida de cognato com o limite mínimo padrão definido na ferramenta, 0,65.[8] Detalhes sobre o processo de alinhamento sentencial desempenhado por TCAalign podem ser obtidos em (CASELI, 2003).

Após o alinhamento sentencial automático, uma verificação manual foi realizada com o intuito de corrigir possíveis erros do alinhador e, para tanto, apenas os alinhamentos diferentes de 1 : 1 foram verificados. Como resultado desse processo de correção manual, foram obtidos dois *corpora*: um com 18.314 alinhamentos sentenciais pt–es e outro com 18.275 alinhamentos sentenciais pt–en. A Tabela 11 apresenta as quantidades (#) e as porcentagens (%) de cada tipo de alinhamento sentencial nos *corpora* pt–es e pt–en.

Tabela 11: Tipos de alinhamento sentencial no CorpusFAPESP pt–es e pt–en após a verificação manual dos alinhamentos gerados automaticamente

	pt–es		pt–en	
Tipo	#	%	#	%
1 : 1	18.006	98,32	17.174	93,97
0 : 1	45	0,24	73	0,40
1 : 0	33	0,18	805	4,40
1 : 2	190	1,04	111	0,61
1 : 3	3	0,02	1	0,01
2 : 1	34	0,18	111	0,61
2 : 2	3	0,02	–	–
TOTAL	**18.314**	**100**	**18.275**	**100**

Como se pode perceber pelos dados da Tabela 11, a maioria dos alinhamentos sentencias é do tipo 1 : 1: 98,32% em pt–es e 93,97% em pt–en. As omissões (0 : 1 ou 1 : 0) representam 0,42% dos alinhamentos em pt–es e 4,80% em pt–en[9] enquanto os alinhamentos restantes – 1,26% em pt–es e 1,23% em pt–en – são aqueles que envolvem mais de uma sentença em um ou ambos os lados do alinhamento (1 : 2, 1 : 3, 2 : 1 ou 2 : 2).

Os *corpora* com os alinhamentos sentenciais corrigidos manualmente foram, então, utilizados como referência na avaliação do alinhamento sentencial automático produzido por TCAalign por meio do cálculo de três medidas: precisão, cobertura e medida-F. Essas três medidas são calculadas com base nas equações (2.14), (2.15) e (2.16), respectivamente, apresentadas na subseção 2.4.4 do Capítulo 2, nas quais *candidatos* são os alinhamentos

[8] O limite mínimo para a medida de cognato LCSR foi determinado empiricamente com base na análise de exemplos positivos e negativos de palavras cognatas nos pares de idiomas pt–es e pt–en.

[9] O alto número de omissões no alinhamento sentencial do par pt–en se deve ao fato de que, em muitos textos desse *corpus*, as sentenças no final dos arquivos em português não foram traduzidas para o inglês resultando, assim, em vários alinhamentos de omissão do tipo 1 : 0 (uma sentença em português sem correspondência no texto em inglês).

sentenciais retornados por TCAalign e *referência*, os alinhamentos do *corpus* de referência. Os resultados dessa avaliação são apresentados na Tabela 12.

Tabela 12: Avaliação do alinhamento sentencial automático de TCAalign para os *corpora* pt–es e pt–en

Medida	precisão	cobertura	medida-F
pt–es	93,01%	95,85%	94,41%
pt–en	97,10%	98,23%	97,66%

De acordo com os valores da Tabela 12, TCAalign apresentou melhor desempenho para o par pt–en do que para o par pt–es, o que pode ser explicado pelos fatos descritos a seguir. Embora TCAalign seja independente de língua, durante seu desenvolvimento no projeto PESA, seus parâmetros foram definidos empiricamente para o par pt–en e usados, no projeto ReTraTos, sem muitas alterações para o par pt–es. Além disso, no alinhamento das sentenças do par pt–en, TCAalign dispunha de mais informação lingüística do que no alinhamento do par pt–es; essa informação lingüística está presente na lista de palavras âncoras gerada como co-produto do projeto PESA.

Por fim, dos 18.314 alinhamentos sentenciais presentes no CorpusFAPESP pt–es, eliminou-se os 78 casos de omissão – uma vez que não representam exemplos de tradução – resultando em um conjunto final composto por 18.236 exemplos de tradução com 1.049.462 *tokens* (503.596 em pt e 545.866 em es).[10] De maneira semelhante, dos 18.275 alinhamentos sentenciais presentes no CorpusFAPESP pt–en, eliminou-se os 878 casos de omissão resultando em um conjunto final de 17.397 exemplos de tradução com 1.026.512 *tokens* (494.391 em pt e 532.121 em en). As Tabelas 13 e 14 apresentam os números de *tokens*, *types* e sentenças nos *corpora* pt–es e pt–en, respectivamente.

Tabela 13: Quantidade de *tokens*, *types* e sentenças no CorpusFAPESP pt–es alinhado sentencialmente

Idioma	*tokens*	*types*	sentenças
pt	503.596	31.318	18.236
es	545.866	32.539	18.236
Total	**1.049.462**	**63.857**	**36.472**

Esses dois conjuntos de 18.236 exemplos de tradução pt–es e de 17.397 exemplos de tradução pt–en foram etiquetados morfossintaticamente como apresentado na próxima seção.

[10] Os alinhamentos sentenciais que envolviam mais de uma sentença em um ou ambos os lados foram tratados concatenando-se as sentenças de cada lado separando-as por um espaço.

Tabela 14: Quantidade de *tokens*, *types* e sentenças no `CorpusFAPESP pt–en` alinhado sentencialmente

Idioma	*tokens*	*types*	sentenças
pt	494.391	30.974	17.397
en	532.121	23.466	17.397
Total	**1.026.512**	**54.440**	**34.794**

4.2 Etiquetação morfossintática

Depois de determinar as correspondências entre as sentenças dos *corpora* paralelos que formam o `CorpusFAPESP` e eliminar aquelas que não possuíam correspondência alguma, procedeu-se com a etiquetação morfossintática dos exemplos de tradução resultantes. O processo de etiquetação morfossintática atribui, a cada palavra, a categoria e os traços morfossintáticos mais adequados considerando-se o contexto no qual tal palavra está inserida.

Para tanto, foram utilizadas as ferramentas presentes no tradutor automático `Apertium` (ARMENTANO-OLLER et al., 2006) com dados lingüísticos para os idiomas `pt`, `es` – dicionários morfológicos do pacote de dados lingüísticos `es`–`pt` (versão 0.9) – e `en` – dicionário morfológico do pacote de dados lingüísticos `en`–`ca` (versão 0.8) – incrementados com novas entradas (informações morfológicas) conforme descrito a seguir.[11]

Os dicionários morfológicos do `Apertium` para os idiomas `pt` e `en` foram aumentados com o auxílio dos dicionários eletrônicos do `Unitex` (PAUMIER, 2006). O `Unitex` é uma coleção de recursos e ferramentas lingüísticas (dicionários eletrônicos, gramáticas etc.) usados para a análise de textos em linguagem natural. Os dicionários eletrônicos do `Unitex` especificam as palavras simples e compostas de uma língua juntamente com seus lemas e um conjunto de códigos gramaticais (semânticos e flexionais). Esses dicionários estão disponíveis para vários idiomas entre eles Inglês, Espanhol e Português.[12]

Assim, ao dicionário morfológico do `Apertium` para o `pt` foram acrescentadas novas entradas provenientes do `Unitex`-PB (MUNIZ, 2004) aumentando a cobertura do dicionário original de 128.772 para 1.136.536 formas superficiais.[13] O dicionário morfológico para `en` também foi aumentado com novas entradas obtidas com o auxílio de uma ferramenta de análise morfossintática desenvolvida durante o projeto ReTraTos, a `anali`, a qual se baseia

[11] Informações sobre o tradutor `Apertium`, bem como os pacotes com os dados lingüísticos utilizados neste projeto, podem ser obtidos em `http://www.apertium.org`.

[12] A ferramenta de processamento de *corpus* `Unitex`, bem como os dicionários eletrônicos, podem ser obtidos em `http://www-igm.univ-mlv.fr/~unitex`.

[13] A construção dos dicionários eletrônicos no formato do `Unitex` para o Português do Brasil, o `Unitex`-PB, foi resultado de um projeto de mestrado desenvolvido no NILC. Para mais informações consulte (MUNIZ, 2004) e `http://www.nilc.icmc.usp.br/nilc/projects/unitex-pb/web/index.html`.

nos dicionários eletrônicos de `Unitex` para retornar as possíveis análises de cada palavra (CASELI & NUNES, 2006). Com a inserção dessas novas entradas, a cobertura do dicionário morfológico para `en` aumentou de 48.759 para 61.601. Por fim, para o idioma `es`, ao dicionário morfológico do `Apertium` foram acrescentadas as entradas provenientes do dicionário morfológico usado no tradutor `interNOSTRUM` (CANALS-MAROTE et al., 2001), aumentando sua cobertura de 116.804 para 337.861 formas superficiais.[14]

O sistema de TA `Apertium` baseia-se na estratégia de transferência parcial, na qual o processo de tradução automática palavra-a-palavra é incrementado com um processamento lexical robusto (que trata expressões multipalavras e desambigüiza adequadamente palavras ambíguas) e um processamento estrutural local baseado em regras simples e bem formuladas para algumas transformações estruturais simples (como reordenamento e concordância) (GARRIDO-ALENDA et al., 2004). Assim, o `Apertium` está composto por 8 módulos que se comunicam por meio de arquivos de texto e executam as tarefas apresentadas a seguir, nessa ordem:

1. `desformatação` – o texto a ser traduzido é separado da informação de formatação que o acompanha;

2. `análise morfológica` – o texto é dividido em formas superficiais (itens lexicais) e para cada uma delas é retornada uma ou mais formas lexicais formadas por lema, categoria lexical e informação de flexão morfológica. Nesse processo de divisão em *tokens* também são tratados os casos de contração (por exemplo, $do = de+o$) e expressões multipalavras (por exemplo, *no entanto*) que podem, inclusive, aparecer flexionadas (por exemplo, *dava na vista*). Esse módulo é compilado a partir de um dicionário morfológico da língua fonte (GARRIDO-ALENDA et al., 1999, 2002 apud GARRIDO-ALENDA et al., 2004);

3. `desambiguação categorial` – os itens lexicais com mais de uma categorização possível são tratados por um etiquetador baseado em um modelo de Markov escondido (*Hidden Markov Model* ou HMM) o qual atribui a melhor forma lexical de acordo com as formas lexicais possíveis para as palavras vizinhas;

4. `transferência lexical` – o módulo de transferência lexical é chamado pelo módulo de transferência estrutural para transferir da forma lexical fonte para a forma lexical alvo correspondente, baseando-se em um léxico bilíngüe;

[14] O dicionário morfológico para o `es` desenvolvido para o tradutor `interNOSTRUM` foi cedido pelo grupo desenvolvedor de tal sistema, o grupo de TA Transducens da Universidade de Alicante (UA), Espanha.

5. `transferência estrutural` – o módulo de transferência estrutural realiza casamento de padrões baseado em estados finitos para detectar e tratar os padrões de formas lexicais que necessitam um tratamento especial por representarem divergências gramaticais entre as línguas fonte e alvo. Esse módulo é compilado a partir de um arquivo de regras de transferência geradas manualmente (GARRIDO-ALENDA & FORCADA, 2001);

6. `geração morfológica` – as formas superficiais correspondentes a cada forma lexical alvo são retornadas. Esse módulo é compilado a partir de um dicionário morfológico da língua alvo;

7. `pós-geração` – operações ortográficas são aplicadas às formas superficiais (como contrações) com base em um arquivo de regras;

8. `reformatação` – a informação de formatação existente, originalmente, no texto fonte é recuperada no texto traduzido.

Desses módulos, apenas os três primeiros são utilizados para etiquetar morfossintaticamente o *corpus* `pt`–`es` usado no projeto ReTraTos: (1) desformatação, (2) análise morfológica e (3) desambiguação categorial. Além disso, como mencionado anteriormente, o módulo de análise morfológica não utiliza os autômatos distribuídos com os dados lingüísticos `es`–`pt` (versão 0.9) e `en`–`ca` (versão 0.8), mas, sim, os autômatos gerados no ReTraTos a partir dos dicionários morfológicos do `Apertium` incrementados com as entradas de `Unitex` (`pt` e `en`) e `interNOSTRUM` (`es`). Assim, a cobertura dos etiquetadores gerados, no ReTraTos, para os idiomas `pt`, `es` e `en` são, respectivamente: 1.136.536, 337.861 e 61.601 formas superficiais.

A Tabela 15 apresenta um exemplo de sentenças `pt`, `es` e `en` após a etiquetação morfossintática, nas quais cada *token* possui uma etiqueta de PoS e zero ou mais atributos dessa etiqueta, todos delimitados pelos caracteres "<" e ">". A lista completa com todos os símbolos gramaticais utilizados no projeto ReTraTos para representar categorias e traços morfossintáticos pode ser consultada no Apêndice A.

Outra consideração importante a respeito da etiquetação morfossintática é que as palavras desconhecidas são identificadas inserindo-se um caractere "*" no seu início, como é o caso da palavra *piquiá* no exemplo da Tabela 15. Além disso, algumas palavras podem ser divididas em várias com o intuito de desfazer a contração que elas representam. Por exemplo, a palavra em `pt` *ao* foi dividida e etiquetada como *a*<pr>+*o*<det><def><m><sg>.

Outra alteração realizada pelo analisador morfológico diz respeito à união de palavras para formar uma unidade multipalavra. Nesse caso, a união é simbolizada pelo caractere

Tabela 15: Exemplo de uma sentença em pt e suas correspondentes em es e en após etiquetação morfossintática

pt	<s snum=87>Embora/Embora<cnjadv> o/o<det><def><m><sg> *piquiá/piquiá não/não<adv> esteja/estar<vblex><prs><p3><sg> sob/sob<pr> risco/risco<n><m><sg> de/de<pr> ser/ser<vbser><inf> extinto/extinto<adj><m><sg> ,/,<cm> a/o<det><def><f><sg> exploração/exploração<n><f><sg> descontrolada/descontrolado<adj><f><sg> pode/poder<vbmod><pri><p3><sg> levar/levar<vblex><inf> ao/a<pr>+o<det><def><m><sg> desaparecimento/desaparecimento<n><m><sg> dessa/de<pr>+esse<det><dem><f><sg> árvore/árvore<n><f><sg> em/em<pr> algumas/algum<det><ind><f><pl> regiões/região<n><f><pl> ./.<sent> </s>
es	<s snum=87>Pese_a/Pese_a<pr> que/que<cnjsub> el/el<det><def><m><sg> *piquiá/piquiá no/no<adv> se/se<prn><pro><ref><p3><mf><sp> encuentra/encontrar<vblex><pri><p3><sg> bajo/bajo<pr> riesgo/riesgo<n><m><sg> de/de<pr> extinción/extinción<n><f><sg> ,/,<cm> la/el<det><def><f><sg> explotación/explotación<n><f><sg> desmesurada/desmesurado<adj><f><sg> puede/poder<vbmod><pri><p3><sg> ocasionar/ocasionar<vblex><inf> su/suyo<det><pos><mf><sg> desaparición/desaparición<n><f><sg> en/en<pr> algunas/alguno<det><ind><f><pl> regiones/región<n><f><pl> ./.<sent> </s>
en	<s snum=87>Although/Although<cnjadv> *pekea/pekea is/be<vbser><pri><p3><sg> not/not<adv> under/under<pr> any/any<det><ind><sp> risk/risk<n><sg> of/of<pr> becoming/become<vblex><ger> extinct/extinct<adj> ,/,<cm> its/its<det><pos><sp> uncontrolled/uncontrolled<adj> exploitation/exploitation<n><sg> may/may<vaux><inf> lead_to/lead<vblex><inf>_to the/the<det><def><sp> disappearance/disappearance<n><sg> of/of<pr> this/this<det><dem><sg> tree/tree<n><sg> in/in<pr> some/some<det><qnt><sp> regions/region<n><pl> ./.<sent> </s>

"_" como ocorre, por exemplo, com a seqüência de palavras em es *Pese a* etiquetada como *Pese_a*<pr> e em en *lead to* etiquetada como *lead*<vblex><inf>*_to* (um exemplo em pt seria *São Paulo*, etiquetada como *São_Paulo*<np><loc>).

4.3 Alinhamento lexical

Por fim, a última tarefa de pré-processamento desempenhada com os exemplos de tradução que formam o CorpusFAPESP foi o alinhamento lexical. O alinhamento lexical de dois textos paralelos é o processo no qual são estabelecidas as correspondências entre as palavras do texto fonte e as palavras do texto alvo.

Para essa tarefa, ferramentas distintas foram utilizadas para o alinhamento lexical dos *corpora* pt–es – o alinhador lexical LIHLA (CASELI et al., 2005) – e pt–en – o alinhador lexical GIZA++ (OCH & NEY, 2000b).

A opção de utilizar ferramentas diferentes para o alinhamento lexical dos dois *corpora* paralelos foi tomada após a realização de experimentos com amostras de cada *corpus*

nos quais LIHLA se saiu melhor do que GIZA++ no alinhamento do par pt–es mas não no do par pt–en. Uma breve descrição de cada uma dessas ferramentas e de seus desempenhos nos experimentos citados é apresentada nas subseções a seguir para os *corpora* pt–es (subseção 4.3.1) e pt–en (subseção 4.3.2).

4.3.1 Alinhamento lexical do *corpus* paralelo pt–es

O alinhador lexical LIHLA (*Language-Independent Heuristics Lexical Aligner*), desenvolvido durante o projeto ReTraTos, utiliza um léxico bilíngüe probabilístico – gerado por NATools[15] – e heurísticas independentes de língua para determinar o melhor alinhamento entre palavras, unidades multipalavras e símbolos de pontuação. Além disso, o usuário pode selecionar o que deseja priorizar na escolha da melhor candidata ao alinhamento (posição na sentença alvo ou no léxico bilíngüe) e, assim, garantir a melhor configuração para línguas que preservam ou não a mesma ordem das palavras na tradução. Para línguas cuja ordem das palavras tende a ser a mesma no original e na tradução – como, por exemplo, pt e es –, deve-se priorizar a candidata mais bem posicionada na sentença alvo; enquanto para as línguas que não possuem uma correspondência direta com relação à posição de suas palavras – como é o caso de es e basco (euskera, eu) – deve-se priorizar as melhores traduções presentes no léxico.[16]

LIHLA alinha os *tokens* fonte e alvo em um processo de dois passos. No primeiro, os *tokens* são alinhados de acordo com seu tipo: os símbolos de pontuação são alinhados entre si priorizando-se os idênticos, enquanto as palavras são alinhadas aplicando-se três heurísticas independentes de língua explicadas a seguir. Esse processo se repete até que nenhum novo alinhamento seja produzido ou um número máximo de iterações (10, por padrão) seja atingido.

A cada alinhamento gerado por LIHLA, é atribuída uma probabilidade que indica quão confiável é este alinhamento. Por exemplo, se uma palavra fonte s_j é alinhada com uma palavra alvo t_i com probabilidade 1 isto significa que t_i é a melhor candidata para s_j e vice-versa ou então t_i é a única candidata disponível. Quanto maior a probabilidade (que

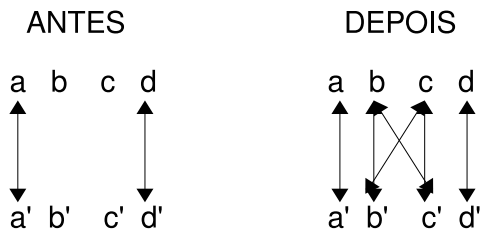
[15] NATools é um conjunto de ferramentas desenvolvidas para trabalhar com *corpora* paralelos, que está disponível livremente em: http://natura.di.uminho.pt/natura/natura/ sob as especificações da *GNU General Public License*.

[16] LIHLA apresentou bons resultados no alinhamento de pares de línguas nos quais a ordem das palavras no original não é a mesma na tradução como constatado nos experimentos realizados com o par es–eu – no qual LIHLA obteve 6% menos AER do que GIZA++ (CASELI et al., 2005) – e inglês–inuktitut – no qual LIHLA foi o sistema com melhor medida-F (57,07%) na competição de alinhadores realizadas no *Workshop of Building and Using Parallel Texts* realizado em conjunto com a conferência da ACL 2005 (MARTIN et al., 2005).

varia de 0 a 1), mais confiável é o alinhamento.

Além disso, para evitar que alinhamentos errados sejam gerados logo nas primeiras iterações do método (uma vez que todos os alinhamentos subseqüentes se baseiam naqueles gerados previamente) um filtro de freqüência é aplicado. Primeiro, são alinhadas as palavras com freqüência menor do que um determinado limite (o limite é dado por aquelas palavras que, juntas, representam 30% da freqüência de todas as palavras do léxico) e, em seguida, alinham-se as demais palavras e os caracteres de pontuação.

No último passo do alinhamento (que é opcional), `LIHLA` alinha os *tokens* de mesmo tipo que permaneceram não alinhados e estão limitados por *tokens* já alinhados gerando um alinhamento $n : m$ envolvendo os n *tokens* fonte e os m *tokens* alvo. Por exemplo, na seqüência apresentada a seguir o *token a* está alinhado com o *token a'* e o *token d* com o *token d'*, assim, os *tokens* fonte entre *a* e *d* e os *tokens* alvo entre *a'* e *d'* são alinhados entre si:



As três heurísticas aplicadas no alinhamento de palavras são: casamento idêntico, léxico bilíngüe e cognato. No casamento idêntico, busca-se uma palavra alvo idêntica à palavra fonte sendo alinhada e, se esta for encontrada, determina-se um alinhamento $1 : 1$ entre elas.

Caso a primeira heurística não seja satisfeita, `LIHLA` criará um conjunto de candidatas ao alinhamento com a palavra fonte consultando o léxico bilíngüe (segunda heurística) em busca de possíveis traduções que ocorrem na sentença alvo; e a sentença alvo em busca de palavras cognatas (terceira heurística). Uma palavra cognata à palavra fonte é determinada aplicando-se a medida de cognato LCSR com limite mínimo igual a 0,65, como no `TCAalign`. Por meio dessa última heurística, `LIHLA` é capaz de lidar com palavras que não ocorrem no léxico bilíngüe e na sentença alvo ao mesmo tempo. As posições de cada uma dessas candidatas na sentença alvo são armazenadas e `LIHLA` determina qual delas é a melhor considerando o critério de prioridade (posição ou léxico) fornecido pelo usuário.

Por fim, uma multipalavra alvo é buscada verificando-se se as posições vizinhas à melhor candidata alvo também são candidatas à tradução da palavra fonte e, além disso,

não são possíveis traduções de nenhuma vizinha da palavra fonte. De maneira similar, uma multipalavra envolvendo a palavra fonte também é buscada e, como resultado, pode-se obter um alinhamento $n : m$, com $n, m \geq 1$. Mais detalhes sobre o modo de processamento de `LIHLA` podem ser obtidos em (CASELI et al., 2005).

Vários experimentos foram realizados com parte do `CorpusFAPESP` `pt`–`es` – 591 exemplos de tradução, num total de 35.822 *tokens* (17.128 em `pt` e 18.694 em `es`) – com o intuito de avaliar o desempenho de `LIHLA` no alinhamento de formas superficiais e lemas. Assim, dois conjuntos de léxicos bilíngües foram gerados por `NATools` a partir de todos os exemplos de tradução do `CorpusFAPESP`: um com base nas formas superficiais e outro com base nos lemas das palavras retornados como co-produto da etiquetação morfossintática.

Além disso, o desempenho de `LIHLA` foi comparado ao do sistema `GIZA++` (OCH & NEY, 2000b), considerado o estado-da-arte nessa área, por meio das medidas precisão, cobertura e taxa de erro (*Alignment Error Rate* ou AER). O cálculo de precisão e cobertura é realizado com base nas equações (2.14) e (2.15), respectivamente, apresentadas na subseção 2.4.4 do Capítulo 2, nas quais *candidatos* são os alinhamentos lexicais retornados por `LIHLA` e *referência*, os alinhamentos do *corpus* de referência. A taxa de erro, por sua vez, é calculada neste projeto como o complemento da média ponderada de precisão e cobertura (também conhecida como medida-F), como apresentado na equação (4.1).

$$\text{AER} = 1 - 2 \times \frac{\text{precisão} \times \text{cobertura}}{\text{precisão} + \text{cobertura}} \tag{4.1}$$

Tanto `LIHLA` quanto `GIZA++` foram executados duas vezes para a produção de alinhamentos nos dois sentidos – `pt`→`es` e `es`→`pt` – e, em seguida, esses alinhamentos foram simetrizados de acordo com os algoritmos descritos em (OCH & NEY, 2003) resultando na união, na intersecção e no refinamento dos alinhamentos em ambos os sentidos. O método de simetrização que apresentou melhor resultado, em ambas as ferramentas, foi a união. Portanto, os valores da Tabela 16 são referentes à união dos alinhamentos produzidos por `LIHLA` e `GIZA++` nos dois sentidos, com base nas formas superficiais e nos lemas.

Tabela 16: Desempenho de `LIHLA` e `GIZA++` após a união dos alinhamentos `pt`–`es` nos dois sentidos

	Formas superficiais			**Lemas**		
Método	**Precisão**	**Cobertura**	**AER**	**Precisão**	**Cobertura**	**AER**
`GIZA++`	93,70%	93,60%	6,35%	92,29%	94,13%	6,80%
`LIHLA`	93,26%	94,42%	6,17%	**94,25%**	**94,97%**	**5,39%**

Analisando-se os valores da Tabela 16 nota-se que `LIHLA` e `GIZA++` apresentaram pra-

ticamente o mesmo desempenho no alinhamento de formas superficiais e LIHLA foi um pouco melhor do que GIZA++ no alinhamento de lemas. Enquanto LIHLA melhorou seu desempenho no alinhamento de lemas quando comparado ao alinhamento de formas superficiais, GIZA++ apresentou um desempenho um pouco pior. O melhor desempenho de LIHLA no alinhamento de lemas do que no alinhamento de formas superficiais pode ser explicado pela capacidade desse método em formar multipalavras para contrações do tipo *de+o* buscando as possíveis candidatas de cada lema da contração, ou seja, as candidatas de *de* e de *o*, nesse exemplo.

O desempenho de LIHLA no alinhamento do par pt–es, considerando-se a união dos alinhamentos gerados para cada sentido de tradução, é apresentado na Tabela 17 para cada categoria de alinhamento – omissão ($1:0$ ou $0:1$), multipalavra ($n:m$, com n ou $m > 1$) e um-para-um ($1:1$) – separadamente. De acordo com os valores dessa tabela é possível notar que a maior taxa de erro de LIHLA está nos alinhamentos $1:0$ e $0:1$ (AER = 43,13%). Além disso, a taxa de erro dos alinhamentos envolvendo multipalavras (AER = 11,19%) esta próxima à taxa calculada para os alinhamentos um-para-um (AER = 9,29%).

Tabela 17: Desempenho de LIHLA no alinhamento pt–es (lemas e união) em cada categoria de alinhamento

Categoria	**Precisão**	**Cobertura**	**AER**
omissão	51,22%	63,93%	43,13%
multipalavra	91,51%	86,25%	11,19%
um-para-um	91,34%	90,09%	9,29%
todas	**94,25%**	**94,97%**	**5,39%**

A Tabela 18 apresenta um exemplo de um par de sentenças pt–es alinhadas lexicalmente por LIHLA. Após o alinhamento lexical, ao final de cada *token* (já etiquetado morfossintaticamente no passo anterior) é adicionado um número (precedido pelo caractere ":") que indica a posição do *token* (na sentença correspondente) com o qual esse *token* se alinha. Assim, o número 1 se refere ao primeiro *token* da sentença, o 2, ao segundo e assim por diante. O número 0 é usado para indicar um alinhamento de omissão ($1:0$ ou $0:1$), como o do segundo *token* alvo *que*, sem correspondência na sentença fonte apresentada na Tabela 18.

Um alinhamento múltiplo, por sua vez, é representado concatenando-se todas as posições dos *tokens* alvo (fonte) com os quais um *token* fonte (alvo) se alinha separando-as por caracteres "_". Por exemplo, o alinhamento múltiplo $1:2$ do *token* em pt *esteja*, na posição 5, com dois *tokens* em es nas posições 6 (*se*) e 7 (*encuentra*).

