Reconhecimento de Informações Comuns para a Fusão de Sentenças Comparáveis do Português

Eloize Rossi Marques Seno, Maria das Graças Volpe Nunes
NILC – ICMC – Universidade de São Paulo
São Carlos – SP, Brasil
{eloize,gracan}@icmc.usp.br

Resumo

A fusão de sentenças é uma tarefa que consiste em produzir, a partir de um conjunto de sentenças relacionadas, uma única sentença que resume as informações comuns apresentadas no conjunto. Essa tarefa é de grande interesse em diversas aplicações do Processamento de Língua Natural (PLN), tais como a Sumarização Automática, a Tradução Automática, os sistemas de Perguntas e Respostas, entre outros. No entanto, um dos principais desafios da fusão consiste em identificar as informações comuns entre as sentenças relacionadas. Este trabalho apresenta um sistema baseado em conhecimento lexical, sintático, semântico e em algumas regras de parafraseamento que permite o reconhecimento de seqüências de palavras distintas, mas com o mesmo significado em sentenças comparáveis do Português. Os experimentos realizados com o sistema mostraram um desempenho de 87% de Precisão, 83% de Cobertura e 85% de Medida-f. Os resultados estão de acordo com outros trabalhos reportados na literatura para outras línguas.

1. Introdução

A fusão de sentenças é uma tarefa de geração de texto a partir de texto (*text-to-text generation*, em inglês) que, dadas duas ou mais sentenças semanticamente relacionadas como entrada, produz uma nova sentença de saída, preservando as informações comuns entre elas (Barzilay, 2003; Barzilay and Mckeown, 2005). A fusão de sentenças é uma área de pesquisa emergente em Processamento de Língua Natural (PLN) e é motivada por aplicações práticas tais como a Tradução Automática (Pang et al., 2003), a Sumarização Automática (vide Barzilay and Mckeown, 2005), os sistemas de Perguntas e Respostas (vide Marsi and Krahmer, 2005; Krahmer et al. 2008), entre outras. Na sumarização multidocumento, por exemplo, o processo de fusão de informações comuns é de grande relevância para eliminar a redundância de informações nos sumários, especialmente no que diz respeito aos métodos de sumarização extrativos que identificam as sentenças (ou parágrafos) mais importantes de um conjunto de documentos e as extraem para compor o sumário. A repetição de informações influencia diretamente a qualidade dos sumários, prejudicando, principalmente, a coesão e a coerência. A fusão de várias sentenças que expressam uma mesma informação em uma única sentença pode minimizar esses problemas, eliminando a repetição de informações e, conseqüentemente, melhorando a textualidade dos sumários.

A Figura 1 apresenta um exemplo de sentença produzida a partir da fusão automática de três sentenças comparáveis sobre um mesmo assunto, porém de fontes distintas, extraídas do corpus de trabalho (Seção 3.1). No exemplo da figura, a sentença resultante da fusão corresponde à intersecção das sentenças [1], [2] e [3] e expressa somente os conceitos comuns a todas elas (em negrito).

[1] O Airbus A320, vôo JJ 3054, partiu de Porto Alegre, às 17h16 da terça-feira e chegou a São Paulo às 18h45.
[2] A aeronave da TAM Airbus A320, vôo JJ 3054, partiu de Porto Alegre, às 17h16 com destino a Congonhas.
[3] Um Airbus A320 com capacidade para 170 passageiros **partiu de Porto Alegre (RS) às 17h16** com destino a Congonhas.
Fusão das sentenças [1], [2] e [3]: O Airbus A320, vôo JJ 3054, partiu de Porto Alegre (RS) às 17h16.

Figura 1: Exemplo de Fusão de Sentenças[1]

[1] Essas sentenças foram identificadas automaticamente pelo sistema de clustering SiSPI (vide Seção 3.1), a partir de um conjunto de cinco documentos sobre o acidente envolvendo o Airbus A320, vôo JJ 3054, da TAM.

Nos trabalhos existentes na literatura (por exemplo, Pang et al. 2003; Barzilay and Mckeown, 2005 e Marsi and Krahmer, 2005) a fusão de sentenças é comumente dividida em três etapas, a saber: i) identificação de informações comuns, ii) fusão de informações e iii) linearização. A primeira etapa consiste em reconhecer informações semanticamente similares (por exemplo, paráfrases e sinônimos) que se repetem nas sentenças. A segunda etapa consiste em escolher os itens lexicais que irão compor a nova sentença e determinar o modo como eles serão combinados na sentença. A última etapa, por sua vez, consiste em realizar em língua natural a sentença obtida a partir da etapa anterior e envolve, portanto, aspectos gramaticais da sentença. A identificação dos elementos que expressam informações comuns e a combinação desses elementos para a geração da nova sentença consistem no maior desafio na construção de algoritmos de fusão.

Neste artigo apresenta-se um alinhador de conceitos similares baseado em informações lexicais, sintáticas e semânticas que permite o reconhecimento de informações comuns em sentenças comparáveis do português. Com base no alinhamento de duas ou mais árvores de dependência sintática que representam cada sentença de um conjunto de sentenças comparáveis, constrói-se uma floresta a partir da união de sentenças previamente alinhadas (ou seja, unindo as informações comuns a cada sentença). A união de todas as sentenças em uma única estrutura de dependência sintática possibilita que um subseqüente módulo de