Diferentemente do processo de alinhamento sentencial, após o alinhamento lexical, não se realizou nenhuma verificação manual para correção de possíveis erros de LIHLA uma

Tabela 18: Exemplo de um par de sentenças pt–es do CorpusFAPESP após alinhamento lexical produzido por LIHLA

pt	<s snum=87>Embora/Embora<cnjadv>:1 o/o<det><def><m><sg>:3 *piquiá/piquiá:4 não/não<adv>:5 esteja/estar<vblex><prs><p3><sg>:6_7 sob/sob<pr>:8 risco/risco<n><m><sg>:9 de/de<pr>:10 ser/ser<vbser><inf>:11 extinto/extinto<adj><m><sg>:11 ,/,<cm>:12 a/o<det><def><f><sg>:13 exploração/exploração<n><f><sg>:14 descontrolada/descontrolado<adj><f><sg>:15 pode/poder<vbmod><pri><p3><sg>:16 levar/levar<vblex><inf>:17 ao/a<pr>+o<det><def><m><sg>:18 desaparecimento/desaparecimento<n><m><sg>:19 dessa/de<pr>+esse<det><dem><f><sg>:0 árvore/árvore<n><f><sg>:0 em/em<pr>:20 algumas/algum<det><ind><f><pl>:21 regiões/região<n><f><pl>:22 ./.<sent>:23 </s>
es	<s snum=87>Pese_a/Pese_a<pr>:1 que/que<cnjsub>:0 el/el<det><def><m><sg>:2 *piquiá/piquiá:3 no/no<adv>:4 se/se<prn><pro><ref><p3><mf><sp>:5 encuentra/encontrar<vblex><pri><p3><sg>:5 bajo/bajo<pr>:6 riesgo/riesgo<n><m><sg>:7 de/de<pr>:8 extinción/extinción<n><f><sg>:9_10 ,/,<cm>:11 la/el<det><def><f><sg>:12 explotación/explotación<n><f><sg>:13 desmesurada/desmesurado<adj><f><sg>:14 puede/poder<vbmod><pri><p3><sg>:15 ocasionar/ocasionar<vblex><inf>:16 su/suyo<det><pos><mf><sg>:17 desaparición/desaparición<n><f><sg>:18 en/en<pr>:21 algunas/alguno<det><ind><f><pl>:22 regiones/región<n><f><pl>:23 ./.<sent>:24 </s>

vez que tal tarefa seria inviável considerando-se a complexidade da mesma e o tamanho do *corpus*.

4.3.2 Alinhamento lexical do *corpus* paralelo pt–en

O alinhamento dos exemplos de tradução pt–en, por sua vez, foi realizado com o auxílio da ferramenta GIZA++ (OCH & NEY, 2000b). GIZA++ utiliza os modelos estatísticos da IBM (BROWN et al., 1993) e modelo de Markov escondido (HMM) (VOGEL et al., 1996 apud OCH & NEY, 2000b, p. 440) (OCH & NEY, 2000a apud OCH & NEY, 2000b, p. 440) para determinar as melhores correspondências entre *tokens* fonte e *tokens* alvo.

GIZA++ (versão 2.0) foi executado de acordo com sua configuração padrão – na qual estão incluídas iterações dos modelos IBM-1, IBM-3, IBM-4 e HMM – e treinado com base no *corpus* completo de 17.397 exemplos de tradução. Os modelos utilizados por GIZA++, em sua configuração padrão, variam no modo como a probabilidade do alinhamento – $Pr({f_1}^S, {a_1}^S|{e_1}^T)$ na qual ${a_1}^S$ é um alinhamento que descreve o mapeamento da palavra fonte f_j na palavra alvo e_{a_j} considerando-se que ${f_1}^S$ é uma *string* fonte e ${e_1}^T$, uma *string* alvo – é calculada. Por exemplo, no modelo IBM-1, todos os alinhamentos têm a mesma probabilidade. O modelo HMM, por sua vez, usa um modelo de primeira ordem $p(a_j|a_{j-1})$ no qual a posição do alinhamento a_j depende da posição do alinhamento anterior a_{j-1}. A partir

do modelo IBM-3, um modelo de fertilidade $p(\phi|e)$ é adicionado ao cálculo da probabilidade. Esse modelo descreve o número de palavras ϕ alinhadas com a palavra alvo e.

Experimentos foram realizados com uma amostra do *corpus* `pt`–`en` – 576 exemplos de tradução, num total de 35.673 *tokens* (17.239 em `pt` e 18.434 em `en`) resultando nos valores apresentados na Tabela 19.

Tabela 19: Desempenho de `LIHLA` e `GIZA++` após a união dos alinhamentos `pt`–`en` nos dois sentidos

	Formas superficiais			**Lemas**		
Método	**Precisão**	**Cobertura**	**AER**	**Precisão**	**Cobertura**	**AER**
`GIZA++`	**90,47%**	**92,34%**	**8,61%**	89,42%	92,77%	8,94%
`LIHLA`	82,82%	86,38%	15,44%	82,02%	85,52%	16,27%

Como se pode notar pelos valores da Tabela 19, `GIZA++` teve um desempenho muito melhor do que `LIHLA` tanto no alinhamento de lemas quanto no alinhamento de formas superficiais. É interessante notar que, no alinhameno lexical do par `pt`–`en`, ambos os alinhadores tiveram pior desempenho no alinhamento de lemas do que no alinhamento de formas superficiais. Devido ao melhor desempenho de `GIZA++`, para esse par de línguas, optou-se por utilizá-la no alinhamento das formas superficiais dos exemplos de tradução `pt`–`en`.

O desempenho de `GIZA++` no alinhamento do par `pt`–`en`, considerando-se a união dos alinhamentos gerados para cada sentido de tradução, é apresentado na Tabela 20 para cada categoria de alinhamento – omissão ($1:0$ ou $0:1$), multipalavra ($n:m$, com n ou $m > 1$) e um-para-um ($1:1$) – separadamente. De acordo com os valores dessa tabela é possível notar que a maior taxa de erro de `LIHLA` está nos alinhamentos $1:0$ e $0:1$ (AER = 55,17%). Além disso, a taxa de erro dos alinhamentos envolvendo multipalavras (AER = 15,71%) foi menor do que a taxa calculada para os alinhamentos um-para-um (AER = 17,55%).

Tabela 20: Desempenho de `GIZA++` no alinhamento `pt`–`en` (formas superficiais e união) em cada categoria de alinhamento

Categoria	**Precisão**	**Cobertura**	**AER**
omissão	37,56%	55,60%	55,17%
multipalavra	87,65%	81,17%	15,71%
um-para-um	81,10%	83,84%	17,55%
todas	**90,47%**	**92,34%**	**8,61%**

A Tabela 21 apresenta um exemplo de um par de sentenças `pt`–`en` alinhadas lexicalmente por `GIZA++`. Como mencionando anteriormente, o alinhamento lexical de cada *token* (já etiquetado morfossintaticamente no passo anterior) é indicado por um número (precedido pelo caractere ":") que indica a posição do *token* (na sentença correspondente) com o qual

esse *token* se alinha. Um exemplo de alinhamento de omissão apresentado na Tabela 21 é o do segundo *token* em `pt` (*o*) e um exemplo de multipalavra é o alinhamento 2 : 1 dos *tokens* em `pt` *levar* e *ao* com a multipalavra em `es` *lead_to*.

Tabela 21: Exemplo de um par de sentenças `pt`–`en` do `CorpusFAPESP` após alinhamento lexical produzido por `GIZA++`

`pt`	<s snum=87>Embora/Embora<cnjadv>:1 o/o<det><def><m><sg>:0 *piquiá/piquiá:2 não/não<adv>:4 esteja/estar<vblex><prs><p3><sg>:3 sob/sob<pr>:5 risco/risco<n><m><sg>:7 de/de<pr>:8 ser/ser<vbser><inf>:6_9_10 extinto/extinto<adj><m><sg>:6_9_10 ,/,<cm>:11 a/o<det><def><f><sg>:12 exploração/exploração<n><f><sg>:14 descontrolada/descontrolado<adj><f><sg>:13 pode/poder<vbmod><pri><p3><sg>:15 levar/levar<vblex><inf>:16 ao/a<pr>+o<det><def><m><sg>:16 desaparecimento/desaparecimento<n><m><sg>:18 dessa/de<pr>+esse<det><dem><f><sg>:20 árvore/árvore<n><f><sg>:21 em/em<pr>:22 algumas/algum<det><ind><f><pl>:23 regiões/região<n><f><pl>:24 ./.<sent>:25 </s>
`en`	<s snum=87>Although/Although<cnjadv>:1 *pekea/pekea:3 is/be<vbser><pri><p3><sg>:5 not/not<adv>:4 under/under<pr>:6 any/any<det><ind><sp>:9_10 risk/risk<n><sg>:7 of/of<pr>:8 becoming/become<vblex><ger>:9_10 extinct/extinct<adj>:9_10 ,/,<cm>:11 its/its<det><pos><sp>:12 uncontrolled/uncontrolled<adj>:14 exploitation/exploitation<n><sg>:13 may/may<vaux><inf>:15 lead_to/lead<vblex><inf>_to:16_17 the/the<det><def><sp>:0 disappearance/disappearance<n><sg>:18 of/of<pr>:0 this/this<det><dem><sg>:19 tree/tree<n><sg>:20 in/in<pr>:21 some/some<det><qnt><sp>:22 regions/region<n><pl>:23 ./.<sent>:24 </s>

Os conjuntos de exemplos de tradução obtidos a partir do `CorpusFAPESP` representam a única fonte de conhecimento lingüístico disponível para a indução de regras de tradução e dos léxicos bilíngües `pt`–`es` e `pt`–`en` no projeto ReTraTos, como apresentado nos próximos capítulos.

5 Processo de indução no projeto ReTraTos

Após o levantamento bibliográfico sobre a indução de regras de tradução e de léxicos bilíngües e a apresentação das tarefas de pré-processamento realizadas com os *corpora* `pt`–`es` e `pt`–`en`, este capítulo apresenta os sistemas de indução de regras de tradução e de léxicos bilíngües desenvolvidos no projeto ReTraTos. Antes de apresentar as técnicas utilizadas na indução no ReTraTos, são especificados os formalismos de representação dos exemplos de tradução (seção 5.1.1), do léxico bilíngüe (seção 5.1.2) e das regras de tradução (seção 5.1.3) utilizados neste trabalho.

Considerando-se que os léxicos bilíngües são recursos fundamentais para a TA, decidiu-se implementar um sistema capaz de induzí-lo a partir dos mesmos exemplos de tradução usados na indução das regras. O processo de indução de léxicos bilíngües no ReTraTos é o tema da seção 5.2. Na seção 5.3, descreve-se o processo de indução das regras de tradução de acordo com as etapas apresentadas no Capítulo 2: identificação de padrões (seção 5.3.2), geração (seção 5.3.3), filtragem (seção 5.3.4) e ordenação (seção 5.3.5) das regras de tradução.

Por fim, a seção 5.4 descreve o processo de tradução automática realizado com base nas regras e no léxico bilíngüe induzidos no projeto ReTraTos.

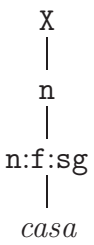
5.1 Formalismos de representação adotados no ReTraTos

Essa seção descreve os formalismos adotados para a representação dos exemplos de tradução (seção 5.1.1), do léxico bilíngüe (seção 5.1.2) e das regras de tradução (seção 5.1.3). Esses formalismos foram definidos com base em um levantamento dos modelos existentes na literatura e nas necessidades e características do projeto ReTraTos.

5.1.1 Formalismo de representação dos exemplos de tradução

O formalismo de representação dos exemplos de tradução foi definido com base nas idéias de (PROBST et al., 2003) e (TENNI et al., 1999). Em (PROBST et al., 2003), os autores apontam que as regras de tradução, por representarem generalizações em nível de PoS ou constituintes, estão sujeitas ao problema de que palavras com a mesma PoS podem se comportar de maneira distinta; e citam como exemplo o caso dos adjetivos em francês que podem aparecer antes ou depois do substantivo que modificam. Os autores enfatizam que, nesses casos, o algoritmo de indução deveria explorar níveis diferentes de informação no espaço de exemplos.

Seguindo a mesma linha de raciocínio, no algoritmo de generalização de exemplos apresentado em (TENNI et al., 1999), cada forma superficial presente em um exemplo é vista como uma árvore em que a palavra é o nó-folha e o nó-raiz é uma variável sem restrições que, conseqüentemente, pode ser instanciada com qualquer palavra. Os nós que estão entre a folha e a raiz possuem conjuntos de características que aumentam a generalização conforme se caminha em direção à raiz. Por exemplo, na árvore representada a seguir, o nó-folha representa o substantivo (`n`) feminino (`f`) singular (`sg`) *casa* do `pt` e a generalização aumenta conforme se caminha do nó-folha para o nó-raiz (representado por uma variável `X`).



Com base nessas idéias, optou-se por representar os exemplos de tradução armazenando seus diversos níveis de informação em campos separados de uma estrutura de dados do tipo vetor associativo (*hash*) permitindo, desse modo, que cada nível de informação seja acessado de maneira rápida e direta, de acordo com a necessidade.

Assim, considerando-se que cada exemplo de tradução $E_i : {E_i}^S \leftrightarrow {E_i}^T$ é composto por uma parte (sentença) fonte ${E_i}^S$ e uma parte (sentença) alvo ${E_i}^T$ e que ${E_i}^S$ representa uma seqüência de itens fontes e ${E_i}^T$, uma seqüência de itens alvo – como apresentado em (1); as informações presentes em cada parte do exemplo de tradução são armazenadas como mostrado em (2).

(1) ${E_i}^S$: $\texttt{item\_fonte}_1$ $\texttt{item\_fonte}_2$ $\texttt{item\_fonte}_3$ ... $\texttt{item\_fonte}_n$

$E_i{}^T$: `item_alvo`$_1$ `item_alvo`$_2$ `item_alvo`$_3$ ... `item_alvo`$_m$

Cada `item` pode possuir até 5 tipos de informação:

`superficial`: forma superficial de uma palavra ou um caractere de pontuação, um número etc., da maneira como é encontrado no exemplo. Por exemplo: casas, amando.

`base`: lema de uma palavra ou um caractere de pontuação, um número etc., quando etiquetados pelo etiquetador. Por exemplo: casa, amar.

`pos`: PoS do item lexical conforme atribuído pelo etiquetador morfossintático. As palavras desconhecidas e a maioria dos caracteres de pontuação não possuem esta informação. Por exemplo: `n` (substantivo), `vblex` (verbo).

`atributo`: cada um dos atributos morfossintáticos de uma PoS. Cada atributo é delimitado pelos caracteres "<" e ">" e a quantidade de atributos varia de acordo com a PoS. Por exemplo: <f><pl> (feminino, plural), <ger> (gerúndio).

`alinhamento`: seqüência de um ou mais números (separados pelo caractere "_"), referentes às posições dos itens, na sentença paralela, com os quais o item em questão se alinha. Por exemplo: 25, 3_4.

Essas informações podem ser encontradas nos exemplos de tradução em uma das três formas apresentadas a seguir (por meio de expressões regulares):[1]

1. *`superficial`/`superficial`:`alinhamento`
   Palavras desconhecidas. Exemplo: *piquiá/piquiá:4
2. `superficial`:`alinhamento`
   Caracteres de pontuação não etiquetados pelo etiquetador. Exemplo: ":27
3. `superficial`/`C`[\+`C`]*:`alinhamento`
   Demais palavras e caracteres de pontuação etiquetados pelo etiquetador; em que
   `C` = `base`<`pos`>`A`* e
   `A` = [`atributo`]+
   Exemplos: ao/a<pr>+o<det><def><f><pl>:18, o/o<det><def><m><sg>:3, ,/,<cm>:12

[1]No formalismo de uma expressão regular utilizado neste documento, o par de caracteres '[' e ']' delimita um conjunto de elementos, o caractere '*' indica zero ou mais ocorrências enquanto o caractere '+' indica uma ou mais ocorrências. Para indicar a ocorrência de um desses caracteres sem as funções especiais descritas anteriormente, deve-se usar o caractere de escape '\'.

Como apresentado no Capítulo 4, as informações `base`, `pos` e `atributo`(s) são obtidas como resultado da etiquetação morfossintática, na qual também é preservada a forma superficial (`superficial`) de entrada. O alinhamento, por sua vez, é definido por um alinhador lexical. Um exemplo de tradução com as informações citadas anteriormente é apresentado na Tabela 18 do Capítulo 4.

(2) Para cada parte fonte (alvo) de um exemplo de tradução E_i, constrói-se um vetor associativo (`%exemplo`[2]) no qual são armazenados, separadamente, cada um dos 5 tipos de informação apresentados em (1). Cada tipo de informação é um campo distinto de `%exemplo` cujas chaves são: `sup` (`superficial`), `lex` (`base`), `pos` (`pos`), `atr` (`atributo`) e `ali` (`alinhamento`). Cada um desses campos tem como valor um vetor (não associativo) de n elementos em que n é o número de itens fonte (alvo) presentes nesse exemplo:

`%exemplo`

{`sup`} = um vetor no qual cada elemento contém uma forma `superficial`

{`lex`} = um vetor no qual cada elemento contém uma forma `base` ou mais de uma separadas por "+"

{`pos`} = um vetor no qual cada elemento contém `NC` (quando nenhuma `pos` foi atribuída ao item em questão) ou uma ou mais `pos` separadas por "+"

{`atr`} = um vetor no qual cada elemento contém `NC` (quando nenhum `atributo` foi atribuído à PoS) ou os `atributo`(s) da PoS a qual esse elemento pertence ou uma combinação de mais de um desses valores separados por "+"

{`ali`} = um vetor no qual cada elemento contém um `alinhamento`

Após criar um vetor associativo `%exemplo` para a parte fonte (${E_i}^S$) e outro para a parte alvo (${E_i}^T$) de um exemplo de tradução (E_i), estes são inseridos em vetores que contêm, respectivamente, todos os exemplos fonte (`@exemplosfonte`) e todos os exemplos alvo (`@exemplosalvo`) do *corpus* paralelo.

Por exemplo, as partes fonte e alvo do exemplo de tradução apresentado na Tabela 18 do Capítulo 4, exemplo 87, são armazenadas, respectivamente, em `$exemplosfonte[87]` e `$exemplosalvo[87]` como apresentado na Figura 14.

[2]Neste documento, as estruturas de dados são representadas por meio da sintaxe da linguagem `Perl` (na qual ReTraTos foi implementado). Assim, um vetor associativo é indicado como `%ass` e o valor do elemento desse vetor com chave `chave1` é obtido acessando-se `$ass{chave1}`; enquanto um vetor (não associativo) é representado como `@vet` e seu primeiro elemento é dado por `$vet[0]`.

```
$exemplosfonte[87]
 {sup}=  Embora | o                | piquiá | não | esteja          | sob | risco | ...
 {lex}=  Embora | o                | piquiá | não | estar           | sob | risco | ...
 {pos}=  cnjadv | det              | NC     | adv | vblex           | pr  | n     | ...
 {atr}=  NC     | <def><m><sg>     | NC     | NC  | <prs><p3><sg>   | NC  | <m><sg> | ...
 {ali}=  1      | 3                | 4      | 5   | 6_7             | 8   | 9     | ...

$exemplosalvo[87]
 {sup}=  Pese_a | que  | el               | piquiá | no  | se                          | ...
 {lex}=  Pese_a | que  | el               | piquiá | no  | se                          | ...
 {pos}=  pr     | cnjsub | det            | NC     | adv | prn                         | ...
 {atr}=  NC     | NC   | <def><m><sg>     | NC     | NC  | <pro><ref><p3><mf><sp>      | ...
 {ali}=  1      | 0    | 2                | 3      | 4   | 5                           | ...
```

Figura 14: Exemplo de tradução armazenado para ser manipulado no ReTraTos

5.1.2 Formalismo de representação do léxico bilíngüe

O formalismo de representação do léxico bilíngüe adotado no projeto ReTraTos é praticamente o mesmo usado no sistema de TA `Apertium`, cuja documentação (FORCADA et al., 2005) pode ser obtida em `http://apertium.sourceforge.net/`. Nessa documentação, define-se um único formato XML[3] – por meio da mesma DTD (*Document Type Definition* ou definição de tipo de documento) – para os três tipos de arquivos com informações lingüísticas usados no `Apertium`: dicionário morfológico (monolíngüe), léxico bilíngüe e dicionário de pós-geração.

Os dicionários morfológicos (monolíngües) são criados para cada um dos idiomas implicados na tradução e são usados para construir analisadores morfológicos e geradores, de acordo com o sentido no qual são lidos. Se o dicionário for lido da esquerda para a direita o resultado será um analisador e se for lido da direita para a esquerda, um gerador. Nesses dicionários, estão definidos os paradigmas de flexão e as palavras com seus respectivos conjuntos de características (PoS, atributos etc.) entre outras informações.

Os léxicos bilíngües, por sua vez, são criados para os pares de idiomas envolvidos na tradução e representam o processo de transferência lexical, ou seja, a atribuição da forma lexical alvo correspondente a uma dada forma lexical fonte. Esses léxicos, quando lidos em sentidos diferentes, produzem módulos de transferência lexical distintos: fonte-alvo (leitura realizada da esquerda para a direita) e alvo-fonte (leitura realizada da direita para a esquerda). As entradas desses léxicos, quando possível, são bastante genéricas especificando apenas as informações indispensáveis para diferenciar a parte fonte da parte alvo.

[3] `http://www.w3.org/XML/`

Por fim, os dicionários de pós-geração são criados para cada um dos idiomas e especificam as transformações ortográficas que algumas palavras devem sofrer quando combinadas com outras. Um exemplo de uma transformação ortográfica é a que acontece, por exemplo, em `pt`, quando a preposição *a* precede o artigo definido masculino *o* exigindo-se que uma contração seja realizada resultando na palavra *ao*.

Dois desses três tipos de arquivos com informações lingüísticas também são utilizados no ReTraTos: os dicionários morfológicos e os léxicos bilíngües. Os dicionários morfológicos são utilizados no momento da etiquetação morfossintática e, no ReTraTos, foram aumentados de maneira automática a partir da conversão do conteúdo dos dicionários eletrônicos de `Unitex` (`pt` e `en`) para o formato de `Apertium`, ou por meio da inserção de entradas provenientes de outro dicionário já no formato de `Apertium` (`es`) como apresentado no Capítulo 4, Seção 4.2.

Os léxicos bilíngües, por sua vez, são induzidos automaticamente no ReTraTos a partir dos exemplos de tradução. Um trecho do léxico bilíngüe `es`–`pt` induzido automaticamente no ReTraTos é apresentado na Figura 15. De acordo com o formalismo de representação, um léxico bilíngüe está composto, principalmente, por uma seção responsável pela definição dos símbolos gramaticais que aparecem nas entradas do léxico (<`sdefs`>) e outra na qual estão definidas as correspondências bilíngües do léxico (<`section`>). Outras duas seções também estão presentes: <`alphabet`> e <`pardefs`> usadas, respectivamente, para definir o alfabeto e os paradigmas, mas que são mantidas como elementos vazios (não englobam nenhum conteúdo) no léxico bilíngüe gerado pelo ReTraTos.

A seção <`sdefs`> agrupa todas as definições de símbolos (`sdef`) presentes em um léxico. Cada definição de símbolo é um elemento vazio cujo propósito é especificar os nomes dos símbolos gramaticais (como o valor de seu atributo `n`) usados na etiquetação morfossintática. A lista completa dos símbolos gramaticais usados no projeto ReTraTos pode ser consultada no Apêndice A.

As entradas bilíngües são definidas em uma ou mais seções (`section`) como a seção `main` apresentada na Figura 15. Cada seção é preenchida com elementos <`e`> que possuem dois atributos opcionais: `r` e `lm`. Desses, apenas o primeiro, `r`, aparece nos léxicos bilíngües. Esse atributo especifica uma restrição de sentido de leitura, ou seja, em qual sentido da tradução essa entrada deverá ser considerada: da esquerda para a direita (`r="LR"`) ou da direita para a esquerda (`r="RL"`). Se esse atributo não estiver especificado, assume-se que a entrada deve ser considerada em ambos os sentidos. Um exemplo de uso do atributo `r` pode ser encontrado na Figura 16. O segundo atributo, `lm`, é usado nos dicionários morfológicos

```
<?xml version="1.0" encoding="iso-8859-1"?>
<dictionary>
  <alphabet/>
  <sdefs>
    <sdef n="n"/>
    <sdef n="pr"/>
    <sdef n="vblex"/>
    <sdef n="ger"/>
    <sdef n="inf"/>
    ...
  </sdefs>
  <pardefs>
  </pardefs>
  <section id="main" type="standard">
    ...
    <e>
      <p>
        <l>abanico<s n="n"/></l>
        <r>leque<s n="n"/></r>
      </p>
    </e>
    ...
    <e>
      <p>
        <l>acerca<b/>de<s n="pr"/></l>
        <r>sobre<s n="pr"/></r>
      </p>
    </e>
    ...
    <e>
      <i>agrupar<s n="vblex"/><s n="ger|inf"/></i>
    </e>
    ...
  </section>
</dictionary>
```

Figura 15: Trecho do léxico bilíngüe `es`–`pt` induzido automaticamente no ReTraTos

para indicar o lema de uma palavra.

O elemento <e>, por sua vez, consiste de elementos <p> (relação em pares), <i> (relação de identidade), <par> (referência a paradigma) ou <re> (expressão regular), dos quais apenas os dois primeiros (<p> e <i>) são encontrados nos léxicos bilíngües gerados pelo ReTraTos.

O elemento básico dos léxicos, <p>, é usado em qualquer tipo de entrada para indicar

a correspondência entre duas cadeias de caracteres. As cadeias são definidas por um par de elementos internos: um esquerdo (elemento <l>, *left*) e outro direito (elemento <r>, *right*). Dentro dos elementos <l> ou <r> pode haver texto e referências a símbolos gramaticais feitas por meio de elementos <s>. Cada elemento <s> especifica uma informação morfossintática como o valor de seu atributo n (definido previamente na seção <sdefs>). Por exemplo, no trecho do léxico bilíngüe apresentado na Figura 15, um elemento <s n="n"/> é usado para designar a PoS (n) do substantivo *abanico*.

Uma diferença entre o formalismo de representação dos léxicos bilíngües de Apertium e o dos léxicos bilíngües de ReTraTos é que, neste último, é possível que mais de um símbolo gramatical apareça como valor do atributo n da etiqueta s. Por exemplo, a entrada para o verbo *agrupar* na Figura 15 é válida para o verbo no gerúndio (ger) ou no infinitivo (inf) o que é especificado na entrada como <s n="ger|inf"/>.

Geralmente indica-se apenas o primeiro símbolo de cada forma lexical (correspondente à PoS) e os que forem necessários para diferenciar a forma lexical fonte da forma lexical alvo. No momento da transferência lexical, os símbolos que não estiverem especificados explicitamente na entrada do léxico são copiados da forma lexical fonte para a forma lexical alvo.

Outro elemento que pode aparecer no interior do elemento <e> é o elemento <i>: uma maneira resumida de representar um elemento <p> quando <l> e <r> são idênticos, resultando em uma notação mais compacta e legível. Na Figura 15, é exemplificada uma entrada deste tipo para o verbo *agrupar*, uma palavra presente no vocabulário dos idiomas pt e es.

Outras etiquetas que podem estar presentes dentro dos elementos <l> e <r> são as usadas nas entradas de multipalavras: <b/> (indica um caractere de espaço em branco como o exemplo apresentado na Figura 15), <j/> (indica que as palavras devem ser tratadas como uma única unidade, por exemplo, em contrações) e <g>...</g> (indica que as formas lexicais entre as etiquetas <g> e </g> não sofrem flexão, por exemplo, em expressões com verbos).

Por fim, esse formalismo também prevê os casos nos quais, em um idioma, uma mesma palavra é válida para diferentes gêneros (masculino–feminino, mf) ou números (singular–plural, sp), mas, no outro idioma, há mais de uma opção de tradução que varia de acordo com o gênero ou o número em questão. Nesse caso, a informação gramatical de uma forma lexical em um idioma não é suficiente para determinar o gênero (masculino, m, ou feminino,

f) ou o número (singular, sg, ou plural, pl) de sua correspondente no outro idioma. Por exemplo, a palavra *bolsista* em pt é masculino–feminino (mf) enquanto que, em es, há duas possibilidades de tradução para esta palavra: *becario*, no masculino (m), e *becaria*, no feminino (f).

Esse problema é tratado como apresentado na Figura 16. Nesse caso, no sentido es–pt (r="LR"), a entrada satisfaz os dois gêneros (f|m) da palavra *becario* em es – feminino (*becaria*) e masculino (*becario*). No sentido inverso (pt–es, r="RL"), como não há informação gramatical suficiente para determinar se a tradução da palavra, em pt, *bolsista* deve estar no masculino (*becario*) ou no feminino (*becaria*), o atributo de gênero recebe o valor GD (gênero a determinar) e a definição do valor final desse atributo é adiada para módulos posteriores. Algo semelhante pode ser feito para o atributo de número, com a atribuição do valor ND (número a determinar).

```
<e r="LR">
  <p>
    <l>becario<s n="n"/><s n="f|m"/><s n="sg"/></l>
    <r>bolsista<s n="n"/><s n="mf"/><s n="sg"/></r>
  </p>
</e>
<e r="RL">
  <p>
    <l>becario<s n="n"/><s n="GD"/><s n="sg"/></l>
    <r>bolsista<s n="n"/><s n="mf"/><s n="sg"/></r>
  </p>
</e>
```

Figura 16: Exemplo de entradas no léxico bilíngüe es–pt induzido por ReTraTos para o tratamento de diferenças gramaticas de acordo com o sentido da tradução

5.1.3 Formalismo de representação das regras de tradução

Como apresentado na seção 2.1, há diversas maneiras de se representar as regras de tradução. No projeto ReTraTos, as regras contêm, basicamente, as mesmas informações especificadas em (LAVIE et al., 2004) (veja Figura 2) além de outras duas: freqüência e peso. Assim, uma regra de tradução induzida automaticamente no ReTraTos, como a apresentada na Figura 17, contém as seguintes informações:

- **Identificador único**: identifica unicamente uma regra e é composto pela letra "R" seguida de um número inteiro maior que 0 (R30 no exemplo da Figura 17);

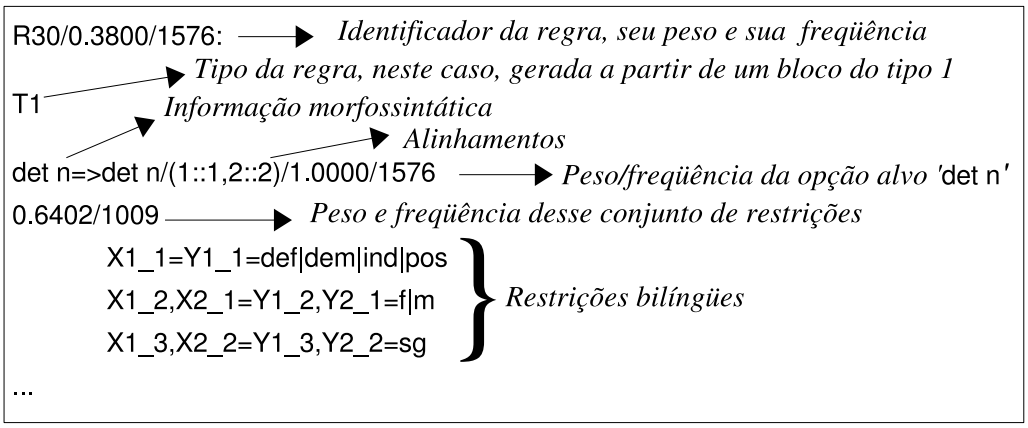


Figura 17: Exemplo de uma regra de tradução no formalismo utilizado no ReTraTos

- **Peso** e **freqüência** da **regra**: a freqüência é a quantidade de vezes em que a parte fonte da regra foi encontrada no conjunto de exemplos de tradução, e o peso é esta freqüência dividida pela freqüência de todas as partes fonte das regras induzidas;

- **Informação de tipo**: indica o tipo da regra que, no ReTraTos, se refere ao tipo do bloco de alinhamento a partir do qual essa regra foi gerada (T1 no exemplo da Figura 17);

- **Informação morfossintática**: está presente na seqüência de PoS das partes fonte e alvo que formam a regra (`det n` no exemplo da Figura 17), bem como nos possíveis valores para seus atributos, quando especificados (por exemplo, os possíveis valores de gênero para `det` e `n` no exemplo da Figura 17: `f|m`);

- **Alinhamentos**: indicações do alinhamento entre os itens fonte e alvo da regra no formato i::j em que i é a posição de um item fonte e j a posição de um ou mais itens alvo (separadas pelo caractere "_") com os quais o item fonte, na posição i, se alinha. Por exemplo, o primeiro item fonte no exemplo da Figura 17 está alinhado com o primeiro item alvo (1::1) e o segundo item fonte, com o segundo item alvo (2::2);

- **Peso** e **freqüência** de uma determinada **opção alvo**: a freqüência é o número de exemplos de tradução nos quais a parte fonte e a parte alvo da regra aparecem alinhadas; e o peso é dado pela freqüência dividida pelo número total de exemplos nos quais a parte fonte aparece (a freqüência da regra). Em gramáticas não-ambígüas há apenas uma opção alvo cujo peso é igual a 1;

- **Peso** e **freqüência** do **conjunto de restrições**: valores semelhantes aos do item anterior, mas, desta vez, calculados para cada um dos possíveis conjuntos de restrições de atributos, quando especificados;

- **Restrições do lado fonte**: informações a respeito dos valores dos atributos fonte. Uma restrição fonte é indicada por meio de variáveis `Xi_k` que informam as posições dos itens fonte (`i`) e de seus atributos (`k`) aos quais a restrição se aplica;

- **Restrições do lado alvo**: informações a respeito dos valores dos atributos alvo. Uma restrição alvo é indicada por meio de variáveis `Yj_h` que informam as posições dos itens alvo (`j`) e de seus atributos (`h`) aos quais a restrição se aplica;

- **Restrições bilíngües** (ou de ambos os lados): correspondências entre os valores dos atributos fonte e alvo, indicadas por meio de variáveis (`Xi_k` e `Yj_h`) que informam as posições dos itens e atributos fonte e alvo aos quais a restrição se aplica. Por exemplo, a terceira restrição bilíngüe da Figura 17 é uma restrição de concordância/valor e especifica que o terceiro atributo do primeiro item fonte (`X1_3`) e do primeiro item alvo (`Y1_3`) e o segundo atributo do segundo item fonte (`X2_2`) e do segundo item alvo (`Y2_2`) devem possuir valores iguais a `sg`, ou seja, o determinante e o substantivo de ambos os lados devem concordar em número e estar no singular.

5.2 Indução dos léxicos bilíngües no ReTraTos

Assim como (WU & XIA, 1994), (RESNIK & MELAMED, 1997) e outros que induzem léxicos bilíngües a partir de correspondências lexicais determinadas por um método estatístico (BROWN et al., 1993), ou com base em posições similares nos textos fonte e alvo (MELAMED, 1996a), o indutor do ReTraTos também se baseia nestas e em outras características uma vez que induz os léxicos a partir dos alinhamentos lexicais gerados por `LIHLA` (`pt–es`) e `GIZA++` (`pt–en`) (veja Capítulo 4, seção 4.3).

O processo de indução dos léxicos bilíngües, no ReTraTos, pode ser dividido em 7 passos como apresentado na Tabela 22. O primeiro passo (P1) é considerado de pré-processamento enquanto os outros 6 são responsáveis pela indução do léxico bilíngüe.

No primeiro passo, P1, os exemplos de tradução fornecidos como entrada são lidos e armazenados nas estruturas de dados apresentadas na seção 5.1.1 (`@exemplosfonte` e `@exemplosalvo`) para que possam ser manipulados adequadamente pelo restante do programa.

O passo 1 dá início ao processo de indução de um léxico bilíngüe no ReTraTos e, para tanto, o vetor de exemplos de tradução fonte (`@exemplosfonte`) é percorrido em busca das possíveis traduções para cada palavra, em cada exemplo. Mais especificamente, para

Tabela 22: Passos do processo de indução de léxicos bilíngües no ReTraTos

P1. Leitura dos exemplos de tradução

1. Criação de um léxico bilíngüe para o sentido fonte–alvo

2. Criação de um léxico bilíngüe para o sentido alvo–fonte

3. União dos léxicos criados nos passos anteriores

4. Generalização das entradas do léxico bilíngüe

5. (opcional) Tratamento de diferenças de gênero ou número

6. Tratamento de multipalavras

cada forma base de cada palavra fonte (campo 'lex' do vetor associativo que armazena as informações de um dado exemplo) sua PoS (campo 'pos') e atributos (campo 'atr'), buscam-se as possíveis traduções na sentença alvo (com base no campo 'ali'). Quando o alinhamento envolve mais de uma palavra em um ou ambos os lados, todas as informações referentes a essas palavras são concatenadas (separadas pelo caractere "+") e consideradas como uma unidade multipalavra.

Ao final deste passo, todas as possíveis traduções para cada palavra (ou multipalavra) fonte são armazenadas juntamente com suas respectivas freqüências de ocorrência (quantas vezes ela aparece alinhada com a correspondente alvo) em um vetor associativo (`%lexfonte`). Algo semelhante é realizado no passo 2, no sentido inverso (com o resultado armazenado em `%lexalvo`).

Por exemplo, considere as opções de tradução encontradas durante a geração dos léxicos bilíngües `es`–`pt` (`%lexfonte`) e `pt`–`es` (`%lexalvo`) para as formas base *el* (cuja PoS é `det`) e *o* (`det`), respectivamente, apresentadas na Figura 18. Cada combinação de possíveis valores de atributos (<def><f><pl>, <def><f><sg>, <def><m><pl> e <def><m><sg>) possui várias opções de tradução acompanhadas de suas freqüências e dos possíveis valores de atributos. Nesse exemplo, a melhor tradução para o determinante em `es` *el* é o determinante em `pt` *o* e vice-versa.

No passo 3, os léxicos criados para cada sentido da tradução (`%lexfonte` e `%lexalvo`) são unidos. Para tanto, as ambigüidades são solucionadas selecionando-se a opção de maior freqüência e especificando-se o sentido de tradução válido para a entrada. Uma entrada é válida para ambos os sentidos de tradução quando a correspondência que representa é a mais

```
$lexfonte{el/det}
   {<def><f><pl>}
     {o/det}
       [0] = 2334
       [1] =
         {<def><f><pl>} = 2095
         {<def><m><pl>} = 145
         ...
     {de+o/pr+det}
       [0] = 97
       [1] =
         {NC+<def><f><pl>} = 77
         {NC+<def><m><pl>} = 9
         ...
     ...
   {<def><f><sg>}
     {o/det}
       [0] = 8854
       [1] =
         {<def><f><sg>} = 8476
         {<def><m><sg>} = 330
         ...
     ...
   {<def><m><pl>}
     {o/det}
       [0] = 3706
       [1] =
         {<def><m><pl>} = 3395
         {<def><f><pl>} = 181
         ...
     ...
   {<def><m><sg>}
     {o/det}
       [0] = 8902
       [1] =
         {<def><m><sg>} = 8147
         {<def><f><sg>} = 728
     ...
```

```
$lexalvo{o/det}
   {<def><f><pl>}
     {el/det}
       [0] = 2322
       [1] =
         {<def><f><pl>} = 2095
         {<def><m><pl>} = 181
         ...
     {suyo/det}
       [0] = 13
       [1] =
         {<pos><mf><pl>} = 13


     ...
   {<def><f><sg>}
     {el/det}
       [0] = 9331
       [1] =
         {<def><f><sg>} = 8476
         {<def><m><sg>} = 728
         ...
     ...
   {<def><m><pl>}
     {el/det}
       [0] = 3569
       [1] =
         {<def><m><pl>} = 3395
         {<def><f><pl>} = 145
         ...
     ...
   {<def><m><sg>}
     {el/det}
       [0] = 8574
       [1] =
         {<def><m><sg>} = 8147
         {<def><f><sg>} = 330
     ...
```

Figura 18: Possíveis traduções para os determinantes *el* em `es` e *o* em `pt` e todas suas combinações de atributos

freqüente nos 2 sentidos: fonte–alvo e alvo–fonte. Se a correspondência for a mais freqüente em apenas um dos sentidos, esse deve ser especificado (veja seção 5.1.2).

O vetor associativo gerado ao final do passo 3, `%lexbil`, diferentemente dos vetores usados para armazenar os léxicos criados para cada sentido da tradução, ordena as opções por sentido, ou seja, para cada base/pos fonte, em um dado sentido da tradução (`LR`, `RL` ou

ambos, ''), são armazenadas as possíveis combinações de atributos fonte, forma base alvo, PoS alvo, atributos alvo e freqüência de ocorrência, nesta ordem, separados pelo caractere "/". Por exemplo, após a realização do passo 3, o exemplo apresentado anteriormente é representado como:

```
$lexbil{el/det}
  {''} =
    [0] = <def><f><pl>/o/det/<def><f><pl>/2095
    [1] = <def><f><sg>/o/det/<def><f><sg>/8476
    [2] = <def><m><pl>/o/det/<def><m><pl>/3395
    [3] = <def><m><sg>/o/det/<def><m><sg>/8147
```

Outra tarefa realizada no passo 3 é o controle na criação de entradas envolvendo multipalavras: quando uma correspondência envolve mais de uma palavra em um ou ambos os lados ela só é inserida no léxico bilíngüe se ocorrer um número mínimo de vezes (parâmetro fornecido pelo usuário). Essa estratégia é adotada para tentar minimizar o impacto da alta taxa de erro no alinhamento automático de multipalavras (veja Tabelas 17 e 20, no Capítulo 4).

No passo 4, tenta-se generalizar as várias combinações de atributos fonte e alvo nas entradas do léxico bilíngüe percorrendo-se o vetor com essas combinações em busca de pares de entradas que sejam diferentes em apenas um valor de atributo. Além disso, o valor de atributo diferente na combinação de atributos fonte deve ser o mesmo na combinação de atributos alvo. Satisfeitas estas condições, as entradas são substituídas por uma nova entrada na qual o atributo com valores divergentes é substituído pela concatenação desses valores (separados pelo caracteres "|") e a freqüência de ocorrência da nova entrada é a soma das freqüências das entradas que foram substituídas.

Por exemplo, as combinações de atributos encontradas no léxico bilíngüe para a entrada `el/det` apresentadas anteriormente são generalizadas no passo 4 resultando na entrada[4]:

```
$lexbil{el/det}
  {''} =
    [0] = <def><f|m><pl|sg>/o/det/<def><f|m><pl|sg>/22113
```

O passo 5, por sua vez, é responsável pelo tratamento de entradas nas quais o valor

[4]Note que, na generalização dos atributos, por questão de padronização, os valores dos atributos são ordenados em ordem alfabética crescente.

do atributo de gênero ou de número não pode ser determinado com base nas informações contidas na entrada e, por isso, deve ser determinado pelo módulo de transferência estrutural. Como apresentado na seção 5.1.2, é possível que uma mesma forma base seja válida para dois gêneros (feminino e masculino) ou números (singular e plural) em um idioma, mas que no outro existam duas traduções diferentes para cada um dos gêneros ou números.

Esse passo, que é opcional, só será realizado se um arquivo com informações a respeito dos possíveis valores para os atributos de gênero e número for fornecido pelo usuário. Nesse arquivo devem estar especificados todos os possíveis valores para gênero (em uma linha) e número (em outra linha), separados por um espaço em branco e na seguinte ordem: valores individuais em ordem alfabética crescente (`f` e `m` ou `pl` e `sg`), valor geral (`mf` ou `sp`) e valor a determinar (`GD` ou `ND`) como apresentado a seguir:

```
f m mf GD
pl sg sp ND
```

A partir dessas especificações, o léxico bilíngüe é percorrido em busca de entradas que contenham os valores individuais enumerados em ordem alfabética crescente (`f|m` ou `pl|sg`) em um lado e os valores gerais (`mf` ou `sp`) em outro. Para essas entradas são criadas duas novas entradas com indicações do sentido da tradução: uma com os atributos inalterados para o lado com o valor geral e outra na qual os valores enumerados são substituídos pelo valor a determinar (`GD` ou `ND`). Por exemplo, considere a entrada apresentada a seguir para a palavra em `es` *tesis*:

```
$lexbil{tesis/n}
  {''} = <f><sp>/tese/n/<f><pl|sg>
```

A melhor tradução encontrada, em `pt`, para a palavra *tesis* (`sp`) foi a palavra *tese* que varia em número (`pl|sg`). Assim, após o passo 5, a entrada apresentada acima, válida para ambos os sentidos, é dividida em duas, cada uma com a especificação do sentido de tradução válido:

```
$lexbil{tesis/n}
  {LR} = <f><sp>/tese/n/<f><ND>
  {RL} = <f><sp>/tese/n/<f><pl|sg>
```

No último passo, passo 6, realizado já no momento da impressão das entradas do léxico bilíngüe no arquivo de saída, as multipalavras são formatadas de acordo com o formalismo de `Apertium` inserindo elementos <b/>, <j/> ou <g>...</g>. O elemento <b/> substitui o caractere "_" inserido pelo etiquetador para delimitar as palavras que formam a unidade

multipalavra. O elemento <j/> por sua vez, substitui o caractere "+" inserido para separar as palavras que formavam um alinhamento multipalavras no momento da criação dos léxicos (passos 1 e 2) ou pelo etiquetador para indicar, por exemplo, uma contração. Por fim, o elemento <g>...</g> delimita o grupo de palavras que segue um verbo em uma expressão multipalavra.

Também no momento da impressão, valores de atributos fonte e alvo iguais não são impressos já que o formalismo especifica que, nestes casos, não é necessário especificá-los explicitamente (veja seção 5.1.2).

O léxico bilíngüe induzido no ReTraTos pode ser usado no momento da TA para auxiliar o processo de transferência de uma representação da sentença fonte para uma representação da sentença alvo (veja seção 5.4). Contudo, é importante citar que as estatégias adotadas para a indução automática do léxico bilíngüe, principalmente as que dizem repeito ao tratamento automático de multipalavras, estão sujeitas a erros, como mostra a avaliação do léxico induzido automaticamente apresentada na seção 6.1 do Capítulo 6.

5.3 Indução das regras de tradução no ReTraTos

O processo de indução de regras de tradução desenvolvido no ReTraTos segue as quatro etapas citadas no Capítulo 2 – (1) identificação de padrões, (2) geração, (3) filtragem e (4) ordenação das regras de tradução – porém se difere dos demais métodos da literatura no modo como as regras são buscadas e filtradas.

O método de indução de regras de tradução proposto no ReTraTos parte do pressuposto de que as regras de tradução devem ser induzidas separadamente para cada tipo de alinhamento (0 - omissão, 1 - em ordem e 2 - com mudança de ordem). Assim, diferentemente dos demais métodos estudados, no ReTraTos os alinhamentos lexicais não são usados apenas para definir as correspondências lexicais entre parte fonte e parte alvo de um exemplo de tradução ou definir o escopo dessa busca mas, principalmente, para guiar a busca por regras de tradução.

Essa abordagem foi adotada porque a freqüência de ocorrência de alinhamentos dos tipos 0 e 2 são bem menores do que a freqüência de ocorrência de alinhamentos do tipo 1 e, sendo assim, se o mesmo limite mínimo de freqüência fosse utilizado para a identificação de padrões independentemente do tipo, os padrões dos dois primeiros tipos raramente seriam encontrados. Até onde se sabe, esta é a primeira vez em que um tratamento diferenciado

para cada tipo de alinhamento é considerado na indução de regras de tradução.

Quanto à filtragem, no ReTraTos são implementadas duas estratégias de resolução de ambigüidades baseadas na busca por valores únicos em níveis de abstração diferentes do nível de PoS: valores lexicais ou de atributos morfológicos. Embora a idéia de utilizar níveis diferentes de informação na filtragem das regras já tenha sido levantada por alguns autores, não se tem conhecimento de nenhuma implementação de estratégias semelhantes às apresentadas neste documento.

Assim, o processo de indução das regras de tradução, no ReTraTos, pode ser dividido em 6 passos como apresentado na Tabela 23. Os 2 primeiros passos (P1 e P2) são considerados de pré-processamento enquanto os outros 4 são responsáveis pela indução das regras. As inovações deste método estão concentradas, principalmente, nos passos P2 e 1 – relacionados à criação e ao processamento de blocos de alinhamentos – e 3 – no qual as regras são filtradas usando as estratégias propostas. Detalhes da implementação do método de indução de regras de tradução do ReTraTos podem ser obtidos em (CASELI & NUNES, no prelo).

Tabela 23: Passos do processo de indução de regras de tradução no ReTraTos

P1. Leitura dos exemplos de tradução
P2. Criação dos blocos de alinhamentos – seção 5.3.1
1. Identificação dos padrões – seção 5.3.2
2. Geração das regras – seção 5.3.3
3. (opcional) Filtragem das regras – seção 5.3.4
4. (opcional) Ordenação das regras – seção 5.3.5

No primeiro passo de pré-processamento (P1), os exemplos de tradução fornecidos como entrada são lidos e armazenados nas estruturas de dados apresentadas na seção 5.1.1 para que possam ser manipulados adequadamente pelo restante do programa.

5.3.1 Criação dos blocos de alinhamentos

O segundo passo de pré-processamento (P2) é responsável por criar os blocos de alinhamentos a partir dos quais as regras serão induzidas. Como já mencionado, o processo de indução de regras de tradução no projeto ReTraTos tem como base os alinhamentos lexicais gerados

automaticamente (veja seção 4.3) a partir dos quais é possível identificar 3 tipos de blocos de alinhamentos (seqüências de itens alinhados) como apresentado na Figura 19:

- **Tipo 0** – contém apenas alinhamentos de omissão, ou seja, itens fonte (ou alvo) que não possuem correspondência na sentença paralela;
- **Tipo 1** – contém apenas itens fonte e alvo que estão alinhados de uma maneira que respeita a ordem em que aparecem, ou seja, o primeiro item fonte do bloco está alinhado com o primeiro item alvo, o segundo item fonte com o segundo item alvo e assim por diante. Para que um bloco do tipo 1 seja criado, é necessário que haja pelo menos 2 itens fonte ou alvo que satisfaçam as condições de alinhamento exigidas por esse tipo de bloco;
- **Tipo 2** – contém itens que estão alinhados com mudança de ordem (alinhamentos cruzados). Esse bloco engloba todos os itens fonte e alvo afetados pelo alinhamento com mudança de ordem mesmo que alguns desses itens possuam alinhamentos distintos do tipo 2. Assim, um bloco do tipo 2 pode englobar blocos de omissão (tipo 0) e que preservam a ordem de ocorrência (tipo 1).

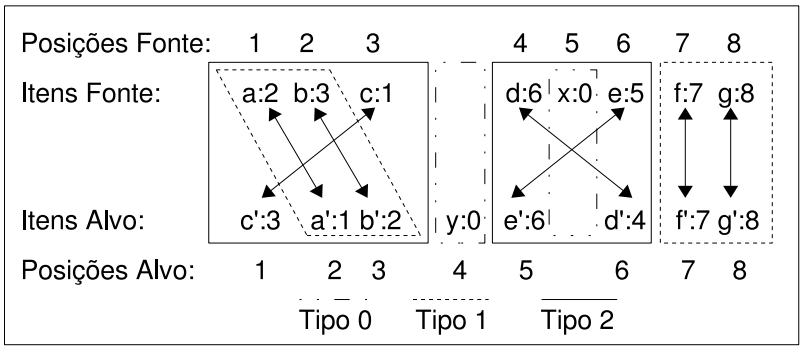


Figura 19: Exemplo dos 3 tipos de blocos de alinhamentos para um exemplo de tradução fictício

A Figura 19 ilustra cada um desses tipos de blocos com um exemplo de tradução fictício no qual existem 6 blocos de alinhamento, nessa ordem: um bloco do tipo 2 no qual está contido um bloco do tipo 1, um bloco do tipo 0, outro bloco do tipo 2 com um bloco do tipo 0 em seu interior e, por fim, um bloco do tipo 1. Note que os blocos de tipo 2 englobam todos os itens afetados pelo alinhamento com mudança de ordem, mesmo que alguns desses itens possuam alinhamentos com tipos distintos de 2. Por exemplo, no primeiro bloco de tipo 2 os itens fonte a e b são alinhados com os itens alvo correspondentes respeitando a

ordem em que aparecem (tipo 1); e no segundo bloco do tipo 2 o item fonte x não possui alinhamento (tipo 0).

Esses blocos de alinhamentos são criados com o intuito de delimitar o escopo da busca por padrões e, conseqüentemente, tentar restringir os problemas tratados pelas regras de tradução. Em (PROBST, 2005), a autora aponta que a divisão dos exemplos de tradução em diferentes níveis é de extrema importância no aprendizado de regras a partir de *corpus* não-controlado (não elicitado) e cita que tal divisão pode ser feita, entre outros, com base na árvore de análise sintática ou nos alinhamentos lexicais relevantes.

Os blocos de alinhamentos são diferentes dos chamados *alignment templates* (OCH & NEY, 2004) usados pelos métodos mais recentes de tradução automática estatística, uma vez que estes últimos são formados seguindo critérios que não incluem o tipo de alinhamento que representam – a melhor segmentação dos exemplos de tradução em *alignment templates* é definida por modelos estatísticos. Além disso, no cálculo das probabilidades de ocorrência desses *templates*, nos métodos estatísticos, não é levado em consideração o tipo de alinhamento que representam.

Outro trabalho que também usa os alinhamentos lexicais com o intuito de segmentar os exemplos de tradução é o de (MARIÑO et al., 2005). Nesse trabalho, tais segmentos são denominados *tuplas* e se diferenciam dos blocos de alinhamentos apresentados aqui em dois pontos. Primeiro, as tuplas englobam o número mínimo de itens envolvidos no alinhamento, enquanto que os blocos de alinhamentos usados no ReTraTos englobam o máximo de itens que satisfazem as condições especificadas para a formação de tal bloco. Segundo, os alinhamentos de omissão do tipo $1:0$ formam uma tupla, mas os do tipo $0:1$ são adicionados à tupla seguinte enquanto que, no ReTraTos, ambos blocos de omissão (tipo 0) formam blocos próprios e também podem estar inseridos em um bloco do tipo 2. Embora as tuplas, assim como os blocos de alinhamentos, sejam criadas levando-se em consideração o tipo de alinhamento lexical, do mesmo modo que os *alignment templates* nenhuma informação a respeito do tipo de alinhamento é usada para induzir conhecimento por meio dos modelos estatísticos.

Assim, embora a utilização de alinhamentos lexicais no aprendizado automático de conhecimento de tradução seja uma prática comum, este é o primeiro trabalho no qual a indução é guiada pelos tipos de alinhamento. Sendo assim, após ler os exemplos de tradução e armazená-los nas estruturas de dados apresentadas na seção 5.1.1 (P1), os blocos de alinhamentos fonte e alvo são criados seguindo o algoritmo apresentado na Figura 20. Para cada exemplo de tradução, percorrem-se os alinhamentos fonte e alvo (laço L1) em busca

de blocos dos tipos 0, 1 e 2. Os blocos do tipo 1 necessariamente englobam itens que estão alinhados respeitando a ordem em que ocorrem (linha 4); os blocos do tipo 0, por sua vez, englobam itens que não estão alinhados (linhas 12 e 19 para blocos fonte e alvo, respectivamente); por fim, os blocos de alinhamentos do tipo 2 são os que englobam alinhamentos com mudança de ordem e, portanto, não satisfazem nenhuma das condições acima (linha 25).

Cada tipo de bloco de alinhamento possui uma sub-rotina específica para tratá-lo: `processa_tipo_0` (linhas 14 e 21), `processa_tipo_1` (linha 7) e `processa_tipo_2` (linha 26). Cada uma dessas sub-rotinas tem o objetivo de encontrar a posição final do bloco (fim_s, fim_t ou ambos) verificando se a condição de parada é satisfeita. Para blocos do tipo 0 a condição de parada é um item cujo alinhamento é diferente de 0; para blocos do tipo 1 a condição de parada é um item cujo alinhamento não é a posição do alinhamento anterior incrementada de 1; e para os blocos do tipo 2 essa condição será o último item envolvido no alinhamento com mudança de ordem.

As duas primeiras sub-rotinas (`processa_tipo_0` e `processa_tipo_1`) são implementadas por meio de um laço, já que todos os itens de blocos dos tipos 0 ou 1 devem, necessariamente, ser do tipo 0 ou 1, respectivamente. A sub-rotina responsável pelo processamento do bloco de tipo 2 (`processa_tipo_2`), por outro lado, não é implementada como um laço; na verdade, ela simplesmente armazena as posições inicias do bloco do tipo 2 e continua processando os itens – possivelmente criando blocos do tipo 0 e 1 – até que a posição final do bloco do tipo 2 seja encontrada e, assim, esse bloco possa ser criado.

O bloco do tipo 0 possui uma sub-rotina especial para criação de uma janela ao redor do(s) item(ns) com alinhamento(s) de omissão: `aplica_janela(ini,fim,n,|A|)`. Sua função é criar uma janela com n itens à esquerda e n itens à direita do bloco do tipo 0 alterando *ini* para *ini*-n ou 0 (se *ini*-n é menor do que 0) e fim para $fim + n$ ou |A| (se $fim + n$ é maior que o número de alinhamentos existentes, ou seja, |A|).

Por fim, a sub-rotina `cria_bloco(E,ini,fim,M,T)` cria um bloco do tipo T, para o exemplo E com posições inicial *ini* e final fim contanto que o tamanho do bloco seja maior ou igual a M, ou seja, fim-$ini \geq M$.

Ao terminar o laço L1 verifica-se se existem itens fonte (linha 33) ou alvo (linha 34) que ainda não foram inseridos em nenhum bloco. Isso acontece quando os últimos itens do exemplo fonte ou alvo não estão alinhados e, portanto, não são considerados por L1. Se for este o caso, um único bloco do tipo 0 (omissão) é criado para englobar os itens fonte ou alvo que restaram.

Entrada
janela: quantidade de posições antes e depois de uma omissão
tam_min: tamanho mínimo permitido para um bloco de alinhamento
indexe: índice do exemplo sendo processado
A^S, A^T: conjuntos com alinhamentos fonte e alvo, respectivamente, do exemplo indexe
Saída
B^S, B^T: conjuntos com blocos de alinhamentos fonte e alvo, inicialmente vazios

Algoritmo
1. $\text{fim}_s \leftarrow 0$
2. $\text{fim}_t \leftarrow 0$
3. **enquanto** ($\text{fim}_s \leq |A^S|$) **e** ($\text{fim}_t \leq |A^T|$) **faça** *#L1*
4. **se** $A^S[\text{fim}_s] = \text{fim}_t + 1$ **então** *#tipo 1*
5. $\text{ini}_s \leftarrow \text{fim}_s$
6. $\text{ini}_t \leftarrow \text{fim}_t$
7. processa_tipo_1($\text{fim}_s, A^S, \text{fim}_t, A^T$)
8. $B^S \leftarrow B^S \cup$ cria_bloco(indexe, $\text{ini}_s, \text{fim}_s$, tam_min, 1)
9. $B^T \leftarrow B^T \cup$ cria_bloco(indexe, $\text{ini}_t, \text{fim}_t$, tam_min, 1)
10. **fim_então**
11. **senão**
12. **se** $A^S[\text{fim}_s] = 0$ **então** *#tipo 0*
13. $\text{ini}_s \leftarrow \text{fim}_s$
14. processa_tipo_0(fim_s, A^S)
15. aplica_janela($\text{ini}_s, \text{fim}_s$, janela, $|A^S|$)
16. $B^S \leftarrow B^S \cup$ cria_bloco(indexe, $\text{ini}_s, \text{fim}_s$, tam_min, 0)
17. **fim_então**
18. **senão**
19. **se** $A^T[\text{fim}_t] = 0$ **então** *#tipo 0*
20. $\text{ini}_t \leftarrow \text{fim}_t$
21. processa_tipo_0(fim_t, A^T)
22. aplica_janela($\text{ini}_t, \text{fim}_t$, janela, $|A^T|$)
23. $B^T \leftarrow B^T \cup$ cria_bloco(indexe, $\text{ini}_t, \text{fim}_t$, tam_min, 0)
24. **fim_então**
25. **senão** *#tipo 2*
26. processa_tipo_2($\text{ini}_s, \text{fim}_s, \text{ini}_t, \text{fim}_t, A^S, A^T$)
27. $B^S \leftarrow B^S \cup$ cria_bloco(indexe, $\text{ini}_s, \text{fim}_s$, tam_min, 2)
28. $B^T \leftarrow B^T \cup$ cria_bloco(indexe, $\text{ini}_t, \text{fim}_t$, tam_min, 2)
29. **fim_senão**
30. **fim_senão**
31. **fim_senão**
32. **fim_enquanto** *#L1*
33. **se** $\text{fim}_s \leq |A^S|$ **então** $B^S \leftarrow B^S \cup$ cria_bloco(indexe, $\text{fim}_s, |A^S|$, tam_min, 0)
34. **se** $\text{fim}_t \leq |A^T|$ **então** $B^T \leftarrow B^T \cup$ cria_bloco(indexe, $\text{fim}_t, |A^T|$, tam_min, 0)

Figura 20: Algoritmo para criação dos blocos de alinhamentos de um dado exemplo de tradução

Os blocos fonte e alvo criados como descrito anteriormente são armazenados nos vetores associativos `%blocosfonte` e `%blocosalvo`, respectivamente. Cada vetor associativo possui como chave o tipo do bloco de alinhamento e como valor um vetor com as informações

necessárias para acessar esse bloco: o identificador do exemplo e as posições inicial e final do bloco.

Por exemplo, considere o exemplo de tradução (cujo identificador é 0) apresentado na Figura 19. Os blocos fonte e alvo desse exemplo seriam armazenados como mostrado a seguir:

```
$blocosfonte
    {0} = ((0,(5,5)))
    {1} = ((0,(1,2)),(0,(7,8)))
    {2} = ((0,(1,3)),(0,(4,6)))
$blocosalvo
    {0} = ((0,(4,4)))
    {1} = ((0,(2,3)),(0,(7,8)))
    {2} = ((0,(1,3)),(0,(5,6)))
```

A partir dos blocos de alinhamentos e de acordo com os parâmetros fornecidos pelo usuário, as regras de tradução são induzidas para um ou vários tipos (0, 1 ou 2). As próximas seções descrevem como as regras são induzidas, ou seja, as quatro etapas do processo de indução no ReTraTos: identificação de padrões (5.3.2), geração (5.3.3), filtragem (5.3.4) e ordenação (5.3.5) das regras de tradução.

5.3.2 Identificação de padrões no ReTraTos

A identificação de padrões, assim como em (MCTAIT, 2003), é realizada em dois passos: monolíngüe e bilíngüe. Porém, diferentemente desse trabalho, no ReTraTos, a identificação de padrões monolíngües é realizada apenas para a língua fonte (5.3.2.1) enquanto os padrões alvo são identificados implicitamente na fase bilíngüe (5.3.2.2).

5.3.2.1 Identificação de padrões monolíngües

Com base no trabalho de (YAMAMOTO et al., 2003), para a identificação de padrões fonte no ReTraTos optou-se pela implementação de um algoritmo inspirado na técnica de *Sequential Pattern Mining* (SPM) e no algoritmo `PrefixSpan` (PEI et al., 2001, 2004).

De acordo com (PEI et al., 2004), o problema de SPM foi introduzido pela primeira vez em (AGRAWAL & SRIKANT, 1995): "Dado um conjunto de seqüências, onde cada seqüência consiste de uma lista de elementos e cada elemento consiste de um conjunto de itens, e

dado um limite ϵ fornecido pelo usuário, SPM irá buscar todas as subseqüências freqüentes, ou seja, as subseqüências cuja freqüência de ocorrência no conjunto de seqüências não seja menor que ϵ".

O algoritmo `PrefixSpan`, apresentado em (PEI et al., 2001, 2004), implementa a técnica de SPM por meio da divisão do conjunto de seqüências por prefixo freqüente aumentando, posteriormente, essas seqüências dando prioridade à profundidade (*depth-first*).

Assim, antes de apresentar o algoritmo desenvolvido no ReTraTos para a identificação dos padrões monolíngües – definido com base na técnica de SPM e no algoritmo `PrefixSpan` – é importante apresentar alguns conceitos de SPM adaptados para o contexto do projeto ReTraTos:

seqüência: na definição original, apresentada em (PEI et al., 2004), uma seqüência q é formada por uma lista de elementos sendo cada elemento um conjunto de itens denotado por $(z_1 z_2 \dots z_n)$, em que z_k é um item. Um exemplo de uma seqüência que satisfaz a definição desses autores seria $\alpha = (a(abc)(ac)d(cf))$ a qual é composta por 5 elementos: a, (abc), (ac), d e (cf).

Porém, analisando-se as características do projeto ReTraTos e, principalmente, tendo em mente que os padrões devem ser seqüências contínuas de itens, a definição de seqüência foi alterada para definí-la diretamente como um conjunto de itens. Assim, a seqüência de PoS "`adv prn vblex cnjsub det n prn`" poderia ser denotada como $(z_1 z_2 \dots z_7)$ em que $z_1 =$ `adv`, $z_2 =$ `prn` e assim por diante até $z_7 =$ `prn`.

conjunto de seqüências: um conjunto de seqüências Q é um conjunto de registros $\langle qid, q \rangle$ em que qid é o identificador da seqüência e q, a seqüência. No ReTraTos, o conjunto inicial de seqüências é fornecido como entrada para o programa de identificação de padrões por meio de um arquivo que contém todos os blocos de alinhamento do tipo para o qual se deseja identificar os padrões. Cada linha deste arquivo corresponde a uma seqüência q, cujo qid é o número da linha que a contém. Por exemplo, um trecho de um conjunto de seqüências de PoS usado para a identificação de padrões fonte é apresentado na Tabela 24.

Tabela 24: Conjunto de seqüências Q

qid	*q*
1	(`adv prn vblex cnjsub det n prn`)
2	(`prn vblex cnjsub`)
3	(`det n prn vblex det n`)

tamanho de uma seqüência: é o número de itens em uma seqüência; uma seqüência com tamanho n é denominada seqüência-n. Por exemplo, as três seqüências apresentadas anteriormente possuem tamanhos 7, 3 e 6, respectivamente.

subseqüência e superseqüência: os conceitos originais de subseqüência e superseqüência também foram alterados como conseqüência da mudança na definição de seqüência apresentada anteriormente. Em (PEI et al., 2004), as definições de subseqüência e superseqüência permitiam a geração de seqüências de itens não consecutivos o que não condiz com o cenário de ReTraTos. Para esses autores, uma seqüência $\alpha = (a_1 a_2 \ldots a_n)$ é subseqüência de outra seqüência $\beta = (b_1 b_2 \ldots b_m)$ e β é superseqüência de α, denotado como $\alpha \sqsubseteq \beta$, se existem inteiros $1 \leq j_1 < j_2 < \ldots < j_n \leq m$ tal que $a_1 \subseteq b_{j_1}, a_2 \subseteq b_{j_2}, \ldots, a_n \subseteq b_{j_n}$. De acordo com essa definição, uma possível subseqüência de $\beta = (a(abc)(ac)d(cf))$ seria $\alpha = (a(bc)df)$.

Assim, no ReTraTos, tem-se que uma seqüência $\alpha = (a_1 a_2 ... a_n)$ é uma subseqüência de outra seqüência $\beta = (b_1 b_2 ... b_m)$ e β uma superseqüência de α se $\alpha \subseteq \beta$. Por exemplo, considerando-se $\beta =$ (`adv prn vblex cnjsub det n prn`) (a primeira seqüência apresentada na Tabela 24), tem-se que $\alpha =$ (`det n`) é uma subseqüência de β uma vez que $\alpha \subset \beta$ enquanto (`det prn`) não é uma subseqüência de β.

suporte: o suporte de uma seqüência α é dado pelo número de vezes que esta seqüência ocorre no conjunto de seqüências Q, ou seja, quantas vezes $\alpha \subseteq q$ para todos os registros $\langle qid, q \rangle$. É importante ressaltar que, no ReTraTos, α pode ocorrer mais de uma vez em q e este fato é considerado na busca por padrões. Por exemplo, dada a seqüência α = (`det n`), seu suporte no conjunto de seqüências Q da Tabela 24 é $suporte_Q(\alpha) = 3$.

padrão seqüencial: uma seqüência α é considerada um padrão seqüencial no conjunto de seqüências Q se $suporte_Q(\alpha) \geq \epsilon$. O valor ϵ é um parâmetro de entrada do programa de identificação de padrões.

prefixo: no ReTraTos[5], dada a seqüência $\alpha = (a_1 a_2 \ldots a_n)$ (em que cada a_i corresponde a um item freqüente em Q) e considerando-se a ordem na qual os itens aparecem em α, uma seqüência $\beta = (b_1 b_2 \ldots b_m)$ $(m < n)$ é um prefixo de α se e somente se $b_i = a_i$ para todo $i \in [1, m]$.

Por exemplo, $\beta =$ (`det n`), é um prefixo de $\alpha =$ (`det n prn`) já que $b_i = a_i$ para todo $i \in [1, 2]$, ou seja, $b_1 = a_1 =$ `det` e $b_2 = a_2 =$ `n`.

[5]O conceito original de prefixo (veja (PEI et al., 2004)) também foi alterado como conseqüência da mudança na definição de seqüência apresentada neste documento.

Assim, considerando-se as definições apresentadas anteriormente e as idéias do algoritmo `PrefixSpan`, foi implementado um programa para identificar os padrões no projeto ReTraTos cujo algoritmo é apresentado na Figura 21.

Além da diferença em relação à definição de seqüência – e, conseqüentemente, subseqüência, superseqüência e prefixo – a principal diferença entre `identifica_padroes.pl` e `PrefixSpan` é que, no primeiro, permite-se que um padrão ocorra mais de uma vez numa dada seqüência, enquanto que, no segundo, um padrão pode ocorrer no máximo uma vez em cada uma das seqüências de um conjunto de seqüências. Essa diferença é de fundamental importância no contexto do projeto ReTraTos uma vez que as seqüências são compostas principalmente por PoS, as quais ocorrem, com freqüência, repetidas vezes na mesma seqüência.

Além disso, no ReTraTos, não é permitida a identificação de padrões com lacunas, possíveis na definição original apresentada em (PEI et al., 2004). Essa restrição é um pressuposto deste projeto e foi inserida devido ao modo como as regras são aplicadas: as posições fonte consecutivas (sem lacunas) de uma entrada são processadas para se determinar as regras passíveis de serem aplicadas (veja seção 5.4).

A busca por padrões em `identifica_padroes.pl` (veja o algoritmo na Figura 21) é realizada por meio de três laços: (L1) que percorre todas as seqüências q do conjunto de seqüências de entrada Q e (L2) e (L3) que controlam, respectivamente, as posições inicial e final das subseqüências de q candidatas a padrões seqüenciais.

Analisando-se com mais atenção o algoritmo da Figura 21 pode-se perceber que a busca por padrões seqüenciais é otimizada em 4 momentos. O primeiro, na linha 5, limita a busca por padrões às seqüências ainda não verificadas por meio da criação de um conjunto auxiliar de identificadores de seqüências, C. A segunda otimização, na linha 8, limita a busca por padrões (α) ainda não identificados ($\alpha \notin P$) nem considerados como não freqüentes ($\alpha \notin NF$).

As duas últimas estratégias de otimização implementam a idéia principal de `PrefixSpan`: limitar a busca por padrões aos seus prefixos. Assim, na linha 12, limita-se a busca por padrões com prefixo igual ao último padrão detectado, α, àquelas seqüências nas quais α ocorre (Q_α) e, na linha 16, interrompe-se a busca por padrões com prefixo igual a α se α não foi reconhecido como um padrão seqüencial já que, se o prefixo não é freqüente, uma seqüência que o contém também não será.

Por fim, duas funções aparecem no algoritmo da Figura 21. Na linha 7, `subseq(q,ini,fim)` retorna uma seqüência α que é subseqüência de q tendo como pri-

Entrada
Q: o conjunto com n registros $\langle qid, q\rangle$
ϵ: suporte mínimo
min, max: os tamanhos mínimo e máximo, respectivamente, do padrão (*opcional*)
Saída
P: o conjunto completo de padrões seqüenciais, inicialmente vazio

Simbologia
tam(q): tamanho de uma seqüência q
C: conjunto de identificadores de seqüências (qid)
α: uma seqüência qualquer, candidata a padrão
Q_α: conjunto dos identificadores das seqüências nas quais α ocorre
NF: conjunto de seqüências consideradas não freqüentes, inicialmente vazio

Algoritmo

```
1. para cada ⟨qid, q⟩ ∈ Q faça #L1
2.     ini ← 1
3.     enquanto ini ≤ (tam(q) − min + 1) faça #L2
4.         fim ← ini + min − 1
5.         C ← {qid ... n}
6.         enquanto fim ≤ tam(q) e fim ≤ (ini + max − 1) faça #L3
7.             α ← subseq(q, ini, fim)
8.             se (α ∉ P) e (α ∉ NF) então
9.                 suporte←ocorrencias(α, C, Q_α)
10.                se suporte ≥ ε então #α é um padrão
11.                    P ← P ∪ {α}
12.                    C ← Q_α
13.                fim_então
14.                senão
15.                    NF ← NF ∪ {α}
16.                    finaliza enquanto #L3
17.                fim_senão
18.            fim_então
19.            fim ← fim + 1
20.        fim_enquanto #L3
21.        ini ← ini + 1
22.    fim_enquanto #L2
23. fim_para #L1
```

Figura 21: Algoritmo de `identifica_padroes.pl`

meiro item aquele na posição *ini* e como último, o item na posição fim. Na linha 9, `ocorrencias`(α, C, Q_α) retorna o número de vezes em que α ocorre nas seqüências q cujos identificadores qid pertencem a C e armazena em Q_α os identificadores das seqüências nas quais α ocorre.

Assim, no ReTraTos, os padrões fonte que ocorrem em um determinado tipo de bloco de alinhamento são identificados seguindo o algoritmo apresentado na Figura 21, o qual recebe como parâmetros de entrada: (1) o arquivo no qual os blocos de alinhamento do

tipo sob estudo foram impressos (Q), (2) o limite mínimo de suporte (freqüência, ϵ) e (3) os tamanhos mínimo (min, por padrão 2) e máximo (max, por padrão 5) de um padrão. Vale ressaltar que o suporte mínimo (ϵ) é definido com base na porcentagem pi (definida pelo usuário ou, por padrão, igual a 0,15%) do número total de blocos impressos: $\epsilon = pi \times n$. Essa estratégia de calcular ϵ separadamente para cada tipo de alinhamento permite que padrões relevantes sejam encontrados para todos os tipos sem prejuízo dos menos freqüentes. Por exemplo, enquanto padrões para blocos do tipo 1 são identificados com suporte mínimo igual a 57, os do tipo 0 e 2 possuem suportes 16 e 6, respectivamente, e mesmo com limites menores estes padrões são tão relevantes quanto os primeiros já que a porcentagem aplicada para definir ϵ é a mesma.

A saída desse programa é um arquivo no qual os padrões identificados (P) são impressos no formato do exemplo a seguir:

```
<pattern>
<freq>3</freq>
<what>det n</what>
<where>1 3 3</where>
</pattern>
```

no qual o padrão "`det n`" (conteúdo do elemento `what`) foi encontrado 3 vezes (conteúdo do elemento `freq`) no conjunto de seqüências da Tabela 24: uma vez na seqüência 1 e duas vezes na seqüência 3 (conteúdo do elemento `where`). As várias ocorrências de um padrão em uma mesma seqüência (linha) são indicadas pela repetição do número da seqüência no conteúdo do elemento `where`. Esse formato é o mesmo utilizado pelo `PrefixSpan` porém com a diferença de que, neste último, um identificador de seqüência aparece no máximo uma vez dentro do elemento <`where`>...</`where`> já que os padrões só podem ocorrer uma vez em cada seqüência.

Por fim, os padrões identificados são lidos e armazenados em um vetor associativo (`%padroes_fonte`) que tem como chave a seqüência de itens fonte correspondente ao padrão e como valor um vetor que armazena as informações a respeito dos exemplos nos quais esse padrão ocorre: o índice do exemplo e o vetor com as posições, nesse exemplo, dos itens que formam o padrão. O índice do exemplo e as posições dos itens que formam o padrão são recuperados verificando-se, para cada ocorrência de um padrão em um bloco (cada número entre <`where`> e </`where`>), qual o índice do exemplo que contém este bloco e quais as posições dos itens que formam este padrão, neste exemplo (informações contidas nos *hashes* de blocos fonte e alvo).

Por exemplo, considere o padrão fonte "`det n`" encontrado como descrito no exemplo anterior e suponha que as seqüências 1 e 2 da Tabela 24 correspondam a blocos do exemplo 42 e a seqüência 3, a um bloco do exemplo 45. Além disso, suponha que o padrão "`det n`" ocorra no exemplo 42 nas posições 5 (`det`) e 6 (`n`) e no 45 nas posições 3 (`det`) e 4 (`n`) e também nas posições 7 (`det`) e 8 (`n`). Desse modo, o padrão "`det n`" é armazenado, no vetor associativo `%padroes_fonte`, como apresentado a seguir:

```
$padroes_fonte{det n}
    [0] = (42,(5,6))
    [1] = (45,(3,4))
    [2] = (45,(7,8))
```

5.3.2.2 Identificação de padrões bilíngües

A partir do vetor associativo no qual os padrões fonte estão armazenados (`%padroes_fonte`) resultante do processo de indução monolíngüe apresentado na seção 5.3.2.1, os padrões bilíngües são buscados e, em seguida, filtrados como apresentado a seguir.

Para cada padrão fonte, buscam-se os itens alvo com os quais os itens do padrão fonte se alinham resultando em uma seqüência bilíngüe. A seqüência bilíngüe resultante só será considerada um padrão bilíngüe se satisfizer duas condições: representar o tipo de alinhamento sob estudo e possuir uma freqüência considerada relevante.

De acordo com a primeira condição, se o objetivo for encontrar padrões para blocos de tipo 0, a seqüência bilíngüe tem que, necessariamente, envolver pelo menos um alinhamento de omissão; se for encontrar padrões de tipo 1, os alinhamentos dos itens da seqüência devem respeitar a ordem de ocorrência desses itens e, para os blocos do tipo 2, exige-se que exista pelo menos um alinhamento com mudança de ordem. A verificação de tal condição é necessária já que, por exemplo, padrões com alinhamentos diferentes do tipo 2 podem ser obtidos a partir de seqüências de itens provenientes de blocos de alinhamento do tipo 2. Isso é possível uma vez que um bloco do tipo 2 pode englobar itens com alinhamentos de tipos distintos de 2, como descrito anteriormente (veja Figura 19).

O filtro por freqüência, por sua vez, considera como padrões bilíngües apenas aquelas seqüências bilíngües nas quais as partes fonte e alvo ocorrem alinhadas, no mínimo, ϵ vezes (o limite mínimo de freqüência usado para padrões bilíngües, no ReTraTos, é o mesmo dos padrões monolíngües).

Os padrões bilíngües resultantes desse processo de identificação de padrões são arma-

zenados em %padroes_bilingues tendo como chave o padrão fonte e como valor um vetor com todos os possíveis padrões alvo. Cada elemento desse vetor de possibilidades possui três informações: [0] a seqüência de itens que representa o padrão alvo, [1] um vetor com os alinhamentos dos itens fonte com os itens alvo e [2] um vetor com os dados dos exemplos nos quais os padrões fonte e alvo ocorrem alinhados, ou seja, o identificador do exemplo e as posições dos itens fonte e alvo.

Um exemplo de um padrão bilíngüe gerado para o padrão fonte "det n" identificado anteriormente é apresentado a seguir.

```
$padroes_bilingues{det n}
    [0] =
        [0] = 'det n'
        [1] = (1,2)
        [2] = ((42,(5,6),(6,7)),(45,(3,4),(8,9)),(45,(7,8),(15,16)),...)
    [1] =
        [0] = 'cnjcoo n'
        [1] = (1,2)
        [2] = ((61,(21,22),(29,30)),(62,(5,6),(11,12)),...)
```

Nesse exemplo, o padrão possui duas possíveis partes alvo – "det n" e "cnjcoo n" – e é do tipo 1, ou seja, os itens fonte e alvo ocorrem alinhados respeitando a ordem de ocorrência (o primeiro item fonte está alinhado com o primeiro item alvo e o segundo item fonte, com o segundo item alvo) como especificado pelo vetor de alinhamento: (1,2). Além disso, a primeira opção alvo ocorre alinhada com a parte fonte nos exemplos 42, 45 etc., enquanto a segunda opção alvo, nos exemplos 61, 62 etc.

Quando há mais de um padrão alvo para um dado padrão fonte, como no exemplo anterior, todas as possibilidades são armazenadas no vetor associativo que contém os padrões bilíngües (%padroes_bilingues). Essa ambigüidade é tratada em passos posteriores do processo de indução, como apresentado em detalhes na seção 5.3.4.

5.3.3 Geração das regras de tradução no ReTraTos

Após a etapa de identificação de padrões para cada tipo de bloco de alinhamento, separadamente, esses padrões são considerados em conjunto para a geração das regras. Assim, o processo de geração das regras tem como entrada os padrões bilíngües identificados na etapa anterior do processo de indução (como descrito na seção 5.3.2) e, como saída, o conjunto de

regras de tradução, ou seja, os padrões bilíngües com restrições. As restrições são definidas com base nos valores dos atributos morfológicos nos exemplos de tradução que deram origem ao padrão bilíngüe, em dois passos: criação e generalização.

As restrições mono ou bilíngües, no ReTraTos, podem ser de dois tipos: restrições de valor e restrições de concordância/valor. Com base nas idéias de (CARBONELL et al., 2002; PROBST, 2005), no ReTraTos, uma restrição de valor especifica qual(is) o(s) valor(es) esperado(s) para os atributos dos itens de um padrão. Uma restrição de concordância/valor, por sua vez, especifica quais itens de um ou ambos os lados possuem os mesmos valores para um ou mais atributos (restrição de concordância) e, ao mesmo tempo, especifica quais são esses valores (restrição de valor).

As restrições de valor são criadas de modo semelhante ao que é apresentado em (PROBST, 2005), já as restrições de concordância/valor são criadas de modo bem distinto. Enquanto Probst (2005) faz generalizações com base em estatística para determinar se uma restrição de concordância deve ser criada, no ReTraTos, nenhuma generalização é feita (todos os possíveis valores são indicados explicitamente) e, por isso, optou-se por denominar uma restrição de concordância como restrição de concordância/valor.

Mais especificamente, Probst (2005) cria restrições de concordância aplicando um teste de significância estatística ou uma heurística para determinar se um dado valor é encontrado, com freqüência, em duas restrições. Se a resposta for afirmativa, uma restrição de concordância (sem qualquer indicação de valor) é criada no lugar das restrições de valor menos gerais.

No ReTraTos, contudo, optou-se por manter o valor que deu origem à restrição de concordância para consultas futuras. Essa decisão foi tomada com base no cenário para o qual ReTraTos foi projetado, que é o de indução de regras que se aplicam a casos específicos (como inclusão ou remoção de artigos, preposições etc.) nos quais quanto mais informação a respeito de como e onde uma regra se aplica, maior a probabilidade de êxito decorrente de sua aplicação.

Assim, no primeiro passo de geração das regras, são criadas as restrições de valor e de concordância/valor entre os atributos nos lados fonte e alvo, separadamente, (restrições monolíngües) e ambos os lados (restrições bilíngües). Para a criação dessas restrições, são considerados os conjuntos de atributos fonte e alvo correspondentes às partes fonte e alvo de cada padrão bilíngüe para cada um dos exemplos de tradução a partir dos quais tal padrão, com tais atributos, foi gerado. Mais especificamente, trata-se de analisar o campo 'atr' nos

vetores de exemplos fonte e alvo apresentados na seção 5.1.1.

A criação de restrições de valor ou de concordância/valor é realizada percorrendo-se o vetor de exemplos a partir dos quais o padrão bilíngüe foi gerado e verificando-se os valores dos atributos de cada lado e de ambos os lados. Se o valor de um determinado atributo ocorrer apenas uma vez em cada lado do padrão bilíngüe, uma restrição de valor será criada. Porém, se esse valor ocorrer mais de uma vez em um (monolíngüe) ou ambos (bilíngüe) os lados, será criada uma restrição de concordância/valor.

As restrições são formadas por variáveis que identificam os itens e os atributos fonte ou alvo que elas restringem. As variáveis fonte possuem a forma `Xi_k` e as alvo, `Yj_h`; em que $i, j, k, h > 0$ identificam os itens (i ou j) e seus atributos (k ou h). Uma restrição de valor possui apenas uma variável fonte ou alvo, enquanto uma restrição de concordância/valor monolíngüe possui pelo menos duas variáveis fonte (ou alvo) e uma restrição de concordância/valor bilíngüe possui uma ou mais variáveis fonte e uma ou mais variáveis alvo. Tanto as restrições de valor quanto as de concordância/valor especificam o(s) valor(es) possível(eis) para os atributos que elas restringem.

De modo geral, uma restrição de valor é da forma $Vi\_j = val$ em que V pode ser `X` (variável fonte) ou `Y` (variável alvo) e val é o valor (ou enumeração de valores separados por '|') que essa variável pode assumir. Uma restrição de concordância/valor, por outro lado, pode ser monolíngüe ou bilíngüe dependendo das variáveis que engloba. Uma restrição de concordância/valor monolíngüe é da forma $\texttt{Xi}_1\texttt{\_k}_1\texttt{ = Xi}_2\texttt{\_k}_2\texttt{(,Xi}_3\texttt{\_k}_3\texttt{)*}= val$ (fonte) ou $\texttt{Yj}_1\texttt{\_h}_1\texttt{ = Yj}_2\texttt{\_h}_2\texttt{(,Yj}_3\texttt{\_h}_3\texttt{)*}= val$ (alvo); enquanto uma restrição de concordância/valor bilíngüe é da forma $\texttt{Xi}_1\texttt{\_k}_1\texttt{(,Xi}_2\texttt{\_k}_2\texttt{)* = Yj}_1\texttt{\_h}_1\texttt{(,Yj}_2\texttt{\_h}_2\texttt{)*}= val$.[6]

Por exemplo, considere o padrão bilíngüe "`det n`→`det n`" identificado em passos anteriores do processo de indução. Suponha que esse padrão bilíngüe, em um dos exemplos de tradução a partir dos quais foi extraído, possua como atributos da parte fonte os especificados pelo conjunto {<def><m><sg>, <m><sg>}, em que o primeiro elemento desse conjunto (<def><m><sg>) corresponde aos atributos da PoS do primeiro item fonte (`det`) e o segundo elemento desse conjunto (<m><sg>) corresponde aos atributos da PoS do segundo item fonte (`n`).[7] Uma restrição de valor é criada para o primeiro atributo do primeiro elemento desse conjunto, `X1_1 = def`, já que este valor ocorre apenas uma vez entre os atributos fonte.

Além disso, duas restrições de concordância/valor também são identificadas: uma entre o valor do segundo atributo do primeiro elemento do conjunto e o valor do primeiro

[6] A seqüência "$(B)*$" indica a ocorrência de zero ou mais elementos do tipo "B".

[7] A lista completa dos símbolos gramaticais usados neste projeto pode ser consultada no Apêndice A.

atributo do segundo elemento do conjunto (X1_2 = X2_1 = m); e outra entre o valor do terceiro atributo do primeiro elemento do conjunto e o valor do segundo atributo do segundo elemento do conjunto (X1_3 = X2_2 = sg). Em termos lingüísticos, essas restrições de concordância/valor estabelecem que há concordância de gênero (masculino, m) e número (singular, sg) entre o primeiro e o segundo item do padrão fonte.

No formalismo de (CARBONELL et al., 2002), as restrições de valor e de concordância identificam o atributo sendo restringido por seu nome. Por exemplo, a concordância de gêneros poderia ser expressa como ((Si gênero) = (Sj gênero)) (veja Figura 9), em que Si indica o i-ésimo item do padrão e Sj, o j-ésimo item. No caso do exemplo apresentado anteriormente, as restrições de concordância de gênero e número poderiam ser expressas, nesse formalismo, como ((S1 gênero) = (S2 gênero)) e ((S1 número) = (S2 número)).

Porém, no ReTraTos, não se sabe que "m" é um possível valor para o atributo gênero, nem que gênero é um possível atributo para uma dada PoS e, portanto, optou-se por uma representação mais genérica na qual a relação entre os atributos é determinada sem a necessidade de se especificar quais os nomes desses atributos ou mesmo quais os atributos possíveis para uma dada PoS.

A criação de restrições alvo e bilíngües é feita de modo semelhante. Por exemplo, considerando-se o padrão bilíngüe citado anteriormente (det n→det n) e um conjunto de atributos alvo igual ao conjunto fonte apresentado anteriormente ({<def><m><sg>, <m><sg>}), as restrições alvo seriam Y1_1 = def, Y1_2 = Y2_1 = m e Y1_3 = Y2_2 = sg resultando em 3 restrições bilíngües: (1) X1_1 = Y1_1 = def, (2) X1_2, X2_1 = Y1_2, Y2_1 = m e (3) X1_3, X2_2 = Y1_3, Y2_2 = sg.

Após a criação das restrições bilíngües, as restrições monolíngües que têm suas informações representadas nas restrições bilíngües são removidas para se evitar redundância. Por exemplo, todas restrições fonte (X1_1 = def, X1_2 = X2_1 = m, X1_3 = X2_2 = sg) e alvo (Y1_1 = def, Y1_2 = Y2_1 = m e Y1_3 = Y2_2 = sg) criadas no exemplo anterior são removidas do conjunto de restrições uma vez que as informações que representam estão expressas nas restrições bilíngües (X1_1 = Y1_1 = def, X1_2, X2_1 = Y1_2, Y2_1 = m e X1_3, X2_2 = Y1_3, Y2_2 = sg).

O processo de criação de restrições de valor e de concordância/valor se repete para todos os possíveis valores de atributos das partes fonte e alvo de um padrão bilíngüe.

Após a criação de restrições mono e bilíngües, o próximo passo para a geração das regras é a generalização das restrições. Nesse processo, para cada restrição criada no passo

anterior, verifica-se se há outra com apenas um valor de atributo diferente. Se tal restrição for encontrada, tais valores são concatenados dando origem a uma nova restrição generalizada.

Por exemplo, considere as restrições bilíngües apresentadas anteriormente (Conj1) e outras que diferem apenas no valor do atributo de número (Conj2):

Conj1: {`X1_1 = Y1_1 = def`, `X1_2, X2_1 = Y1_2, Y2_1 = m`, `X1_3, X2_2 = Y1_3, Y2_2 = sg`}
Conj2: {`X1_1 = Y1_1 = def`, `X1_2, X2_1 = Y1_2, Y2_1 = m`, `X1_3, X2_2 = Y1_3, Y2_2 = pl`}

Esses conjuntos possuem apenas uma diferença correspondente ao número (`sg` e `pl`), a qual está presente no valor do terceiro atributo do primeiro elemento (`det`) e no valor do segundo atributo do segundo elemento (`n`) tanto na parte fonte quanto na parte alvo. Desse modo, um novo conjunto é criado contendo os dois valores possíveis (em ordem alfabética crescente) separados pelo caractere "|":

ConjG: {`X1_1 = Y1_1 = def, X1_2, X2_1 = Y1_2, Y2_1 = m, X1_3, X2_2 = Y1_3, Y2_2 = pl|sg`}

O novo conjunto de restrições (ConjG) cobre os exemplos cobertos pelos conjuntos com base nos quais ele foi criado (Conj1 e Conj2) e, portanto, recebe como seu conjunto de exemplos a união dos conjuntos de exemplos de Conj1 e Conj2. Após a criação do conjunto generalizado, os conjuntos nos quais este se baseou são removidos. O processo de generalização das restrições se repete até que nenhuma nova generalização seja possível, resultando em conjuntos generalizados de restrições de valor e de concordância/valor.

Após a realização dos dois passos para a geração das regras, os padrões bilíngües com seus conjuntos de restrições são considerados como regras de tradução (embora ainda não estejam no formalismo apresentado na seção 5.1.3). Essas regras são armazenadas em um vetor associativo `%regras` tendo como chave a parte fonte e como valor um vetor com todas as possíveis partes alvo (a ambigüidade é tratada na etapa de filtragem das regras de tradução).

Cada elemento do vetor de possíveis partes alvo possui três informações: (`[0]`) a seqüência de itens que representa a parte alvo, (`[1]`) um vetor com indicações dos alinhamentos de cada item fonte com cada item alvo e (`[2]`) um vetor com todas as possíveis restrições de valor e de concordância/valor. Cada elemento do vetor de restrições, por sua vez, possui quatro informações: (`[0]`) um vetor com as restrições fonte, (`[1]`) um vetor com as restrições alvo, (`[2]`) um vetor com as restrições bilíngües e (`[3]`) um vetor com as informações dos exemplos a partir dos quais essa regra, com essas restrições, foi gerada – mesmas informações contidas no `%padroes_bilingues`: identificador do exemplo e posições

dos itens fonte e alvo nesse exemplo.

Assim, por exemplo, o padrão bilíngüe identificado na etapa anterior do processo de indução poderia dar origem a uma regra como a apresentada a seguir:

```
$regras{det n}
    [0] =
        [0] = 'det n'
        [1] = (1,2)
        [2] =
            [0] =
                [0] = ()
                [1] = ()
                [2] =
                    [0] = ('X1_1 = Y1_1 = def|dem|ind')
                    [1] = ('X1_2,X2_1 = Y1_2,Y2_1 = m')
                    [2] = ('X1_3,X2_2 = Y1_3,Y2_2 = pl|sg')
                [3] = ((42,(5,6),(6,7)),(45,(3,4),(8,9)),(45,(7,8),(15,16))...)
            [1] = ...
    [1] =
        [0] = 'cnjcoo n'
        [1] = (1,2)
        [2] = ...
```

Como se pode perceber, no exemplo apresentado anteriormente, todas as possíveis partes alvo são apresentadas ("`det n`" e "`cnjcoo n`"). O tratamento desse tipo de ambigüidade é um dos temas da próxima seção.

5.3.4 Filtragem das regras de tradução no ReTraTos

A filtragem é realizada geralmente com o objetivo de (1) diminuir o tamanho da gramática de tradução – o que, segundo (MENEZES & RICHARDSON, 2001), acelera o processo de TA – ou (2) resolver ambigüidades. No projeto ReTraTos, a diminuição do tamanho da gramática de tradução, primeiro objetivo da filtragem, é garantida ao se exigir que padrões mono e bilíngües ocorram um número mínimo de vezes (ϵ), como apresentado na seção 5.3.2.

As estratégias adotadas para se alcançar o segundo objetivo da filtragem, a resolução de ambigüidades, têm como base as idéias de (KAJI et al., 1992) e (PROBST et al., 2003;

PROBST, 2005). Kaji et al. (1992) utilizam informações semânticas para refinar as regras e resolver a ambigüidade e Probst et al. (2003), Probst (2005) citam como trabalho futuro a utilização de informações de níveis mais baixos (lexical, por exemplo) com o intuito de resolver casos em que palavras com a mesma PoS se comportam de maneira distinta. Em (FONT-LLITJÓS et al., 2005) os autores apresentam simulações de uma possível estratégia de refinamento de regras de tradução que tem como base o refinamento de entradas lexicais.

Como no projeto ReTraTos informações semânticas não estão disponíveis, a abordagem adotada para tentar resolver casos de ambigüidades em nível de PoS foi utilizar informações de outros níveis – valores de atributos (campo 'atr' do vetor de exemplos) ou valores lexicais (campo 'lex') – para tentar distinguir regras de tradução com mesma parte fonte e diferentes partes alvo. O algoritmo de filtragem das regras de tradução é apresentado na Figura 22.

Entrada
R: o conjunto com n regras do tipo $\langle s, O^T \rangle$
E: o conjunto de exemplos fonte
pf: porcentagem da freqüência da melhor opção alvo usada para determinar se o filtro deve ou não ser aplicado a uma outra opção menos freqüente
Saída
R: o conjunto de regras alterado após a filtragem

Simbologia
ϵ: suporte mínimo para que uma regra seja filtrada
s: parte fonte de uma regra
O^T: conjunto de opções alvo de uma dada regra

Algoritmo
1. **para cada** $\langle s, O^T \rangle \in R$ **faça** *#L1*
2. **se** $|O^T| > 1$ **então** *#regra ambígua*
3. $\epsilon \leftarrow$ ordena_filtra(O^T,pf) *#a melhor opção alvo está na primeira posição, 1*
4. **para** $i = 2 \ldots |O^T|$ **faça** *#L2*
5. **se** filtra_restricoes(O^T,i,ϵ) $= 0$ **então**
6. filtra_valores_lexicais(s,R,i,E,ϵ)
7. remove(O^T,i)
8. **fim_então**
9. **fim_para** *#L2*
10. **fim_então**
11. **fim_para** *#L1*

Figura 22: Algoritmo de filtragem das regras de tradução

De acordo com o algoritmo da Figura 22, considerando-se que cada regra é composta por uma parte fonte (s) e um conjunto de possíveis partes alvo (O^T), a filtragem é realizada para cada regra (laço L1) que possui mais de uma possível parte alvo ($|O^T| > 1$, linha 2), ou seja, para cada regra ambígua. Para essas regras ambíguas, antes de iniciar a filtragem com base nos valores de atributos ou valores lexicais, uma filtragem por freqüência é reali-

zada: o conjunto de opções alvo é ordenado por ordem decrescente de freqüência (subrotina `ordena_filtra(`O^T`,`pf`)`, linha 3)[8] e as opções alvo são filtradas com base na freqüência da melhor opção (na posição 1), $freq_{melhor}$, ou seja, todas as opções alvo com freqüência menor do que ϵ ($\epsilon = pf \times freq_{melhor}$) são desconsideradas. O intuito do filtro por freqüência é limitar a aplicação dos demais filtros àquelas opções que ocorram, no mínimo, ϵ vezes. A porcentagem de filtro, pf, assim como a porcentagem usada na identificação de padrões, pi, é um parâmetro do programa fornecido pelo usuário (ou considerado como 0,5 por padrão).

As opções alvo com uma freqüência considerada relevante são, então, filtradas (laço L2, linha 4) por restrições (linha 5) ou por valores lexicais (linha 6). A filtragem por restrições não cria uma nova regra, apenas diminui o conjunto de opções alvo mantendo aquelas que podem ser diferenciadas da melhor opção alvo por uma restrição fonte ou bilíngüe. A filtragem por valores lexicais, por outro lado, cria uma nova regra ao inserir restrições lexicais na parte fonte da regra sendo filtrada.

As duas estratégias de filtragem – por restrições e por valores lexicais – baseiam-se no mesmo princípio: a busca por algum valor que possa distinguir uma dada opção alvo menos freqüente da melhor opção alvo (a mais freqüente).

Na filtragem por restrições, as restrições de uma opção menos freqüente são comparadas aos conjuntos de restrições das n opções mais freqüentes do que ela em busca de um conjunto que seja capaz de torná-la única quando comparada a essas n opções. Embora a filtragem por restrições seja uma estratégia mais geral e, portanto, preferida em relação à filtragem por valores lexicais, nem sempre ela tem sucesso. Às vezes os valores dos atributos das PoS que formam a parte fonte da regra ambígua não são suficientes para diferenciar a escolha de uma ou outra opção alvo. Deste modo, quando a filtragem por restrições falha, a filtragem por valores lexicais é realizada seguindo um procedimento semelhante.

A filtragem por valores lexicais tenta diferenciar uma dada opção alvo menos freqüente verificando se há valores lexicais fonte que ocorrem apenas nessa opção, ou seja, valores lexicais únicos capazes de diferenciá-la das demais opções. Se um ou mais desses valores lexicais únicos forem encontrados, uma nova regra é criada com restrições lexicais na parte fonte para limitar sua aplicação às seqüências com tais valores lexicais.

Há, ainda, um outro filtro aplicado apenas quando o usuário delimita o escopo de criação das regras de tradução para PoS específicas. Assim, por meio de parâmetros passados ao programa, é possível restringir quais valores devem ou não estar presentes nas partes fonte

[8] A freqüência de uma opção alvo é dada pelo número de exemplos de tradução a partir dos quais essa opção foi gerada (informação contida no conjunto de restrições/exemplos vinculado a esta opção).

das regras induzidas. Por exemplo, pode-se especificar que as regras induzidas contenham em suas partes fonte preposições (`pr`) ou determinantes (`det`), mas não vírgulas (`cm`).

Se tal restrição for estabelecida, o processo de indução é otimizado para satisfazê-la: (1) apenas os blocos de alinhamento que satisfazem essas restrições são considerados na identificação dos padrões, (2) os padrões monolíngües identificados são filtrados para que apenas os que estão de acordo com essas restrições sejam considerados na busca por padrões bilíngües e (3) a filtragem das regras é realizada se as opções alvo menos freqüentes diferem nos itens especificados (filtro por escopo).

Como a implementação atual do ReTraTos gera apenas gramáticas não-ambíguas, quando todos os filtros falham, apenas a opção de maior freqüência é mantida.

5.3.5 Ordenação das regras de tradução no ReTraTos

A ordenação visa preparar as regras de tradução para serem usadas no sistema de TA especificando a ordem na qual elas devem ser aplicadas. Enquanto alguns autores ordenam as regras por especificidade (ou generalização) contando-se, por exemplo, a quantidade de símbolos terminais (palavras e não variáveis) (CICEKLI & GÜVENIR, 2001); outros as ordenam por meio da atribuição de pesos calculados com base em estatística (LAVOIE et al., 2001; ÖZ & CICEKLI, 1998).

No projeto ReTraTos a ordenação das regras é realizada implicitamente por meio do armazenamento da freqüência e do peso de cada regra. A freqüência de uma regra é o número de vezes em que suas partes fonte e alvo ocorrem alinhadas no conjunto de exemplos de tradução usado na indução; enquanto o peso é a probabilidade de ocorrência da regra, ou seja, sua freqüência dividida pela freqüência total das regras.

Freqüências e pesos são calculados não apenas para cada regra, mas também para cada opção alvo e cada conjunto de restrições. Para tanto, as freqüências das regras, das opções alvo e dos conjuntos de restrições são calculadas de modo incremental: a freqüência de uma regra é dada pela soma das freqüências de suas opções alvo; a freqüência de uma opção alvo é a soma das freqüências de seus conjuntos de restrições; e a freqüência de um conjunto de restrições é o número de exemplos a partir dos quais tais restrições foram derivadas (o número de elementos no conjunto de exemplos).

Em seguida, são atribuídos pesos para cada regra, opção alvo e conjunto de restrições calculados como suas freqüências divididas pelas freqüências totais das regras, das opções

alvo e dos conjuntos de restrições, respectivamente.

Ao final do processo de indução, a gramática de tradução possui regras não ambíguas (considerando-se que a filtragem foi realizada), dotadas de restrições que limitam sua aplicação e com informações de peso e freqüência, ou seja, prontas para serem utilizadas em um sistema de TA baseado em regras. Sendo assim, se em um dado momento da TA existir mais de uma regra passível de ser aplicada a uma entrada, certamente as regras candidatas contêm ou estão contidas umas nas outras. A resolução desse tipo de ambigüidade – que pode ser feita priorizando-se as regras maiores, as de maior peso etc. – não cabe ao processo de indução, mas sim ao sistema de TA, uma vez que está vinculada à estratégia de aplicação das regras adotada por tal sistema.

5.4 Tradução automática no ReTraTos

Após induzir as regras de tradução, o próximo passo é usá-las no processo de tradução de sentenças fonte em sentenças alvo. Para tanto, implementou-se um sistema de recombinação das regras induzidas automaticamente que recebe como entrada uma representação de uma sentença fonte (a sentença pré-processada, SF) e produz como saída uma representação da sentença alvo correspondente (sentença fonte transferida, SA). Mais especificamente, o sistema de recombinação equivale aos dois passos do processo de transferência apresentado na Figura 11: (1) busca/seleção e (2) aplicação das regras induzidas automaticamente.

Como apresentado no Capítulo 2, a busca das regras de tradução é realizada com base no casamento dos padrões existentes na SF e nas regras do repositório de regras de tradução, resultando em um conjunto de regras passíveis de serem aplicadas à sentença de entrada. A partir desse conjunto de regras candidatas, vários critérios podem ser usados para selecionar a melhor regra a ser aplicada em um determinado momento como: tamanho, especificidade, técnicas de aprendizado de máquina ou pesos baseados na freqüência ou na probabilidade das regras candidatas. Por fim, no último passo da transferência, a regra selecionada é aplicada, ou seja, um paralelo é estabelecido entre os itens no lado esquerdo da regra e os valores na SF e as transformações especificadas no lado direito da regra são realizadas resultando em uma seqüência de itens na língua alvo (SA).

O algoritmo do sistema de recombinação das regras induzidas implementado no ReTraTos é apresentado na Figura 23. De acordo com esse algoritmo, a sentença fonte de entrada (SF) é percorrida em busca da regra que se aplica à maior seqüência de itens fonte consecutivos. Essa busca é realizada pela sub-rotina `busca_candidatas(`C^S, s_i, C^R`)` (linha

6), a qual retorna (em C^R) as regras do repositório de regras (R) nas quais a parte fonte tem como prefixo C^S seguido pelo item s_i. Essa busca se repete para cada novo s_i até que o conjunto de regras candidatas se torne vazio ou s_i seja o último item de SF (linha 8).

Entrada
SF: a representação da senteça fonte
R: o conjunto de regras induzidas
B: o léxico bilíngüe
Saída
SA: a sentença alvo correspondente à fonte

Simbologia
s_i: i-ésimo item de SF
C^S: conjunto de itens fonte
C^R: conjunto de regras aplicáveis à seqüência de itens fonte $C^S s_i$

Algoritmo
1. SA $\leftarrow \emptyset$
2. $C^R \leftarrow R$
3. $C^S \leftarrow \emptyset$
4. $i = 1$
5. **faça** *#L1*
6. busca_candidatas(C^S, s_i, C^R)
7. **se** $C^R \neq \emptyset$ **então** $C^S \leftarrow C^S \cup s_i$
8. **se** $C^R = \emptyset$ **ou** $i = |SF|$
9. **então**
10. **se** $C^S = \emptyset$ **então** $SA \leftarrow SA \cup$ busca_lexico(s_i, B)
11. **senão**
12. $\alpha \leftarrow$ aplica_regra(C^S, R, B)
13. **se** $\alpha \neq \emptyset$ **então** $SA \leftarrow SA \cup \alpha$
14. **senão**
15. $i \leftarrow$ posicao_primeiro(C^S)
16. $SA \leftarrow SA \cup$ busca_lexico(s_i, B)
17. **fim_senão**
18. **fim_senão**
19. $C^R \leftarrow R$
20. $C^S \leftarrow \emptyset$
21. **fim_então**
22. $i \leftarrow i + 1$
23. **até_que** $i > |SF|$ *#L1*
24. **retorna** SA

Figura 23: Algoritmo de tradução usando as regras induzidas

Ao final dessa busca, a melhor (=maior) regra é aquela cuja parte fonte contém os itens em C^S. Se C^S é um conjunto vazio, então não existe nenhuma regra com parte fonte iniciada com s_i e, neste caso, s_i é traduzido apenas com base no léxico bilíngüe (linha 10). Caso contrário, a sub-rotina `aplica_regra(`C^S, R, B`)` (linha 12) busca o melhor conjunto de restrições fonte, aplica as transformações especificadas no conjunto de restrições bilíngües e retorna a tradução (α) consultando o léxico bilíngüe quando necessário. Dentro da sub-rotina

`aplica_regra`, se nenhuma das opções de restrições fonte for compatível com a seqüência de itens fonte, esta seqüência é diminuída removendo-se o último item e a nova parte fonte é considerada na busca por um conjunto de restrições aplicáveis se repete. Caso não seja possível traduzir nenhuma subseqüência de itens de C^S, o primeiro item em C^S será traduzido com base apenas no léxico bilíngüe (linhas 15 e 16) e a busca continua a partir de seu sucessor.

Após a tradução de uma seqüência de itens fonte, os conjuntos de regras candidatas (C^R) e de itens fonte (C^S) são reiniciados (linhas 19 e 20) e o processo de busca–seleção–aplicação se repete até que toda a sentença fonte tenha sido processada (laço L1). Por fim, a seqüência de itens resultantes na língua alvo, SA, é retornada pelo algoritmo (linha 24).

6 Avaliação no ReTraTos

Este capítulo descreve a metodologia adotada e os resultados obtidos na avaliação dos recursos lingüísticos induzidos automaticamente no projeto ReTraTos: léxicos bilíngües (seção 6.1) e regras de tradução (seção 6.2).

6.1 Avaliação dos léxicos bilíngües no ReTraTos

Os léxicos bilíngües para os pares `pt`–`es` e `pt`–`en` foram induzidos utilizando o método de indução de léxicos descrito na seção 5.2 com um limite de freqüência mínima para as entradas de multipalavras definido empiricamente como 50. Além desse, outros parâmetros de entrada – como a especificação de valores de atributos e informações necessárias para formatar os léxicos de acordo com o padrão de `Apertium` – também foram usados.

Os léxicos induzidos possuem o formato apresentado na seção 5.1.2 do Capítulo 5, segundo o qual cada entrada pode conter: sentido de tradução (especificado como o valor do atributo `r` do elemento <e>) e informações retornadas pelo etiquetador morfossintático (agrupadas entre as etiquetas <l> e </l>, na parte fonte, e entre as etiquetas <r> e </r>, na parte alvo).

Apesar do método utilizado para induzir os léxicos ser o mesmo, a metodologia empregada para avaliá-los apresenta algumas distinções, como descrito nas seções 6.1.1 e 6.1.2 para os léxicos induzidos para os pares `pt`–`es` e `pt`–`en`, respectivamente.

6.1.1 Avaliação do léxico bilíngüe `pt`–`es`

O léxico bilíngüe induzido para o par `pt`–`es` foi gerado tendo como língua fonte o `es` e como língua alvo o `pt`. Essa decisão foi tomada para facilitar sua comparação com o léxico bilíngüe

es–pt usado no sistema de TA Apertium[1] na avaliação intrínseca automática (seção 6.1.1.1). Além dessa, outra metodologia de avaliação também foi empregada: a avaliação intrínseca manual (seção 6.1.1.2) das entradas.

6.1.1.1 Avaliação intrínseca automática do léxico bilíngüe pt–es

Na primeira metodologia de avaliação empregada para avaliar as entradas do léxico bilíngüe es–pt (LR), as entradas de palavras e multipalavras foram comparadas de modo automático às entradas presentes no léxico bilíngüe de Apertium (LA) e agrupadas, separadamente, em um dos três grandes grupos apresentados a seguir.

- **IDÊNTICAS** – Uma entrada de LR idêntica a uma entrada de LA é aquela na qual as partes fonte e alvo, bem como o sentido de tradução, são exatamente iguais nos dois léxicos. Exemplo de uma entrada idêntica para a palavra em es "*trofeo*":

LR

```
<e>
  <p>
    <l>trofeo<s n="n"/></l>
    <r>troféu<s n="n"/></r>
  </p>
</e>
```

LA

```
<e>
  <p>
    <l>trofeo<s n="n"/></l>
    <r>troféu<s n="n"/></r>
  </p>
</e>
```

- **NOVAS** – Uma entrada nova de LR é aquela na qual a parte fonte não aparece em nenhuma entrada de LA (a seqüência NC indica a "não consta") ou aparece com um sentido de tradução e uma correspondência alvo diferentes (o que representa uma nova entrada). Exemplo de uma entrada nova para a multipalavra "*reino unido*":

LR	**LA**
<e>	NC
reino<b/>unido<s n="np"/>	NC
</e>	NC

- **DIFERENTES** – Uma entrada de LR diferente de uma entrada de LA é aquela na qual a forma base e a PoS fonte são as mesmas nas entradas de LR e LA mas há alguma diferença no sentido de tradução, na parte alvo etc. Exemplo de uma entrada diferente para a palavra em es "*aceite*":

[1] O léxico bilíngüe es–pt de Apertium (versão 0.9 de 05/05/2006) usado nessa comparação, assim como os demais recursos lingüísticos de Apertium, foi gerado manualmente.

LR

```
<e>
  <p>
    <l>aceite<s n="n"/></l>
    <r>óleo<s n="n"/></r>
  </p>
</e>
```

LA

```
<e>
  <p>
    <l>aceite<s n="n"/></l>
    <r>azeite<s n="n"/></r>
  </p>
</e>
```

Embora a classificação das entradas de LR nesses três grandes grupos forneça uma idéia de como elas se comportam em relação às entradas de LA, no caso das entradas novas e diferentes é interessante especificar qual a informação na entrada de LR é nova ou diferente em relação às entradas de LA. Para tanto, esses três grandes grupos foram subdivididos em 14 classes apresentadas a seguir.

- **IDÊNTICAS**

  1. **Idêntica** – as partes fonte e alvo e o sentido de tradução da entrada em LR são exatamente iguais aos encontrados em LA;

- **NOVAS**

  2. **Incompleta fonte** – a parte fonte da entrada de LR não possui informações morfossintáticas (a palavra fonte não foi reconhecida pelo etiquetador);
  3. **Incompleta alvo** – a parte alvo da entrada de LR não possui informações morfossintáticas (a palavra alvo não foi reconhecida pelo etiquetador);
  4. **Incompleta** – as partes fonte e alvo da entrada de LR não possuem informações morfossintáticas (as palavras fonte e alvo não foram reconhecidas pelos etiquetadores);
  5. **Nova** – as partes fonte e alvo da entrada de LR estão completas (foram etiquetadas morfossintaticamente) e a parte fonte não é encontrada em nenhuma entrada de LA;
  6. **Novo sentido** – a parte fonte da entrada de LR ocorre em LA, porém com um sentido de tradução e uma correspondência (parte alvo) diferentes, ou seja, a parte fonte em LR possui uma nova correspondência para um novo sentido de tradução;

- **DIFERENTES**

7. **Diferente** – a parte fonte da entrada em LR ocorre em uma entrada de LA com o mesmo sentido de tradução, mas com uma parte alvo diferente;
8. **Categoria alvo diferente** – a parte fonte da entrada em LR ocorre em uma entrada de LA com a mesma forma base alvo, porém com uma PoS alvo diferente da encontrada em LR;
9. **Sentido mais específico** – a parte fonte da entrada em LR ocorre em uma entrada de LA com a mesma parte alvo, porém LR especifica o sentido de tradução válido para a entrada enquanto LA não o faz;
10. **Sentido mais geral** – a parte fonte da entrada em LR ocorre em uma entrada de LA com a mesma parte alvo, porém LA especifica o sentido de tradução válido para a entrada enquanto LR não o faz;
11. **Sentido diferente** – a parte fonte da entrada em LR ocorre em uma entrada de LA com a mesma parte alvo, mas com um sentido de tradução diferente;
12. **Atributos mais específicos** – a forma base e a PoS fonte da entrada em LR ocorrem em uma entrada de LA com as mesmas forma base e PoS alvo e o mesmo sentido de tradução, mas a entrada de LR especifica atributos fonte ou alvo não especificados na entrada de LA;
13. **Atributos mais gerais** – a forma base e a PoS fonte da entrada em LR ocorrem em uma entrada de LA com as mesmas forma base e PoS alvo e o mesmo sentido de tradução, mas a entrada de LA especifica atributos fonte ou alvo não especificados na entrada de LR;
14. **Atributos diferentes** – a forma base e a PoS fonte da entrada em LR ocorrem em uma entrada de LA com as mesmas forma base e PoS alvo e o mesmo sentido de tradução, mas a entrada de LR possui um ou mais atributos fonte ou alvo diferentes da entrada de LA.

É importante citar que as entradas foram comparadas seguindo a ordem decrescente de prioridade para: forma base, PoS, sentido de tradução e atributos. Assim, no momento da classificação de uma entrada que difere, por exemplo, no sentido de tradução e nos valores dos atributos, essa entrada será classificada como “sentido diferente”, pois o sentido de tradução tem maior prioridade em relação aos valores de atributos.

Considerando-se a classificação apresentada anteriormente, as 23.450 entradas de LR (23.129 de palavras e 321 de multipalavras) foram comparadas às 11.288 entradas de LA

(10.360 de palavras e 928 de multipalavras) e classificadas em uma das 14 classes.[2] A grande diferença no número de entradas de palavras nos dois léxicos se deve, em parte, ao fato do léxico induzido automaticamente possuir várias entradas para a mesma correspondência entre palavras fonte e alvo com pequenas variações nos valores de seus atributos (14,43% das entradas de LR possuem atributos mais específicos do que as entradas de LA). Comparando-se apenas a quantidade de *types* (formas base independentemente do número de entradas existentes para elas), a cobertura de LA (9.812 *types*) pode ser aumentada para 22.826 *types* com a inserção de entradas de LR não presentes em LA.

Os resultados dessa primeira etapa da avaliação – avaliação intrínseca automática – do léxico bilíngüe induzido automaticamente para o par `pt–es` são apresentados na Tabela 25. Cerca de 13% das entradas de palavras e 15% das entradas de multipalavras em LR são idênticas às encontradas em LA. Além disso, cerca de 23% das entradas de palavras e 13% das entradas de multipalavras diferem em algum aspecto nos dois léxicos – o que não significa, necessariamente, que as entradas em LR não são válidas. Contudo, o dado mais relevante levantado nessa primeira análise está no número de entradas novas: cerca de 63% das entradas de palavras e 72% das entradas de multipalavras induzidas automaticamente não aparecem no léxico bilíngüe de `Apertium`.

6.1.1.2 Avaliação intrínseca manual do léxico bilíngüe `pt–es`

Assim como em (SCHAFER & YAROWSKY, 2002), a avaliação intrínseca automática do léxico induzido no ReTraTos restringiu o escopo de entradas a serem avaliadas manualmente para as entradas novas e diferentes (uma vez que as idênticas já podem ser consideradas válidas) as quais estão agrupadas em 13 classes e representam 86,53% do total de entradas induzidas. Dessas entradas, sete das treze classes são descartadas da avaliação manual por um dos três motivos apresentados a seguir:

1. estão incompletas – incompleta fonte (`es`) (15,77% das entradas), incompleta alvo (`pt`) (0,89% das entradas) e incompleta (20,20% das entradas) – e, por isso, necessitam a etiquetação morfossintática manual das palavras fonte ou alvo antes de serem utilizadas na TA;

[2]Na realidade, LR possui 23.804 entradas, das quais 354 foram excluídas da comparação automática uma vez que representam um tipo de multipalavra (formatada com o elemento <`j/`>) não presente na versão 0.9 do léxico bilíngüe `es–pt` de `Apertium`. LA, por sua vez, possui 11.307 entradas (<`e`>...<`/e`>), porém 19 delas não foram incluídas na comparação automática já que 7 utilizam expressões regulares na definição de símbolos (<`re`>...<`/re`>) e 12 são utilizadas para a definição de paradigmas (<`pardefs`>...<`/pardefs`>).

Tabela 25: Resultados da avaliação intrínseca automática do léxico induzido no ReTraTos (LR) com o léxico utilizado no `Apertium` (LA) para o par `pt`–`es`

	Palavras		**Multipalavras**		**Todas**	
Classe	#	%	#	%	#	%
IDÊNTICAS	3.111	13,45	48	14,95	3.159	13,47
Idêntica	3.111	13,45	48	14,95	3.159	13,47
NOVAS	14.675	63,45	231	71,96	14.906	63,57
Incompleta fonte	3.685	15,93	14	4,36	3.699	15,77
Incompleta alvo	209	0,90	0	0,00	209	0,89
Incompleta	4.736	20,48	0	0,00	4.736	20,20
Nova	5.119	22,13	165	51,40	5.284	22,53
Novo sentido	926	4,00	52	16,20	978	4,17
DIFERENTES	5.343	23,10	42	13,08	5.385	22,96
Diferente	1.352	5,85	35	10,90	1.387	5,91
Categoria alvo diferente	7	0,03	0	0,00	7	0,03
Sentido mais específico	398	1,72	6	1,87	404	1,72
Sentido mais geral	198	0,86	1	0,31	199	0,85
Sentido diferente	22	0,10	0	0,00	22	0,09
Atributos mais específicos	3.338	14,43	0	0,00	3.338	14,23
Atributos mais gerais	28	0,12	0	0,00	28	0,12
Atributos diferentes	0	0,00	0	0,00	0	0,00
Total	**23.129**	**100,00**	**321**	**100,00**	**23.450**	**100,00**

2. especificam as mesmas correspondências encontradas em LA, porém com mais informações a respeito do sentido de tradução – sentido mais específico (1,72% das entradas) – ou dos atributos fonte ou alvo – atributos mais específicos (14,23% das entradas) – e, portanto, já são contempladas pelas entradas de LA;

3. provavelmente estão incorretas – categoria alvo diferente (0,03% das entradas) e atributos diferentes (0% das entradas).

As entradas restantes (33,67% do total de entradas induzidas) pertencem às seis classes avaliadas manualmente: nova (22,53% das entradas), novo sentido (4,17% das entradas), diferente (5,91% das entradas), sentido mais geral (0,85% das entradas), sentido diferente (0,09% das entradas) e atributos mais gerais (0,12% das entradas).

Amostras aleatórias com cerca de 10% das entradas de palavras e multipalavras em cada uma dessas seis classes foram geradas para serem avaliadas manualmente. Essas amostras, na verdade, foram divididas entre dois juízes com conhecimentos dos idiomas `pt` e `es` (um nativo do `pt` e outro nativo do `es`) cabendo a cada um avaliar 6% delas. A sobreposição de 2% foi determinada propositalmente para medir a concordância entre os juízes por meio do cálculo do valor da medida Kappa (CARLETTA, 1996).

Cada juiz avaliou manualmente 474 entradas (459 para palavras e 15 para multipalavras) classificando-as como:

Válida (V) a correspondência entre as partes fonte e alvo da entrada é válida, ou seja, a parte fonte é uma possível tradução da parte alvo considerando-se o sentido de tradução especificado;

Parcialmente válida (PV) a correspondência entre as partes fonte e alvo da entrada seria válida se alguma alteração nas informações morfossintáticas (PoS ou atributos) ou sentido de tradução fosse feita;

Não válidas (NV) a correspondência entre as partes fonte e alvo não é válida.

Além dessas três classes, outras duas foram usadas para identificar as entradas com erro de grafia (EG) e as que envolviam termos científicos ou estrangeiros (TE) – as quais não fazem parte do escopo do léxico e, portanto, não devem ser consideradas na avaliação final.

Os resultados da avaliação manual das entradas de palavras e multipalavras são apresentados, respectivamente, nas Tabelas 26 e 27. O valor levantado para kappa, nesse experimento, foi 0,63 indicando uma boa concordância entre os juízes já que os mesmos discordaram em 15,61% das entradas de palavras e 6,06% das entradas de multipalavras.

Tabela 26: Classificação manual das entradas de palavras no léxico `es`–`pt` induzido automaticamente no ReTraTos

		Juiz 1					**Juiz 2**				
Tipo de entrada	**#**	**V**	**PV**	**NV**	**TE**	**EG**	**V**	**PV**	**NV**	**TE**	**EG**
Nova	306	211	75	17	3	0	170	97	37	0	2
Novo sentido	54	25	16	13	0	0	25	13	16	0	0
Diferente	81	48	21	10	2	0	47	13	19	2	0
Sentido mais geral	12	11	1	0	0	0	10	2	0	0	0
Sentido diferente	3	3	0	0	0	0	3	0	0	0	0
Atributo mais geral	3	3	0	0	0	0	3	0	0	0	0
Total	**459**	**301**	**113**	**40**	**5**	**0**	**258**	**125**	**72**	**2**	**2**

A quantidade de entradas classificadas como não-válidas foi relativamente pequena para as entradas de palavras – cerca de 9% e 16% das entradas classificadas pelos juízes 1 e 2, respectivamente – porém muito grande para as entradas de multipalavras – cerca de 47% das entradas classificadas por ambos os juízes. A menor qualidade das entradas de multipalavras já era esperada uma vez que elas são geradas, principalmente, com base nos

Tabela 27: Classificação manual das entradas de multipalavras no léxico es–pt induzido automaticamente no ReTraTos

		Juiz 1					Juiz 2				
Tipo de entrada	**#**	**V**	**PV**	**NV**	**TE**	**EG**	**V**	**PV**	**NV**	**TE**	**EG**
Nova	9	6	0	3	0	0	1	1	7	0	0
Novo sentido	3	2	0	1	0	0	2	1	0	0	0
Diferente	3	0	0	3	0	0	2	1	0	0	0
Total	**15**	**8**	**0**	**7**	**0**	**0**	**5**	**3**	**7**	**0**	**0**

alinhamentos lexicais automáticos nos quais os alinhamentos $n : m$ apresentam uma alta taxa de erro (AER = 11,19%).

Além disso, a estratégia de união dos alinhamentos lexicais gerados nos dois sentidos de tradução (fonte–alvo e alvo–fonte) aplicada no pré-processamento do *corpus* aumenta a cobertura do alinhador em detrimento de sua precisão. A maioria dos problemas nas entradas de multipalavras está relacionada a correspondências incompletas como a encontrada entre a expressão multipalavra, em es, *como consecuencia* e a palavra, em pt, *conseqüência*.

É importante citar, também, que muitas entradas, embora representem correspondências possíveis no contexto do alinhamento de palavras, foram consideradas como não-válidas no contexto de um léxico bilíngüe. Por exemplo, a entrada apresentada a seguir foi classificada como não-válida embora, no contexto do alinhamento lexical seja totalmente possível alinhar o substantivo em es *organizadoras* com o substantivo em pt *organização*.

```
<e r="LR">
  <p>
    <l>organizador<s n="n"/><s n="f"/><s n="pl"/></l>
    <r>organização<s n="n"/><s n="f"/><s n="sg"/></r>
  </p>
</e>
```

As entradas classificadas como parcialmente válidas – aproximadamente 24% pelo juiz 1 e 27% pelo juiz 2 – são, na maioria dos casos, decorrentes de erros de etiquetação. Por exemplo, a entrada apresentada a seguir seria uma entrada válida se a palavra em es *sofisticado* tivesse sido etiquetada como adjetivo (adj) e não como verbo (vblex).

```
<e>
  <p>
    <l>sofisticar<s n="vblex"/><s n="pp"/><s n="m"/><s n="sg"/></l>
    <r>requintado<s n="adj"/><s n="m"/><s n="sg"/></r>
  </p>
```

</e>

Com o intuito de filtrar os erros de etiquetação morfossintática, as entradas classificadas como parcialmente válidas (PV) nas quais o motivo da não-validade é um erro de etiquetação foram desconsideradas na avaliação geral. Assim, a Tabela 28 apresenta os resultados da avaliação geral, ou seja, as quantidades (#) e as respectivas porcentagens (%) de entradas válidas de palavras, multipalavras e ambas calculadas desconsiderando-se as entradas classificadas como TE, EG e PV com erro de etiquetação morfossintática.

Tabela 28: Resultados da avaliação intrínseca manual do léxico bilíngüe induzido por ReTraTos para o par `pt`–`es`

	Palavras			**Multipalavras**			**Todas**		
Classe	**#**	**# V**	**% V**	**#**	**#V**	**% V**	**#**	**#V**	**% V**
NOVAS	463	344	74,30	19	8	42,11	482	352	73,03
Nova	396	307	77,53	14	5	35,71	410	312	76,10
Novo sentido	67	37	55,22	5	3	60,00	72	40	55,56
DIFERENTES	135	103	76,30	5	0	0	140	103	73,57
Diferente	107	76	71,03	5	0	0	112	76	67,86
Sentido mais geral	18	17	94,44	0	0	0	18	17	94,44
Sentido diferente	5	5	100	0	0	0	5	5	100
Atributos mais gerais	5	5	100	0	0	0	5	5	100
Total	**598**	**447**	**74,75**	**24**	**8**	**33,33**	**622**	**455**	**73,15**

Com base nos valores da Tabela 28, pode-se concluir que cerca de 75% das entradas de palavras e 33% das entradas de multipalavras são válidas. É importante mencionar que neste cálculo, entre as entradas avaliadas por ambos os juízes, apenas aquelas classificadas como válidas por ambos foram consideradas.

Desse modo, o conjunto formado pelas entradas idênticas (3.159) e as derivadas das classes avaliadas manualmente (6.262 novas com 73,03% de precisão e 1.636 entradas diferentes com 73,57% de precisão) possui 11.057 entradas com uma precisão estimada em 81%. Além disso, considerando-se apenas as entradas bilíngües das classes de melhor desempenho – palavras: nova (5.119), diferente (1.352), sentido mais geral (198), sentido diferente (22) e atributo mais geral (28); e multipalavras: novo sentido (52) – tem-se um conjunto com 6.771 entradas – melhores classes induzidas – com uma precisão estimada em 77%. Esse conjunto unido com as entradas idênticas dá origem a um novo conjunto com 9.930 entradas – melhores classes induzidas + idênticas – com uma precisão estimada em 84%.

6.1.2 Avaliação do léxico bilíngüe pt–en

O léxico bilíngüe pt–en foi avaliado seguindo a metodologia de avaliação intríseca manual (seção 6.1.2.1) já que não estava disponível, no momento da avaliação, um léxico bilíngüe pt–en de referência (nos moldes do léxico es–pt encontrado em Apertium) nem um sistema de TA no qual as entradas pudessem ser testadas. Porém, para restringir o escopo das entradas avaliadas manualmente, uma classificação automática foi realizada comparando-se a parte fonte da entrada bilíngüe com a parte alvo correspondente.

Assim, as entradas do léxico pt–en foram classificadas em três grandes grupos – iguais, incompletas e diferentes – apresentados a seguir:

- **IGUAIS** – Uma entrada do léxico bilíngüe pt–en é classificada como igual quando as informações morfossintáticas nas partes fonte e alvo são iguais. Exemplo de uma entrada igual para a palavra em pt *abaixo*:

  léxico bilíngüe pt–en

```
<e>
  <p>
    <l>abaixo<s n="adv"/></l>
    <r>below<s n="adv"/></r>
  </p>
</e>
```

- **INCOMPLETAS** – Uma entrada do léxico bilíngüe pt–en é classificada como incompleta quando a parte fonte ou a parte alvo não possui informação morfossintática. Exemplo de uma entrada incompleta para a palavra em pt *piracicaba*:

  léxico bilíngüe pt–en

```
<e>
  <p>
    <l>piracicaba<s n="n"/><s n="f"/><s n="sg"/></l>
    <r>piracicaba</r>
  </p>
</e>
```

- **DIFERENTES** – Uma entrada do léxico bilíngüe pt–en é classificada como diferente quando as partes fonte e alvo diferem no valor de alguma informação morfossintática. Exemplo de uma entrada diferente para a palavra em pt *aborígine*:

léxico bilíngüe `pt–en`

```
<e>
  <p>
    <l>aborígine<s n="adj"/><s n="mf"/></l>
    <r>aborigine<s n="n"/></r>
  </p>
</e>
```

Além da classificação das entradas em um desses três grandes grupos elas foram, ainda, subdivididas em oito classes similares às usadas na avaliação do léxico `es–pt` só que, desta vez, produzidas comparando-se as partes fonte e alvo da entrada bilíngüe:

- **IGUAIS**
  1. **Categorias e atributos iguais** – as informações morfossintáticas fonte e alvo são iguais;
- **INCOMPLETAS** (definidas na seção 6.1.1.1)
  2. **Incompleta fonte**
  3. **Incompleta alvo**
  4. **Incompleta**
- **DIFERENTES**
  5. **Categoria diferente** – a PoS da parte fonte não é a mesma da parte alvo;
  6. **Atributos mais específicos** – a parte fonte especifica mais atributos morfossintáticos do que a parte alvo;
  7. **Atributos mais gerais** – a parte fonte especifica menos atributos morfossintáticos do que a parte alvo;
  8. **Atributos diferentes** – a parte fonte e a parte alvo especificam atributos morfossintáticos porém, pelo menos um desses atributos possui valores diferentes nos dois lados.

Considerando-se a classificação apresentada anteriormente, as 19.191 entradas do léxico `pt–en` (15.949 de palavras e 3.242 de multipalavras) foram classificadas automaticamente em uma das 8 classes resultando nos valores apresentados na Tabela 29. Cerca de 6%

das entradas de palavras e 16% das entradas de multipalavras possuem mesma PoS e atributos nos lados fonte e alvo. Além disso, cerca de 29% das entradas de palavras e 12% das entradas de multipalavras estão incompletas. Por fim, cerca de 65% das entradas de palavras e 72% das entradas de multipalavras apresentam alguma diferença na PoS ou atributos fonte e alvo.

Tabela 29: Classificação automática das entradas no léxico induzido no ReTraTos para o par pt–en

	Palavras		**Multipalavras**		**Todas**	
Classe	#	%	#	%	#	%
IGUAIS	962	6,03	525	16,19	1.487	7,75
Categorias e atributos iguais	962	6,03	525	16,19	1.487	7,75
INCOMPLETAS	4.635	29,06	390	12,03	5.025	26,18
Incompleta fonte	1.094	6,86	380	11,72	1.474	7,68
Incompleta alvo	1.244	7,80	6	0,19	1.250	6,51
Incompleta	2.297	14,40	4	0,12	2.301	11,99
DIFERENTES	10.352	64,91	2.327	71,78	12.679	66,07
Categoria diferente	2.642	16,57	1.217	37,54	3.859	20,11
Atributos mais específicos	5.253	32,94	76	2,34	5.329	27,77
Atributos mais gerais	123	0,77	49	1,51	172	0,90
Atributos diferentes	2.334	14,63	985	30,38	3.319	17,29
Total	**15.949**	**100,00**	**3.242**	**100,00**	**19.191**	**100,00**

6.1.2.1 Avaliação intrínseca manual do léxico bilíngüe pt–en

Após a classificação automática das entradas do léxico bilíngüe pt–en, constatou-se que algumas entradas não deveriam ser avaliadas manualmente, como já havia sido detectado para o par pt–es. No caso do léxico pt–en essas entradas pertencem a quatro classes: categoria diferente (20,11% das entradas), incompleta fonte (7,68% das entradas), incompleta alvo (6,51% das entradas) e incompleta (11,99% das entradas). Embora possam indicar correspondências válidas, as entradas com PoS diferentes nos lados fonte e alvo são, em grande parte, ocasionadas por erros de etiquetação. As entradas incompletas, por sua vez, necessitam de especialistas humanos para a inserção de informações morfossintáticas essenciais.

As entradas restantes (53,71% do total de entradas induzidas) pertencem às quatro classes avaliadas manualmente: categorias e atributos iguais (7,75%), atributos mais específicos (27,77%), atributos mais gerais (0,90%) e atributos diferentes (17,29%). Amostras aleatórias com cerca de 10% das entradas de palavras e multipalavras em cada uma dessas quatro classes foram geradas para serem avaliadas por dois juízes com conhecimentos dos idiomas pt e en (ambos nativos do pt e tradutores profissionais pt–en–pt), com uma sobreposição de 2%.

Cada juiz avaliou manualmente 618 entradas (519 para palavras e 99 para multipalavras) classificando-as como: válida (V), parcialmente válida (PV), não válidas (NV) e termos científicos ou estrangeiros (TE). Nenhuma entrada com erro de grafia (EG) foi encontrada na avaliação do léxico `pt–en`.[3]

Os resultados da avaliação manual das entradas de palavras e multipalavras são apresentados, respectivamente, nas Tabelas 30 e 31. O valor levantado para kappa, nesse experimento, foi 0,48. Esse baixo valor de kappa reflete a baixa concordância na análise das entradas de multipalavras: os juízes discordaram na avaliação de 45,45% das entradas de multipalavras e em 16,18% das entradas de palavras.

Essa discordância, embora seja significante, não foi considerada relevante já que foi causada porque um dos juízes, na maioria das entradas de multipalavras, sugeriu a especificação de um valor para o sentido da tradução (classificando a entrada como PV) e o outro não julgou necessária tal alteração. Como apontado por Craggs & Wood (2005), é impossível estabelecer limites únicos para a kappa com base nos quais todas as codificações podem ser julgadas. Segundo esses autores, cada um deve decidir, com base no uso pretendido para o esquema de codificação, se os níveis de concordância observados são suficientes e, assim, realizar a análise dos resultados.

Tabela 30: Classificação manual das entradas de palavras no léxico `pt–en` induzido automaticamente no ReTraTos

		Juiz 1				**Juiz 2**			
Tipo de entrada	**#**	**V**	**PV**	**NV**	**TE**	**V**	**PV**	**NV**	**TE**
Categorias e atributos iguais	57	53	0	4	0	54	0	3	0
Atributos mais específicos	315	284	8	23	0	291	3	9	12
Atributos mais gerais	6	5	1	0	0	6	0	0	0
Atributos diferentes	141	83	41	17	0	70	58	11	2
Total	**519**	**425**	**50**	**44**	**0**	**421**	**61**	**23**	**14**

Tabela 31: Classificação manual das entradas de multipalavras no léxico `pt–en` induzido automaticamente no ReTraTos

		Juiz 1				**Juiz 2**			
Tipo de entrada	**#**	**V**	**PV**	**NV**	**TE**	**V**	**PV**	**NV**	**TE**
Categorias e atributos iguais	30	10	7	13	0	24	2	3	1
Atributos mais específicos	6	2	0	4	0	5	0	1	0
Atributos mais gerais	3	1	0	2	0	3	0	0	0
Atributos diferentes	60	15	5	40	0	20	23	15	2
Total	**99**	**28**	**12**	**59**	**0**	**52**	**25**	**19**	**3**

A Tabela 32 apresenta os resultados da avaliação geral, ou seja, as quantidades (#)

[3]A descrição de cada uma das classes usadas na avaliação manual pode ser obtida na seção 6.1.1.

e as respectivas porcentagens (%) de entradas válidas de palavras, multipalavras e ambas desconsiderando-se as entradas classificadas como TE.

Tabela 32: Resultados da avaliação intrínseca manual do léxico bilíngüe induzido no ReTraTos para o par pt–en

	Palavras			**Multipalavras**			**Todas**		
Classe	**#**	**# V**	**% V**	**#**	**#V**	**% V**	**#**	**#V**	**% V**
IGUAIS	95	89	93,68	49	26	53,06	144	115	79,86
Categoria e atributo igual	95	89	93,68	49	26	53,06	144	115	79,86
DIFERENTES	756	605	80,03	113	33	29,20	869	638	73,42
Atributos mais específicos	513	472	92,01	10	5	50,00	523	477	91,20
Atributos mais gerais	10	9	90,00	5	3	60,00	15	12	80,00
Atributos diferentes	233	124	53,22	98	25	25,51	331	149	45,02
Total	**851**	**694**	**81,55**	**162**	**59**	**36,42**	**1011**	**753**	**74,48**

Com base nos valores da Tabela 32 é possível verificar que, para quase todas as classes (com exceção das entradas com atributos diferentes), a maioria das entradas foi classificada como válida. As entradas com atributos fonte e alvo diferentes poderiam ser resolvidas com a remoção desses atributos tornando, assim, as entradas mais gerais. Cerca de 82% das entradas de palavras e 36% das entradas de multipalavras são válidas. A porcentagem menor de entradas válidas para multipalavras é decorrência da alta taxa de erro no alinhamento lexical nessa categoria (AER = 15,71%). Um exemplo de uma entrada não-válida gerada como decorrência de um erro de alinhamento é a que estabelece a correspondência entre a palavra em pt *jantar* e a expressão multipalavra, em en, *dinner already.*

Assim, considerando-se que 79,86% das 1.487 entradas iguais e 73,42% das 8.820 entradas diferentes, de acordo com as porcentagens levantadas na avaliação manual, são válidas; tem-se um conjunto de 10.307 entradas com uma precisão estimada em 74%. Além disso, o conjunto resultantes da união das entradas bilíngües nas classes de melhor desempenho – categoria e atributo igual (1.487), atributos mais específicos (5.329) e atributos mais gerais (172) – possui 6.988 entradas e uma precisão estimada em 88%. Por fim, considerando-se apenas as entradas de palavras para essas melhores classes o conjunto resultante com 6.338 entradas possui uma precisão estimada em 92%.

6.2 Avaliação das regras de tradução no ReTraTos

As regras de tradução induzidas automaticamente para os pares pt–es e pt–en foram avaliadas direta e indiretamente, de modo automático, em um *corpus* de teste com 649 sentenças paralelas nos três idiomas: pt, es e en. Esse *corpus* pertence ao mesmo gênero dos *corpora*

usados na indução, já que está composto por sentenças provenientes de textos de edições da revista *Pesquisa FAPESP* não usados na indução.

Tanto na avaliação direta quanto na avaliação indireta realizadas de modo automático, considerou-se como referência as contra-partes do *corpus* de teste. Por exemplo, o conjunto de referência usado na avaliação da tradução das sentenças de `pt` para `es` foi o conjunto original de sentenças em `es` derivado da revista *Pesquisa FAPESP*. O mesmo se aplica às demais combinações dos pares `pt`–`es` e `pt`–`en`.

As sentenças do *corpus* de teste foram traduzidas utilizando-se os recursos produzidos no ReTraTos: (1) o módulo de tradução apresentado no Capítulo 5, seção 5.4; (2) as regras de tradução induzidas como apresentado no Capítulo 5, seção 5.3 e (3) os léxicos bilíngües induzidos como descrito no Capítulo 5, seção 5.2. Além disso, as etapas de pré-processamento e geração que antecedem e sucedem, respectivamente, a tradução por meio do módulo implementado no ReTraTos (veja Figura 11 do Capítulo 2) foram desempenhadas pelos analisadores e geradores, nesta ordem, compilados pelas ferramentas distribuídas com o sistema de TA `Apertium` e a partir dos dicionários morfológicos construídos no ReTraTos (veja seção 4.2 do Capítulo 4).

Nesses experimentos foram avaliados seis conjuntos de regras de tradução induzidas automaticamente no ReTraTos com os parâmetros especificados na Tabela 33, na qual pi e pf são as porcentagens usadas para cálculo das freqüências mínimas nas etapas de identificação de padrões e filtragem das regras, respectivamente (veja seção 5.3 no Capítulo 5). Essas configurações foram projetadas para testar o impacto das porcentagens no filtro e da especificação de PoS na indução.

Tabela 33: Configurações avaliadas na indução das regras de tradução no ReTraTos

	PoS incluídas	***pi***	***pf***
1	todas	0,0007	0,50
2	`pr`, `det`, `pr+det`	0,0007	0,50
3	todas	0,0015	0,50
4	`pr`, `det`, `pr+det`	0,0015	0,50
5	todas	0,0030	0,50
6	`pr`, `det`, `pr+det`	0,0030	0,50

As Tabelas 34 e 35 trazem as quantidades de regras induzidas, em cada uma dessas configurações, para cada tipo de alinhamento e par de línguas, nos dois sentidos da tradução. Vale lembrar que as configurações ímpares foram usadas para induzir regras para todas as PoS e as pares, apenas para `pr`, `det`, `pr+det`. Além disso, quanto menor o valor de pi, menor

o limite mínimo para a freqüência de um padrão usado na etapa de identificação de padrões e maior a quantidade de regras induzidas.

Tabela 34: Quantidade de regras induzidas, por tipo de alinhamento, nas configurações ímpares

	Configuração 1				**Configuração 3**				**Configuração 5**			
	T0	T1	T2	TODAS	T0	T1	T2	TODAS	T0	T1	T2	TODAS
pt→es	46	1374	1	1421	18	747	1	766	10	430	0	440
es→pt	92	1237	0	1329	46	679	0	722	18	400	0	418
pt→en	129	488	30	647	60	264	19	343	32	148	10	190
en→pt	233	457	32	722	118	237	18	373	76	126	8	210

Tabela 35: Quantidade de regras induzidas, por tipo de alinhamento, nas configurações pares

	Configuração 2				**Configuração 4**				**Configuração 6**			
	T0	T1	T2	TODAS	T0	T1	T2	TODAS	T0	T1	T2	TODAS
pt→es	54	1583	0	1637	17	871	0	888	7	404	0	411
es→pt	109	1521	0	1630	41	855	0	896	10	417	0	427
pt→en	165	480	18	663	80	276	5	361	43	124	3	170
en→pt	365	546	14	925	195	262	6	463	104	119	4	227

Os detalhes da avaliação dessas configurações em cada uma das metodologias de avaliação, bem como os resultados obtidos, são apresentados nas seções a seguir: (6.2.1) avaliação direta automática e (6.2.2) avaliação indireta automática.

6.2.1 Avaliação direta automática das regras de tradução

Na avaliação direta automática, para cada regra, foram calculados os valores de precisão (equação 2.14), cobertura (equação 2.15) e medida-F (equação 2.16) cujas fórmulas são apresentadas no Capítulo 2. Para tanto, as sentenças traduzidas utilizando as regras induzidas foram comparadas com as sentenças de referência analisando-se apenas o trecho traduzido por cada regra. Nestes cálculos considerou-se como *candidatos* a quantidade de *tokens* no trecho traduzido pela regra e como *referência* a quantidade de *tokens* no trecho equivalente na sentença de referência, desconsiderando-se a ordem de ocorrência dos itens.

A avaliação direta de todas as regras induzidas em todas as configurações é uma meta difícil de ser alcançada já que, para tanto, seria necessário um *corpus* de teste tão grande quanto (ou maior que) o *corpus* usado na indução. Considerando-se tal limitação, realizou-se um levantamento da quantidade de regras induzidas que foram aplicadas na tradução do

corpus de teste com o intuito de restringir a avaliação das regras às aplicadas. O resultado deste levantamento é apresentado nas Tabelas 36 e 37 referentes aos dados das Tabelas 34 e 35, respectivamente. O caractere "–" presente nessas tabelas indica que nenhuma regra foi induzida para um determinado tipo em uma dada configuração e, portanto, nenhuma regra desse tipo nessa configuração poderia ter sido aplicada.

Tabela 36: Quantidade de regras aplicadas na tradução do *corpus* de teste, por tipo de alinhamento, nas configurações ímpares

	Configuração 1				**Configuração 3**				**Configuração 5**			
	T0	T1	T2	TODAS	T0	T1	T2	TODAS	T0	T1	T2	TODAS
pt→es	5	736	0	741	4	543	0	547	1	371	–	372
es→pt	11	564	–	575	6	416	–	422	4	278	–	282
pt→en	27	208	12	247	20	156	9	185	11	103	6	120
en→pt	20	170	9	199	16	120	7	143	17	81	4	102

Tabela 37: Quantidade de regras aplicadas na tradução do *corpus* de teste, por tipo de alinhamento, nas configurações pares

	Configuração 2				**Configuração 4**				**Configuração 6**			
	T0	T1	T2	TODAS	T0	T1	T2	TODAS	T0	T1	T2	TODAS
pt→es	6	742	–	748	4	528	–	532	1	334	–	335
es→pt	22	619	–	641	13	433	–	446	4	263	–	267
pt→en	39	224	5	268	24	157	1	182	15	92	1	108
en→pt	57	197	2	256	37	125	0	162	23	75	2	100

Como resultado desta primeira avaliação foi possível determinar quais das regras induzidas não tiveram êxito, ou seja, quais foram aplicadas ao conjunto de teste com valores de precisão ou cobertura iguais a 0. As quantidades das regras sem êxito, para cada tipo de alinhamento, nas configurações ímpares e pares são apresentadas, respectivamente, nas Tabelas 38 e 39. Essas quantidades são acompanhadas de suas porcentagens calculadas como a quantidade de regras sem êxito dividida pela quantidade de regras aplicadas.

De acordo com os valores das Tabelas 38 e 39, a porcentagem de regras com baixo desempenho é inversamente proporcional ao limite de freqüência mínima usado na identificação de padrões (valor de *pi*), ou seja, quanto menor este limite, maior o número de regras induzidas e estas novas regras estão mais propensas a gerarem traduções erradas. O grande desafio, então, é estabelecer um valor para *pi* que gere um número razoável de regras – para garantir uma boa cobertura – com um desempenho satisfatório – para garantir uma boa precisão. Neste sentido, talvez as configurações com valores medianos para *pi* – 3 e 4 – sejam

Tabela 38: Quantidade e porcentagem de regras sem êxito, por tipo de alinhamento, nas configurações ímpares

	Configuração 1			**Configuração 3**			**Configuração 5**		
	T0	T1	T2	T0	T1	T2	T0	T1	T2
pt→es	1 (20,00%)	21 (2,85%)	–	0	12 (2,21%)	–	0	8 (2,16%)	–
es→pt	2 (18,18%)	18 (3,19%)	–	1 (16,67%)	4 (0,96%)	–	0	2 (0,72%)	–
pt→en	0	13 (6,25%)	0	0	5 (3,21%)	0	0	2 (1,94%)	0
en→pt	2 (10,00%)	13 (7,65%)	0	2 (12,50%)	8 (6,67%)	0	1 (5,88%)	1 (1,23%)	0

Tabela 39: Quantidade e porcentagem de regras sem êxito, por tipo de alinhamento, nas configurações pares

	Configuração 2			**Configuração 4**			**Configuração 6**		
	T0	T1	T2	T0	T1	T2	T0	T1	T2
pt→es	0	16 (2,16%)	–	0	8 (1,52%)	–	0	3 (0,90%)	–
es→pt	2 (9,09%)	19 (3,07%)	–	0	4 (0,92%)	–	0	2 (0,76%)	–
pt→en	4 (10,26%)	13 (5,80%)	0	1 (4,17%)	10 (6,37%)	0	1 (6,67%)	4 (4,35%)	0
en→pt	10 (17,54%)	16 (8,12%)	0	4 (10,81%)	2 (1,60%)	0	3 (13,04%)	2 (2,67%)	0

as de melhor desempenho na avaliação indireta das regras induzidas. Para verificar esta hipótese, as sentenças alvo geradas pelo módulo de tradução utilizando as regras induzidas em cada uma das configurações apresentadas anteriormente, foram avaliadas indiretamente como apresentado na próxima seção.

6.2.2 Avaliação indireta automática das regras de tradução

Na avaliação indireta automática, as sentenças alvo geradas com base nas regras de tradução foram comparadas às sentenças de referência e 5 medidas foram calculadas automaticamente: BLEU (PAPINENI et al., 2002) (equação 2.8), NIST (DODDINGTON, 2002) (equação 2.11) e as três já calculadas anteriormente apenas para o trecho traduzido por cada regra – precisão (P), cobertura (C) e medida-F (F) (MELAMED et al., 2003) – só que, desta vez, calculadas para toda a sentença.

As duas primeiras – BLEU e NIST – levam em consideração os n-gramas em comum nas sentenças candidatas (traduzidas automaticamente) e de referência (traduzidas por humano), enquanto as outras três medidas são calculadas, neste trabalho, sem levar em consideração a ordem das palavras. Assim, BLEU e NIST apontam como a melhor tradução aquela similar à referência em tamanho, escolha e ordem das palavras enquanto precisão, cobertura e medida-F – do modo como foram calculadas neste trabalho – verificam apenas

a escolha das palavras. Os valores de BLEU, precisão, cobertura e medida-F variam de 0 a 1 enquanto NIST possui valores maiores do que 0, mas sem um limite superior.

Além da avaliação das traduções geradas pelos recursos construídos no ReTraTos (módulo de tradução, regras e léxicos bilíngües), outros sistemas de TA disponíveis para os idiomas sob estudo também foram analisados: dois sistemas para o par `pt`–`es` – `Apertium`[4] (`AP`) e seu protótipo na *web* `Apertium-P`[5] (`AP-P`) – e três sistemas disponíveis *online* para o par `pt`–`en` – `FreeTranslation`[6] (`FT`), `BabelFish`[7] (`BF`) e `Google`[8] (`GO`). A tradução palavra-a-palavra – obtida por meio do módulo de tradução implementado no ReTraTos – também foi avaliada em todos os sentidos da tradução e é indicada nas tabelas a seguir sob a denominação `PA`.

Quanto aos sistemas utilizados na comparação, sabe-se que `BabelFish` e `Google` têm por trás o sistema `Systran` (SENELLART & SENELLART, 2005) e as diferenças em seus desempenhos provavelmente decorrem de uma maior cobertura do léxico bilíngüe de um sistema em relação ao outro. Além disso, pretendia-se, também, avaliar a tradução para o par `pt`–`es` utilizando os sistemas `AutomaticTrans`[9] e `Universia`[10], porém tal avaliação foi impossibilitada devido à limitação do primeiro sistema de traduzir, no máximo, 50 palavras e ao estado temporário do segundo que se encontra fora do ar (10/04/2007).

Os valores para as cinco medidas são apresentados nas Tabelas 40 (`pt`–`es`) e 41 (`pt`–`en`) para as configurações de ReTraTos – representadas pelos números atribuídos a elas na Tabela 33 – e os sistemas em comparação.

Com relação aos valores da Tabela 40 é possível notar que todos os sistemas avaliados apresentaram valores similares na tradução `pt`→`es`, porém as configurações com as regras induzidas foram ligeiramente melhores, destacando-se as configurações 1 e 3. Embora os sistemas `AP` e `AP-P` tenham apresentado valores de precisão, cobertura e medida-F maiores do que os obtidos nas configurações de ReTraTos esse fato representa, apenas, que mais palavras em comum com a referência foram retornadas por tais sistemas, mas não que estas palavras foram usadas na ordem adequada uma vez que os valores de BLEU e NIST são menores.

[4] O sistema de TA de código aberto `Apertium` está disponível em `http://apertium.org` juntamente com o pacote de dados língüísticos `es`–`pt`. A versão do sistema utilizada nesta avaliação foi a 2.0 e a versão dos dados lingüísticos, a 0.9.1.

[5] `http://xixona.dlsi.ua.es/prototype/`.

[6] `www.freetranslation.com`.

[7] `http://babelfish.altavista.com/`.

[8] `http://www.google.com.br/language_tools`.

[9] `www.automatictrans.es`.

[10] `www.universia.com.br/tradutor`.

Tabela 40: Avaliação indireta das regras induzidas no ReTraTos para o par pt–es e o desempenho de outros sistemas de TA

	pt→es					es→pt				
Sistema	**BLEU**	**NIST**	**P**	**C**	**F**	**BLEU**	**NIST**	**P**	**C**	**F**
1	0,6513	10,8516	0,7991	0,7944	0,7968	0,6666	10,9756	0,8003	0,8068	0,8035
2	0,6511	10,8440	0,7986	0,7939	0,7962	0,6655	10,9695	0,8004	0,8061	0,8033
3	0,6514	10,8510	0,7991	0,7945	0,7968	0,6660	10,9719	0,8004	0,8068	0,8036
4	0,6507	10,8392	0,7981	0,7935	0,7958	0,6654	10,9671	0,8002	0,8062	0,8032
5	0,6510	10,8502	0,7992	0,7945	0,7968	0,6657	10,9754	0,8006	0,8068	0,8037
6	0,6506	10,8352	0,7979	0,7937	0,7958	0,6650	10,9599	0,7997	0,8061	0,8029
PA	0,6490	10,8188	0,7971	0,7932	0,7952	0,6649	10,9503	0,7991	0,8074	0,8033
AP	0,6382	10,6379	0,8080	0,7964	0,8021	0,6098	10,3057	0,7714	0,7853	0,7783
AP-P	0,6387	10,6438	0,8082	0,7966	0,8024	0,6288	10,5073	0,7841	0,7969	0,7904

A avaliação das traduções no sentido es→pt também apresentou valores bastante similares principalmente para as configurações de ReTraTos e a tradução palavra-a-palavra. O fato deste grupo de melhores configurações englobar a tradução palavra-a-palavra e estar relativamente distante dos demais sistemas é um indício de que o bom desempenho se deve, em grande parte, à maior cobertura do léxico bilíngüe usado nessas configurações e não às regras de tradução propriamente ditas. Novamente, a configuração de ReTraTos de melhor desempenho na tradução es→pt foi a de número 1 acompanhada de perto pelas configurações 3 e 5.

Tabela 41: Avaliação indireta das regras induzidas no ReTraTos para o par pt–en e o desempenho de outros sistemas de TA

	pt→en					en→pt				
Sistema	**BLEU**	**NIST**	**P**	**C**	**F**	**BLEU**	**NIST**	**P**	**C**	**F**
1	0,2832	7,0869	0,6132	0,5986	0,6058	0,2400	6,1133	0,4707	0,4942	0,4822
2	0,2646	7,0327	0,6133	0,5870	0,5999	0,2439	6,2318	0,4799	0,4975	0,4885
3	0,2852	7,1049	0,6129	0,5979	0,6053	0,2401	6,1149	0,4706	0,4937	0,4819
4	0,2636	7,0251	0,6125	0,5864	0,5991	0,2421	6,2266	0,4794	0,4955	0,4873
5	0,2628	7,0902	0,6182	0,5842	0,6008	0,2377	6,1511	0,4749	0,4911	0,4829
6	0,2551	6,9943	0,6118	0,5824	0,5968	0,2428	6,2393	0,4807	0,4956	0,4880
PA	0,2606	6,7712	0,5964	0,5885	0,5924	0,2324	6,0173	0,4640	0,4973	0,4800
FT	0,3294	7,6509	0,6670	0,6586	0,6628	0,3053	6,8454	0,5367	0,5846	0,5596
BF	0,3161	7,4648	0,6517	0,6438	0,6477	0,3666	7,6799	0,6064	0,6419	0,6237
GO	0,3295	7,6112	0,6609	0,6470	0,6539	0,3121	6,8767	0,5379	0,5805	0,5584

Na avaliação do par pt–en as configurações de ReTraTos não apresentaram desempenhos tão bons quanto na tradução pt–es. Este fato já era esperado considerando-se que o sistema de indução de regras não foi projetado para lidar com mudanças complexas na estrutura da tradução (freqüentes em línguas mais distantes como pt e en), mas apenas tratar casos locais de mudanças de ordem, ausências ou inserções de itens. De acordo com os valores da Tabela 41, as melhores configurações de ReTraTos no sentido pt→en

foram as de números 3 e 1, as quais apresentam uma melhora considerável em relação à tradução palavra-a-palavra, porém com desempenhos ainda distantes dos melhores sistemas – `FreeTranslation` e `Google`.

Por fim, a avaliação das traduções no sentido `en`→`pt` apontou como melhor sistema o `BabelFish` o qual parece ter recursos que lidam adequadamente com o idioma alvo (`pt`). Nesse sentido, todas as configurações de ReTraTos apresentaram praticamente os mesmos resultados e foram ligeiramente melhores do que a tradução palavra-a-palavra, destacando-se a configuração de número 2. Esse pior desempenho de ReTraTos no sentido `en`→`pt` pode ser justificado, em parte, por erros decorrentes do mapeamento de palavras sem valores de atributos (gênero, número, tempo verbal etc.), no inglês, para palavras em português. Nesta transferência, muitas vezes, não é possível determinar o valor do atributo da palavra em `pt` e, por isso, a forma base é mantida no lugar da forma flexionada.

Assim, com relação aos parâmetros testados nos experimentos apresentados neste documento – limite mínimo de freqüência na identificação de padrões e especificação de PoS – pode-se concluir que: (1) as configurações com limites de freqüência mediano ($pi = 0,0015$) ou baixo ($pi = 0,0007$) foram melhores e (2) a especificação de valores de PoS só se mostrou útil na tradução `en`→`pt`. Quanto ao primeiro aspecto sob estudo, sabe-se que, quanto menor a porcentagem do filtro para identificação de padrões, menor o limite de freqüência mínima usado na identificação e, portanto, maior o número de regras induzidas. Além disso, também constatou-se que o aumento no número de regras acarreta um aumento na probabilidade de ocorrência de erros de aplicadação das mesmas – como apresentado nas Tabelas 38 e 39 – e talvez por este motivo as configurações com melhores desempenho tenham sido as de porcentagem de freqüência mediana.

Com relação à especificação de valores para PoS (configurações pares), a indução foi limitada a regras contendo as PoS consideradas mais problemáticas na tradução dos pares de língua sob estudo – conforme estudo apresentado na seção 1.1 do Capítulo 1 – ou seja, preposições (`pr`), determinantes (`det`) e a contração dessas duas PoS (`pr+det`). Embora a estratégia de indução de regras para PoS específicas tenha apresentado bons resultados, ficou claro que ela não é suficiente para garantir a tradução de uma sentença completa. Assim, uma proposta de experimento futuro é a de induzir, separadamente, conjuntos para PoS específicas e, em seguida, unir os conjuntos gerados em apenas um.

A Tabela 43 apresenta exemplos de sentenças traduzidas pelo módulo de tradução implementado no ReTraTos com os léxicos bilíngües e as regras de tradução induzidos neste projeto. As traduções foram obtidas utilizando-se as regras da configuração 3 nos três

primeiros sentidos e as da configuração 2, no último; e considerando-se como sentenças originais (de referência) as apresentadas na Tabela 42.

Nos exemplos da Tabela 43, os trechos traduzidos pelas regras são apresentados entre colchetes e vêm acompanhados do número que identifica unicamente cada regra, em cada configuração. Por exemplo, na tradução `pt→es` uma regra do tipo 0, a 190, foi aplicada para remover o artigo que precede o nome próprio *Japón*. Vale ressaltar que o módulo TA (aplicação das regras) está ainda em desenvolvimento e a versão preliminar, usada nos experimentos apresentados neste documento, não explora por completo todas as informações presentes nas regras de tradução.

Tabela 42: Exemplos de sentenças originais (de referência) do *corpus* de teste

`pt`	Estatísticas mostram que o Japão é o país com menor índice de mortes por doenças do coração , enquanto os Estados Unidos têm um dos índices mais altos .
`es`	Las estadísticas muestran que Japón es el país con el menor índice de muertes por enfermedades del corazón , mientras que Estados Unidos tiene uno de los índices más altos .
`en`	Statistics show that Japan is the country with the lowest indices of deaths from heart disease , whilst the United States has one of the highest indices .

Tabela 43: Exemplos de sentenças traduzidas por meio dos recursos induzidos no ReTraTos

`pt→es`	[Estadísticas muestran que]417 [Japón]190 [es el país con]744 [menor índice de muertes por]28 [enfermedades del corazón]331 , [mientras que los]103 [Estados Unidos tienen]385 uno [de los índice]466 más altos .
`es→pt`	As [estatísticas mostram que]413 Japão [é o país com]711 o [menor índice de mortes por]29 [doenças do coração]322 , enquanto [Estados Unidos tem]372 um dos taxas mais altas .
`pt→en`	Statistics show [that the]55 Japan [is the]333 [country with lower level]141 of [deaths by diseases]155 of the heart , [while the]30 United States have one of the [more high levels]112 .
`en→pt`	Estatísticas apresentar que Japão [é o país]442 [com o]460 menor index [de mortes]225 from coração doença , whilst [Estados Unidos]341 tem um de [altos]296 index .

7 Conclusões e trabalhos futuros

Este documento apresenta o projeto ReTraTos, que pode ser visto como uma alternativa para o processo árduo de construção de tradutores uma vez que propõe a indução, de modo completamente automático, de recursos úteis para a tradução automática (TA) – regras de tradução e léxicos bilíngües – a partir de *corpora* paralelos alinhados, empregando métodos empíricos para minimizar os custos de desenvolvimento.

Além da minimização do esforço na construção de tradutores automáticos, outros fatos que motivaram este projeto são: o avanço nos estudos da TA para o português (`pt`), o baixo desempenho dos sistemas de TA disponíveis para o `pt` e a utilização da abordagem de Aprendizado de Máquina juntamente com *Example Based Machine Translation* como uma tentativa de superar os problemas encontrados nos sistemas de TA atuais.

Os sistemas para a indução automática de léxicos bilíngües e de regras de tradução implementados no ReTraTos, como apresentado no Capítulo 5 (seções 5.2 e 5.3, respectivamente), foram avaliados na tradução de sentenças em `pt` de e para outros dois idiomas: espanhol (`es`) e inglês (`en`).

Dois léxicos bilíngües foram gerados pelo sistema implementado no projeto ReTraTos: um `es`–`pt` e outro `pt`–`en`. Os valores obtidos na avaliação intrínseca desses léxicos bilíngües – 81%–84% de precisão nas entradas `pt`–`es` e 74%–88% de precisão nas entradas `pt`–`en` – estão de acordo com os melhores valores obtidos por outros métodos avaliados seguindo a mesma metodologia: 73% e 86% de precisão nas entradas inglês–chinês em (FUNG, 1995) e (WU & XIA, 1994), respectivamente, 55%–56% de precisão nas entradas francês–inglês (RESNIK & MELAMED, 1997) e 43%–58% de precisão nas entradas inglês–sérvio (SCHAFER & YAROWSKY, 2002).

O valor obtido para a medida de concordância entre juízes, kappa, na avaliação do léxico bilíngüe `pt`–`es`, 0,63, também está de acordo com os valores relatados em (RESNIK & MELAMED, 1997): entre 0,55 e 0,74. Porém, o valor de kappa na avaliação do léxico `pt`–`en` foi um pouco menor, 0,48 quando palavras e multipalavras foram consideradas e de

0,52 quando apenas palavras foram consideradas. Contudo, notou-se que a maior parte das discordâncias entre os juízes deste último par envolvia as classes válida e parcialmente válida (com sugestão de alguma alteração no valor de um atributo ou no sentido da tradução) o que torna essa discordância menos problemática. Além disso, como apontado por Craggs & Wood (2005), é impossível estabelecer limites únicos para a kappa com base nos quais todas as codificações podem ser julgadas. Segundo esses autores, cada um deve decidir, com base no uso pretendido para o esquema de codificação, se os níveis de concordância observados são suficientes e, assim, realizar a análise dos resultados.

Outra constatação obtida com a avaliação dos léxicos bilíngües no ReTraTos está relacionada às entradas com erros de etiquetação morfossintática. Assim como Fung (1995) e Resnik & Melamed (1997), também verificou-se que a maioria das entradas classificadas como parcialmente válidas no ReTraTos – em média de 25% para `pt–es` e 12% para `pt–en` – seria válida se não fosse por um erro de etiquetação.

A partir da avaliação exaustiva dos léxicos bilíngües induzidos no ReTraTos e das constatações obtidas como resultado, várias propostas de trabalhos surgem como conseqüência natural. Uma delas é a de aplicar filtros para selecionar as melhores entradas, como o filtro de significância de Wu & Xia (1994) que se baseia em critérios que, simultameamente, penalizam os dados esparsos e consideram as palavras ambíguas.

Outra proposta de trabalho futuro é alterar o método de indução de léxicos para que este generalize as entradas bilíngües removendo todos os valores de seus atributos. Esse processo de generalização de entradas por meio da remoção de seus atributos é uma alternativa para aumentar a cobertura do léxico induzido. Tal generalização foi realizada, apenas para entradas de verbos, no momento da utilização dos léxicos bilíngües na tradução das sentenças do *corpus* de teste (durante a avaliação indireta das regras de tradução) por meio do módulo de tradução automática implementado no ReTraTos.

Quanto ao sistema de indução de regras de tradução, o mesmo foi implementado seguindo uma abordagem não encontrada, até o momento, em nenhum outro método de indução de regras: a indução baseada em blocos de alinhamento. Nesta abordagem, os exemplos de tradução são divididos em blocos de alinhamentos de acordo com o tipo de alinhamento lexical estabelecido entre os itens fonte e alvo (seção 5.3.1) – tipo 0 (alinhamento de omissão), tipo 1 (alinhamento que preserva a ordem dos itens) e tipo 2 (alinhamento com mudança de ordem) – e a identificação dos padrões (primeira etapa do processo de indução) é realizada, separadamente, para cada um desses tipos de blocos.

Seguindo-se esta abordagem, limites mínimos de freqüência diferentes são aplicados em cada tipo de bloco uma vez que estes limites são determinados como uma porcentagem do número total de seqüências nos blocos. Deste modo, padrões de tipo 0 e 2 – menos freqüentes do que padrões de tipo 1– são identificados com limites de freqüência menores do que os usados na identificação de padrões do tipo 1. A vantagem de se realizar a identificação de padrões para cada tipo de bloco de alinhamento separadamente é permitir que padrões relevantes de tipos menos freqüentes também sejam identificados.

A estratégia de identificação de padrões fonte adotada no ReTraTos (seção 5.3.2.1) está fortemente baseada na técnica de *Sequential Pattern Mining* (AGRAWAL & SRIKANT, 1995) e no algoritmo `PrefixSpan` (PEI et al., 2004), o qual propõe a divisão do conjunto de seqüências por prefixo freqüente aumentando-se passo-a-passo essas seqüências priorizando-se a profundidade. Embora utilize a mesma idéia de `PrefixSpan`, a identificação de padrões fonte, no ReTraTos, difere deste algoritmo principalmente no modo como os padrões são buscados: no ReTraTos, permite-se que um padrão ocorra mais de uma vez numa dada seqüência, enquanto que, no `PrefixSpan`, um padrão pode ocorrer no máximo uma vez em cada uma das seqüências de um conjunto de seqüências. Essa diferença é de fundamental importância no contexto do projeto ReTraTos, uma vez que as seqüências são compostas por PoS e itens lematizados, os quais ocorrem, com freqüência, repetidas vezes na mesma seqüência.

A abordagem de filtragem das regras de tradução adotada no ReTraTos também é um diferencial em relação aos outros métodos de indução uma vez que estes, geralmente, realizam apenas o filtro por freqüência (ou por outra medida estatística, como um peso atribuído a cada regra) mantendo-se apenas a opção alvo de maior valor. No ReTraTos, uma filtragem mais elaborada das regras induzidas é realizada com base nos valores lexicais e de atributos morfológicos, e o filtro por freqüência é utilizado apenas quando todos os outros filtros falham. Neste caso, a saída do sistema é sempre uma gramática não-ambígua.

O método de indução de regras de tradução desenvolvido no ReTraTos implementa muitas das características apontadas em (PROBST, 2005) como trabalhos futuros. Primeiro, no ReTraTos, as regras são aprendidas de uma maneira completamente automática. Além disso, no ReTraTos, os exemplos de tradução são divididos automaticamente em trechos menores (blocos de alinhamentos) e níveis de abstração diferentes (valores lexicais e de atributos morfológicos) são utilizados para filtrar as regras.

Com relação à avaliação das regras de tradução induzidas no ReTraTos para os dois pares de idiomas sob estudo – `pt`–`es` e `pt`–`en` – nos quatro sentidos de tradução – `pt`→`es`,

es→pt, pt→en e en→pt – cabem, aqui, algumas considerações. Para o primeiro par, pt–es, o desempenho da tradução com as regras induzidas no ReTraTos foi melhor do que o dos sistemas em comparação (seção 6.2.2), o que se deve, em grande parte, à maior cobertura dos léxicos bilíngües induzidos no ReTraTos quando comparada à cobertura dos léxicos dos outros sistemas, o que pode ser comprovado pelo bom desempenho da tradução palavra-a-palavra.

Para o segundo par, pt–en, como já era esperado, a tradução por meio das regras induzidas no ReTraTos não foi melhor do que os sistemas em comparação, uma vez que estes estão preparados para tratar mudanças (sintáticas, por exemplo) na estrutura da tradução ainda não consideradas na primeira versão do sistema implementado no ReTraTos. Contudo, o desempenho da tradução com as regras induzidas foi consideravelmente melhor do que a tradução palavra-a-palavra no sentido pt→en, provando que, mesmo com limitações, o método proposto traz um ganho para a tradução. Quanto à tradução no sentido en→pt, vale ressaltar que esta foi bastante prejudicada pela não atribuição de valores para atributos das palavras alvo (pt) quando definidos com base nos valores da palavra fonte (en), os quais nem sempre estão presentes.

Uma proposta de trabalho futuro que surge naturalmente a partir dos valores levantados nos primeiros experimentos é a de estender o método de indução de regras de tradução para lidar com mudanças na estrutura das sentenças na tradução entre línguas mais distantes, como o pt e o en (na qual um tratamento sintático parece ser essencial). Neste sentido, pretende-se utilizar *parsers*, como o PALAVRAS (BICK, 2000), em uma próxima versão do indutor de regras. Além dessa, outra alteração que auxiliaria o tratamento de línguas mais distantes é a adaptação do método para lidar com casos nos quais a língua fonte não possui valores de atributos que precisam ser determinados adequadamente na língua alvo, como ocorreu na tradução en→pt.

É importante dizer que mesmo utilizando uma versão preliminar do módulo de TA implementado para a aplicação dos recursos induzidos, em todos os experimentos, a tradução com base nas regras induzidas apresentou melhor desempenho, em menor ou maior grau, que a tradução palavra-a-palavra. Esse fato demonstra a funcionalidade das regras de tradução e a relevância da abordagem proposta uma vez que esta conseguiu solucionar alguns dos problemas da TA envolvendo o pt levantados no início deste projeto como, por exemplo, a remoção do determinante antes de nomes próprios na tradução pt→es.

Uma comparação dos valores obtidos na avaliação indireta das regras induzidas no ReTraTos com os relatados na literatura para outros métodos de indução de regras de tradução,

infelizmente, não é possível uma vez que nenhum dos métodos estudados aborda a indução de regras entre os pares `pt`–`es` e `pt`–`en`, e a maioria deles estuda idiomas muito diferentes, como nos pares coreano–inglês (LAVOIE et al., 2002) ou inglês–turco (ÖZ & CICEKLI, 1998). Dos métodos pesquisados, o que mais se aproxima da metodologia de avaliação e dos idiomas estudados no ReTraTos é o de (MEYERS et al., 1998), o qual induz regras `es`–`en` a partir de exemplos de tradução analisados sintaticamente. A tradução por meio das regras induzidas por esse método, após a seleção da melhor regra a ser aplicada em um determinado momento (MEYERS et al., 2000), foi avaliada por meio de uma medida semelhante à medida-F utilizada no ReTraTos, resultando em valores de 62,6% a 70,9%. Esses valores são maiores do que os obtidos para as regras `pt`–`en` (de 59,7% a 60,6%) e, mesmo considerando-se as diferenças nas línguas fonte (`es` e `pt`), esses valores indicam, mais uma vez, que um tratamento mais profundo (sintático) é essencial na indução de regras para línguas distantes.

O projeto ReTraTos, apresentado nesta tese, trouxe várias contribuições para a área de PLN com a produção de recursos lingüístico–computacionais e, principalmente, com as abordagens inovadoras para a indução e a filtragem das regras de tradução.

Entre os recursos lingüístico-computacionais obtidos neste projeto estão: (1) *corpora* de textos paralelos `pt`–`en` e `pt`–`es` alinhados sentencial e lexicalmente e etiquetados morfossintaticamente, bem como suas versões intermediárias; (2) ferramentas computacionais construídas ou adaptadas para pré-processar os *corpora* paralelos (alinhadores, etiquetadores etc.); (3) regras de tradução e léxicos bilíngües para os pares `pt`–`en` e `pt`–`es` no domínio dos *corpora* utilizados na extração; (4) sistemas capazes de induzir novas regras e novos léxicos a partir de novos *corpora* paralelos; (5) um sistema simples de TA baseado na recombinação das regras de tradução e na consulta aos léxicos bilíngües; e (6) diversos documentos relatando as etapas do projeto e os resultados obtidos. Vale ressaltar que esses recursos lingüístico-computacionais têm grande valor na área de PLN e poderão ser usados em outros projetos desenvolvidos no NILC e em outros grupos de pesquisa, impulsionando, ainda mais, a pesquisa nessa área.

Em vista de todo o conteúdo apresentado, pode-se afirmar que a utilização de técnicas de AM e EBMT para o objetivo proposto – indução de recursos lingüísticos úteis para a TA – se mostrou bastante adequada.

Referências

AGRAWAL, R.; SRIKANT, R. Mining sequential patterns. In: YU, P. S.; CHEN, A. S. P. (Ed.). *Eleventh International Conference on Data Engineering*. Taipei, Taiwan: IEEE Computer Society Press, 1995. p. 3–14.

ARMENTANO-OLLER, C.; CARRASCO, R. C.; CORBÍ-BELLOT, A. M.; FORCADA, M. L.; GINESTÍ-ROSELL, M.; ORTIZ-ROJAS, S.; PÉREZ-ORTIZ, J. A.; RAMÍREZ-SÁNCHEZ, G.; SÁNCHEZ-MARTÍNEZ, F.; SCALCO, M. A. Open-source Portuguese-Spanish machine translation. In: *Proceedings of the VII Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada*. Itatiaia-RJ, Brazil: [s.n.], 2006. p. 50–59.

BICK, E. *The parsing system Palavras, automatic grammatical analysis of Portuguese in a constraint grammar framework*. 503 p. Tese (Doutorado), December 2000.

BROWN, P.; DELLA-PIETRA, V.; DELLA-PIETRA, S.; MERCER, R. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, v. 19, n. 2, p. 263–312, 1993.

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

BROWN, R. D. Adding linguistic knowledge to a lexical example-based translation system. In: *Proceedings of the 8th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-99)*. Chester, England: [s.n.], 1999. p. 22–32.

BROWN, R. D. Transfer-rule induction for example-based translation. In: *Proceedings of the MT Summit VIII Workshop on Example-Based Machine Translation*. Santiago de Compostela, Spain: [s.n.], 2001. p. 1–11.

CANALS-MAROTE, R.; ESTEVE-GUILLÉN, A.; GARRIDO-ALENDA, A.; GUARDIOLA-SAVALL, M.; ITURRASPE-BELLVER, A.; MONTSERRAT-BUENDIA, S.; ORTIZ-ROJAS, S.; PASTOR-PINA, H.; PÉREZ-ANTÓN, P.; FORCADA, M. The Spanish-Catalan machine translation system interNOSTRUM. In: *Proceedings of MT Summit VIII: Machine Translation in the Information Age*. Santiago de Compostela, Spain: [s.n.], 2001. p. 73–76.

CARBONELL, J.; PROBST, K.; PETERSON, E.; MONSON, C.; LAVIE, A.; BROWN, R.; LEVIN, L. Automatic rule learning for resource-limited MT. In: *AMTA '02: Proceedings of the 5th Conference of the Association for Machine Translation in the Americas on Machine Translation: From Research to Real Users. Lecture Notes In Computer Science; Vol. 2499*. London, UK: Springer-Verlag, 2002. p. 1–10. ISBN 3-540-44282-0.

CARL, M. Inducing probabilistic invertible translation grammars from aligned texts. In: *Proceedings of CoNLL-2001*. Toulouse, France: [s.n.], 2001. p. 145–151.

CARLETTA, J. Assessing agreement on classification tasks: the kappa statistics. *Computational Linguistics*, v. 22, n. 2, p. 249–254, 1996.

CASELI, H. M. *Alinhamento sentencial de textos paralelos português-inglês*. 101 p. Dissertação (Mestrado) — ICMC-USP, Abril 2003. Disponível em: <http://www.nilc.icmc.usp.br/nilc/download/DissHelena.pdf>.

CASELI, H. M.; NUNES, M. G. V. *Anali: uma ferramenta de análise morfossintática*. Série de relatórios do NILC (NILC-TR-06-09), São Carlos-SP, 2006. 44 p. Disponível em: <http://www.nilc.icmc.usp.br/nilc/download/NILC-TR-06-09.zip>.

CASELI, H. M.; NUNES, M. G. V. *O sistema de indução de regras de tradução do projeto ReTraTos*. Série de relatórios do NILC, São Carlos-SP, no prelo.

CASELI, H. M.; NUNES, M. G. V.; FORCADA, M. L. Evaluating the LIHLA lexical aligner on Spanish, Brazilian Portuguese and Basque parallel texts. *Procesamiento del Lenguaje Natural*, Granada, Spain, v. 35, p. 237–244, 2005. ISSN 1135-5948.

CICEKLI, I.; GÜVENIR, H. A. Learning translation templates from bilingual translation examples. *Applied Intelligence*, v. 15, p. 57–76, 2001.

CRAGGS, R.; WOOD, M. M. Evaluating discourse and dialogue coding schemes. *Computational Linguistics*, v. 31, p. 289–295, 2005.

DAGAN, I.; CHURCH, K.; GALE, W. Robust word alignment for machine aided translation. In: *Proceedings of the Workshop on Very Latge Corpora: Academic and Industrial Perpectives*. [S.l.: s.n.], 1993. p. 1–8.

DAGAN, I.; ITAI, A. Word sense disambiguation using a second language monolingual corpus. *Computational Linguistics*, v. 20, n. 4, p. 563–596, 1994.

DICE, L. R. Measures of the amount of ecological association between species. *Geology*, v. 26, p. 297–302, 1945.

DODDINGTON, G. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In: *Proceedings of ARPA Workshop on Human Language Technology*. San Diego: [s.n.], 2002. p. 128–132.

FINCH, A.; AKIBA, Y.; SUMITA, E. How does automatic machine translation evaluation correlate with human scoring as the number of reference translation increases? In: *Proceedings if the 4th International Conference on Language Resources and Evaluation (LREC 2004)*. [S.l.: s.n.], 2004. p. 2019–2022.

FONT-LLITJÓS, A.; CARBONELL, J. G.; LAVIE, A. A framework for interactive and automatic refinement of transfer-based machine translation. In: *Proceedings of the EAMT 2005*. [S.l.: s.n.], 2005. p. 1–10.

FORCADA, M. L.; ROSELL, M. G.; BONEV, B. I.; ROJAS, S. O.; PÉREZ-ORTIZ, J. A.; RAMÍREZ-SÁNCHEZ, G.; SÁNCHEZ-MARTÍNEZ, F. *Documentación del sistema de código abierto Apertium de traducción automática de transferencia sintáctica superficial*. [S.l.], 2005. 118 p.

FOSSEY, M. F.; PEDROLONGO, T.; MARTINS, R. T.; NUNES, M. G. V. *Análise comparativa de tradutores automáticos inglês-português*. São Carlos, SP, 2004. 18 p.

FUNG, P. A pattern matching method for finding noun and proper noun translations from noisy parallel corpora. In: *Proceedings of the Meeting of the Association for Computational Linguistics*. [S.l.: s.n.], 1995. p. 236–243.

FUNG, P.; CHURCH, K. W. K-vec: a new approach for aligning parallel texts. In: *Proceedings of the 15th International Conference on Computational Linguistics*. Kyoto, Japan: [s.n.], 1994.

FURUSE, O.; IIDA, H. An Example-Based Method for Transfer-Driven Machine Translation. In: *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation. Empiricist vs. Rationalist Methods in MT. TMI-92*. Montréal, Québec: [s.n.], 1992. p. 139–150.

GALE, W. A.; CHURCH, K. W. Identifying word correspondences in parallel texts. In: *Proceedings of the 4th DARPA Speech and Language Workshop*. Pacific Grove, CA: [s.n.], 1991. p. 152–157.

GALLEY, M.; HOPKINS, M.; KNIGHT, K.; MARCU, D. What's in a translation rule? In: *Proceedings of the NAACL-HLT*. [S.l.: s.n.], 2004. p. 273–280.

GARRIDO-ALENDA, A.; FORCADA, M. L. Morphtrans: un lenguaje y un compilador para especificar y generar módulos de transferencia morfológica para sistemas de tradución automática. *Procesamiento del Lenguaje Natural*, v. 27, p. 157–164, 2001.

GARRIDO-ALENDA, A.; FORCADA, M. L.; CARRASCO, R. C. Incremental construction and maintenance of morphological analysers based on augmented letter transducers. In: *Proceedings of TMI 2002 (Theoretical and Methodological Issues in Machine Translation)*. Keihanna/Kyoto, Japan: [s.n.], 2002. p. 53–62.

GARRIDO-ALENDA, A.; ITURRASPE, A.; MONTSERRAT, S.; PASTOR, H.; FORCADA, M. L. A compiler for morphological analysers and generators based on finite-state transducers. *Procesamiento del Lenguaje Natural*, v. 25, p. 93–98, 1999.

GARRIDO-ALENDA, A.; ZARCO, P. G.; PÉREZ-ORTIZ, J. A.; PERTUSA-IBÁÑEZ, A.; RAMÍREZ-SÁNCHEZ, G.; SÁNCHEZ-MARTÍNEZ, F.; SCALCO, M. A.; FORCADA, M. L. Shallow parsing for Portuguese-Spanish machine translation. In: BRANCO, A.; MENDES, A.; RIBEIRO, R. (Ed.). *Language technology for Portuguese: shallow processing tools and resources*. Lisboa: [s.n.], 2004. p. 135–144. ISBN 972-772-494-9.

GILDEA, D. Loosely tree-based alignment for machine translation. In: *Proceedings of the 41st Annual Conference of the Association for Computational Linguistics (ACL-03)*. Sapporo, Japan: [s.n.], 2003. p. 80–87.

GROVES, D.; WAY, A. Hybrid example-based SMT: the best of both worlds? In: *Proceedings of the ACL Workshop on Building and Using Parallel Texts*. Ann Arbor, USA: [s.n.], 2005. p. 183–190.

GÜVENIR, H. A.; CICEKLI, I. Learning translation templates from examples. *Information Systems*, v. 23, n. 6, p. 353–363, 1998.

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

HUTCHINS, J. Towards a definition of example-based machine translation. In: *Proceedings of MT Summit X*. [S.l.: s.n.], 2005. p. 63–70.

IMAMURA, K.; SUMITA, E.; MATSUMOTO, Y. Automatic construction of machine translation knowledge using translation literalness. *Natural Language Processing (Japan)*, v. 11, n. 2, p. 85–99, 2004.

KAJI, H.; KIDA, Y.; MORIMOTO, Y. Learning translation templates from bilingual text. In: *Proceedings of COLING-92*. [S.l.: s.n.], 1992. p. 672–678.

KITAMURA, M. *Translation knowledge acquisition for pattern-based machine translation*. 114 p. Tese (Doutorado) — Departament of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, November 2004.

KOEHN, P.; KNIGHT, K. Learning a translation lexicon from monolingual corpora. In: *Proceedings of the Workshop of the ACL Special Interest Group on the Lexicon (SIGLEX)*. Philadelphia: [s.n.], 2002. p. 9–16.

LANGLAIS, P.; FOSTER, G.; LAPALME, G. Integrating bilingual lexicons in a probabilistic translation assistant. In: *Proceedings of the 8th MT Summit: Machine Translation in the Information Age*. Santiago de Compostela, Spain: [s.n.], 2001. p. 197–202.

LAVIE, A.; PROBST, K.; PETERSON, E.; VOGEL, S.; LEVIN, L.; FONT-LLITJÓS, A.; CARBONELL, J. A trainable transfer-based machine translation approach for languages with limited resources. In: *Proceedings of the 9th Workshop of the European Association for Machine Translation (EAMT-04)*. Valletta, Malta: [s.n.], 2004. p. 1–8.

LAVOIE, B.; WHITE, M.; KORELSKY, T. Inducing lexico-structural transfer rules from parsed bi-texts. In: *Proceedings of the Workhop on Data-driven Machine Translation at 39th Annual Meeting of the Association for Computational Linguistics (ACL'01)*. Toulouse, France: [s.n.], 2001. p. 17–24.

LAVOIE, B.; WHITE, M.; KORELSKY, T. Learning domain-specific transfer rules: an experiment with Korean to English translation. In: *Proceedings of the COLING 2002 Workshop on Machine Translation in Asia*. Tapei, Taiwan: [s.n.], 2002. p. 60–66.

LEVENSHTEIN, V. I. Binary codes capable of correcting deletions, insertions, and reversals. *Cybernetics and Control Theory*, v. 10, p. 707–710, 1966.

LIU, Y.; ZONG, C. The technical analysis on translation templates. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*. [S.l.: s.n.], 2004. p. 4799–4803.

MANNING, C. D.; SCHUTZE, H. *Foundations of statistical natural language processing*. [S.l.]: MIT Press, 1999. 172–175 p.

MARIÑO, J. B.; BANCHS, R.; CREGO, J. M.; GISPERT, A. de; LAMBERT, P.; FONOLLOSA, J. A. R. Modelo estocástico de traducción basado en n-gramas de tuplas bilingües y combinación log-lineal de características. *Procesamiento del Lenguaje Natural*, n. 35, p. 69–76, 2005.

MARTIN, J.; MIHALCEA, R.; PEDERSEN, T. Word alignment for languages with scarce resources. In: *Proceedings of the ACL Workshop on Building and Exploiting Parallel Texts: Data Driven Machine Translation and Beyond*. Ann Arbor, United States: [s.n.], 2005. p. 1–10.

MARUYAMA, H.; WATANABE, H. Tree cover search algorithm for example-based translation. In: *Proceedings of the 4th International Conference on Theoretical and Methodological Issues in Machine Translation: Empiricist vs. Rationalist Methods in MT (TMI'92)*. Montreal, Canada: [s.n.], 1992. p. 173–184.

MATSUMOTO, Y.; ISHIMOTO, H.; UTSURO, T.; NAGAO, M. Structural matching of parallel texts. In: *Proceedings of the 31st Annual Meeting of the ACL*. [S.l.: s.n.], 1993. p. 23–30.

MCTAIT, K. Translation patterns, linguistic knowledge and complexity in an approach to EBMT. In: CARL, M.; WAY, A. (Ed.). *Recent Advances in Example-Based Machine Translation*. Printed in Netherlands: Kluwer Academic Publishers, 2003. p. 1–28.

MCTAIT, K.; TRUJILLO, A. A language-neutral sparse-data algorithm for extracting translation patterns. In: *Proceedings of the 8th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-99)*. Chester, England: [s.n.], 1999. p. 98–108.

MELAMED, I. D. Automatic evaluation and uniform filter cascades for inducing n-best translation lexicons. In: *Proceedings of 3rd Annual Workshop on Very Large Corpora (WVLC-95)*. [S.l.: s.n.], 1995. p. 184–198.

MELAMED, I. D. A geometric approach to mapping bitext correspondence. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Philadelphia, USA: [s.n.], 1996.

MELAMED, I. D. Automatic construction of clean broad-coverage translation lexicons. In: *Proceedings of the 2nd Conference of the Association for Machine Translation in the Americas (AMTA-1996)*. Montreal, Canada: [s.n.], 1996. p. 125–134.

MELAMED, I. D. Automatic detection of omissions in translations. In: *Proceedings of the 16th International Conference on Computational Linguistics (COLING-1996)*. Copenhagen, Denmark: [s.n.], 1996. p. 764–769.

MELAMED, I. D. A portable algorithm for mapping bitext correspondence. In: COHEN, P. R.; WAHLSTER, W. (Ed.). *Proceedings of the 35th Annual Meeting of the ACL and 8th Conference of the EACL*. Somerset, New Jersey: Association for Computational Linguistics, 1997. p. 305–312.

MELAMED, I. D. *A scalable architecture for bilingual lexicography*. Department of Computer and Information Science, MS-CIS-97-01, 1997.

MELAMED, I. D.; GREEN, R.; TURIAN, J. P. Precision and recall of machine translation. In: *Proceedings of NAACL/HLT 2003*. Edmonton, Canada: [s.n.], 2003. p. 61–63.

MENEZES, A. Better contextual translation using machine learning. In: *Proceedings of the 5th conference of the Association for Machine Translation in the Americas*. Tiburon, California: [s.n.], 2002. p. 124–134.

MENEZES, A.; RICHARDSON, S. D. A best-first alignment algorithm for automatic extraction of transfer mappings from bilingual corpora. In: *Proceedings of the Workshop on Data-driven Machine Translation at 39th Annual Meeting of the ACL*. Toulouse, France: [s.n.], 2001. p. 39–46.

MEYERS, A.; KOSAKA, M.; GRISHMAN, R. Chart-based translation rule application in machine translation. In: *Proceedings of COLING-2000*. [S.l.: s.n.], 2000. p. 537–543.

MEYERS, A.; YANGARBER, R.; GRISHMAN, R. Alignment of shared forests for bilingual corpora. In: *Proceedings of COLING-96*. [S.l.: s.n.], 1996. p. 460–465.

MEYERS, A.; YANGARBER, R.; GRISHMAN, R.; MACLEOD, C.; MORENO-SANDOVAL, A. Deriving transfer rules from dominance-preserving alignments. In: *Proceedings of Coling-ACL98: The 17th International Conference on Computational Linguistics and the 36th Meeting of the ACL*. [S.l.: s.n.], 1998. p. 843–847.

MUNIZ, M. C. M. *A construção de recursos lingüístico-computacionais para o português do Brasil: o projeto Unitex-PB*. 72 p. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos-SP, 2004.

NAGAO, M. A framework of a mechanical translation between Japanese and English by analogy principle. In: ELITHORN, A.; BANERJI, R. (Ed.). *Artificial and Human Intelligence*. [S.l.]: NATO Publications, 1984. p. 173–180.

OCH, F. J. An efficient method for determining bilingual word classes. In: *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*. Bergen, Norway: [s.n.], 1999. p. 71–76.

OCH, F. J.; NEY, H. A comparison of alignment models for statistical machine translation. In: *Proceedings of the 18th International Conference on Computational Linguistics (COLING'00)*. Saarbrücken, Germany: [s.n.], 2000. p. 1086–1090.

OCH, F. J.; NEY, H. Improved statistical alignment models. In: *Proceedings of the 38th Annual Meeting of the ACL*. Hong Kong, China: [s.n.], 2000. p. 440–447.

OCH, F. J.; NEY, H. A systematic comparison of various statistical alignment models. *Computational Linguistics*, v. 29, n. 1, p. 19–51, 2003.

OCH, F. J.; NEY, H. The alignment template approach to statistical machine translation. *Computational Linguistics*, v. 30, n. 4, p. 417–449, 2004.

OLIVEIRA Jr., O. N.; MARCHI, A. R.; MARTINS, M. S.; MARTINS, R. T. A critical analysis of the performance of English-Portuguese-English MT systems. In: NUNES, M. G. V. (Ed.). *Proceedings of V Encontro para o processamento computacional da língua portuguesa escrita e falada (PROPOR 2000)*. Atibaia, SP: [s.n.], 2000. p. 85–92.

ÖZ, Z.; CICEKLI, I. Ordering translation templates by assigning confidence factors. In: *Lecture Notes in Computer Science*. [S.l.]: Springer-Verlag, 1998. v. 1529, p. 51–61.

PAPINENI, K.; ROUKOS, S.; WARD, T.; ZHU, W. BLEU: a method for automatic evaluation of machine translation. In: *Proceedings of the 40th Annual Meeting of the ACL*. Philadelphia, PA: [s.n.], 2002. p. 311–318.

PAUMIER, S. *Unitex 1.2 User Manual*. Université de Marne-la-Vallée, June 2006. 217 p.

PEI, J.; HAN, J.; MORTAZAVI-ASL, B.; PINTO, H.; CHEN, Q.; DAYAL, U.; HSU, M. PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth. In: *Proceedings of International Conference of Data Engineering (ICDE2001)*. [S.l.: s.n.], 2001. p. 215–224.

PEI, J.; HAN, J.; MORTAZAVI-ASL, B.; WANG, J.; PINTO, H.; CHEN, Q.; DAYAL, U.; HSU, M. Mining sequential patterns by pattern-growth: the PrefixSpan approach. *IEEE Transactions on Knowledge and Data Engineering*, v. 16, n. 10, p. 1–17, October 2004.

PROBST, K. *Learning transfer rules for machine translation with limited data*. Tese (Doutorado) — Language Technologies Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, U.S.A., August 2005.

PROBST, K.; LEVIN, L.; PETERSON, E.; LAVIE, A.; CARBONELL, J. MT for minority languages using elicitation-based learning of syntactic transfer rules. *Machine Translation*, Kluwer Academic Publishers, Netherlands, v. 17, n. 4, p. 1–30, August 2003.

RESNIK, P.; MELAMED, I. D. Semi-automatic acquisition of domain-specific translation lexicons. In: *ANLP*. [S.l.: s.n.], 1997. p. 340–347.

RICHARDSON, S. D.; DOLAN, W.; CORSTON-OLIVER, M.; MENEZES, A. Overcoming the customization bottleneck using example-based MT. In: *Workshop on Data-Driven Machine Translation, ACL 2001*. Toulouse, France: [s.n.], 2001. p. 9–16.

SÁNCHEZ-MARTÍNEZ, F.; NEY, H. Using alignment templates to infer shallow-transfer machine translation rules. In: SALAKOSKI FILIP GINTER, S. P. T.; PAHIKKALA, T. (Ed.). *Advances in Natural Language Processing, Proceedings of 5th International Conference on Natural Language Processing FinTAL*. [S.l.]: Springer-Verlag, 2006. (Lecture Notes in Computer Science, v. 4139), p. 756–767.

SCHAFER, C.; YAROWSKY, D. Inducing translation lexicons via diverse similarity measures an bridge languages. In: *Proceedings of CoNLL-2002*. [S.l.: s.n.], 2002. p. 1–7.

SENELLART, P.; SENELLART, J. SYSTRAN translation stylesheets: machine translation driven by XSLT. In: *XML Conference e Exposition*. Atlanta, USA: [s.n.], 2005. p. 1–15.

SIMARD, M.; FOSTER, G. F.; ISABELLE, P. Using cognates to align sentences in bilingual corpora. In: *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*. Montreal, Canada: [s.n.], 1992. p. 67–82.

SOMERS, H. Review article: Example-based machine translation. *Machine Translation*, v. 14, n. 2, p. 113–157, 1999.

TENNI, J.; LEHTOLA, A.; CATHERINE, B.; KRISTIINA, J. Machine learning of language translation rules. In: *1999 IEEE Systems, Man and Cybernetics Conference (SMC'99)*. Tokyo: [s.n.], 1999. p. 171–177.

TURIAN, J. P.; SHEN, L.; MELAMED, I. D. Evaluation of machine translation and its evaluation. In: *Proceedings of the IX MT Summit*. New Orleans, USA: [s.n.], 2003. p. 386–393.

VEALE, T.; WAY, A. Gaijin: A template-driven bootstrapping approach to example-based machine translation. In: *Proceedings of the NeMNLP'97, New Methods in Natural Language Processing*. Sofia, Bulgaria: [s.n.], 1997. p. 1–14.

VOGEL, S.; NEY, H.; TILLMANN, C. HMM-based word alignment in statistical translation. In: *COLING'96: The 16th International Conference on Computational Linguistics*. Copenhagen: [s.n.], 1996. p. 836–841.

WU, D.; XIA, X. Learning an English-Chinese lexicon from parallel corpus. In: *Proceedings of the 1st Conference of the Association for Machine Translation in the Americas (AMTA-1994)*. Columbia, MD: [s.n.], 1994. p. 206–213.

YAMADA, K.; KNIGHT, K. A syntax-based statistical translation model. In: *Proceedings of the 39th Meeting of the ACL*. Toulouse, France: [s.n.], 2001. p. 1–8.

YAMAMOTO, K.; KUDO, T.; TSUBOI, Y.; MATSUMOTO, Y. Learning sequence-to-sequence correspondences from parallel corpora via sequential pattern mining. In: *Proceedings of HLT-NAACL 2003*. [S.l.: s.n.], 2003. p. 73–80.

APÊNDICE A

Símbolos gramaticais usados no projeto ReTraTos

As Tabelas 44, 45 e 46 apresentam as etiquetas utilizadas no projeto ReTraTos para a PoS (primeira tabela) e os traços morfossintáticos (demais tabelas) de cada palavra nos exemplos de tradução utilizados neste trabalho. O conjunto de etiquetas utilizado no ReTraTos é, basicamente, o mesmo utilizado por `Apertium`, porém novas etiquetas foram inseridas, pricipalmente, como resultado da utilização dos dados lingüísticos de `Unitex` no incremento dos dicionários morfológicos de `pt` e `en`.

Na simbologia usada nas Tabelas 44, 45 e 46, as etiquetas de ReTraTos que não fazem parte do conjunto de etiquetas definidas no `Apertium` são apresentadas seguidas do caractere "*". A não ocorrência de uma etiqueta nos exemplos em uma determinada língua é indicada pela seqüência "NC" (não consta) e sua não aplicação a um dado idioma (não faz parte do conjunto de etiquetas definido para tal idioma) é indicada pelo caractere "–".

A Tabela 44 apresenta as etiquetas que representam PoS, bem como uma descrição sucinta das mesmas e exemplos de palavras etiquetadas dessa maneira nos três idiomas: `pt`, `es` e `en`. As Tabelas 45 e 46 apresentam as etiquetas que representam os traços morfossintáticos atribuídos às palavras de acordo com suas PoS. Além das etiquetas, essas tabelas apresentam uma descrição sucinta das mesmas, as principais PoS às quais se aplicam e exemplos de palavras etiquetadas dessa maneira para os três idiomas: `pt`, `es` e `en`.

Nas Tabelas 45 e 46, para simplificar, as várias etiquetas de verbos foram agrupadas em uma classe maior denominada `verbos`, assim, o uso da denominação `verbos` corresponde às etiquetas `vblex`, `vbser`, `vbhaver`, `vbmod`, `vaux` e `v`. Além disso, a PoS a qual o exemplo se refere é a que aparece primeiro na lista de etiquetas ou a que vem entre parênteses, no caso dos verbos.

Tabela 44: Etiquetas utilizadas para representar PoS no ReTraTos

Etiqueta	**Descrição**	**Exemplos**		
		pt	es	en
abr*	abreviatura	Jr	–	–
adj	adjetivo	antigo	antiguo	oldest
adv	advérbio	mais	más	more
apos	apóstrofo	–	–	’
cm	vírgula	,	,	,
cnjadv	conjunção adverbial	se	si	if
cnjcoo	conjunção coordenativa	e	y	and
cnjsub	conjunção subordinativa	que	que	that
det	determinante	o	el	the
detnt	determinante neutro	o	lo	NC
fw*	palavra estrangeira	–	–	versa
gen	genitivo	–	–	’s
guio	hífen	–	–	-
ij	interjeição	por_favor	por_favor	Ah
lpar	(	(	(	(
lquest	¿	NC	¿	NC
n	substantivo	dentes	dientes	teeth
np	nome próprio	Tailândia	Tailandia	Thailand
num	numeral	12	12	12
pfx*	prefixo	hiper	–	intra
pr	preposição	de	de	of
preadv	pré-advérbio	muito	muy	most
predet	pré-determinante	todos	todos	all
prn	pronome	ninguém	nadie	nobody
rel	relativo	como	como	that
rpar	)	)	)	)
sent	.?;:!	.	.	.
sig*	sigla	USP	–	NC
v*	verbo	abanar	–	fossilized
vaux	verbo auxiliar	–	–	will
vbhaver	verbo haver	teria	habría	have
vblex	verbo léxico	estimular	estimular	found
vbdo	verbo *do*	–	–	do
vbmod	verbo modal	poderiam	podrían	have_to
vbser	verbo ser	é	es	is

Tabela 45: Etiquetas utilizadas para representar os traços morfossintáticos no ReTraTos (parte 1)

Etiqueta	**Descrição**	**PoS**	**Exemplos**		
			pt	es	en
aa	adjetivo-adjetivo	rel	cujas	cuyas	NC
acr	acrônimo	n	OMS	OMS	DNA
adv	adverbial	rel	como	como	where
al	outro	np	Alzheimer	Alzheimer	NC
an	adjetivo-nome	rel	que	que	that
ant	antropônimo	np	Garcia	Garcia	Alex
cni	condicional	verbos (vblex)	poderiam	podrían	NC
comp	comparativo	adj, adv	–	–	greater
def	definido	det	o	el	the
dem	demonstrativo	det	esse	ese	this
enc	enclítico	prn	NC	iniciar**se**	NC
f	feminino	adj, det, predet, n, np, num, prn, rel, verbos	antigas	antiguas	Clarice
fti	futuro ind.	verbos (vblex)	permitirão	permitirán	NC
fts	futuro subj.	verbos (vbser)	for	fuere	–
ger	gerundio	verbos (vblex)	comparando	comparando	comparing
ifi	pret. perfeito ou indefinido	verbos (vbmod)	pôde	pudo	NC
imp	imperativo	verbos (vblex)	considere	considérese	NC
ind	indeterminado	det	um	un	a
inf	infinitivo	verbos (vblex)	escrever	escribir	write
infps	infinitivo pessoal	verbos	serem	–	–
itg	interrogativo	adv, prn	por_que	por_qué	why
loc	locativo	np	Europa	Europa	Europe
m	masculino	predet, adj, det, n, np, num, prn, rel, verbos	todos	todos	Alex
mf	masculino-feminino	num, adj, det, n, prn, rel	mil	mil	they
nn	nome-nome	rel	quem	quienes	those_who
nt	neutro	rel, prn	o_que	lo_que	it
obj	objeto	prn	–	–	it
ord	ordem	det	–	–	last

Tabela 46: Etiquetas utilizadas para representar os traços morfossintáticos no ReTraTos (parte 2)

Etiqueta	**Descrição**	**PoS**	**Exemplos**		
			pt	es	en
p1	1a. pessoa	verbos (vbmod), prn	deveria	debería	we
p2	2a. pessoa	prn, verbos	te	ti	you
p3	3a. pessoa	verbos (vblex), prn	disse	dijo	has
past	passado	verbos	–	–	connected
pii	pret. imperfeito ind.	verbos (vblex)	estavam	estaban	underwent
pis	pret. imperfeito sub.	verbos (vbser)	fosse	fuese	NC
pl	plural	verbos (vblex), adj, det, predet, n, vnum, prn, rel	precisamos	necesitamos	teeth
pmp	pret. mais que perf.	verbos	comprovara	–	–
pos	possessivo	adj, det	nossa	nuestra	our
pp	particípio	verbos (vblex)	construído	construido	built
pres	presente	verbos (vbhaver)	–	–	have
pri	presente indicativo	verbos (vblex)	é	es	is
pro	proclítico	prn	verifica-**se**	**se** verifica	NC
prs	presente subjuntivo	verbos (vblex)	integre	abarque	NC
qnt	quantitativo	det	–	–	some
ref	reflexivo	prn	verifica-**se**	**se** verifica	themselves
sep	pode ser separado	verbos (vblex)	–	–	set up
sint	sintético (-er, -est)	adj	–	–	light
sg	singular	n, adj, det, predet, num, prn, rel, verbos (vblex)	trabalho	trabajo	work
sp	singular-plural	rel, adj, det, predet, n, num, prn	que	que	the
subj	sujeito	prn	–	–	I
sup	superlativo	adj	NC	NC	oldest
tn	tônico	prn	com **isso**	con **ello**	both
unc	incontável (*uncountable*)	n	–	–	safety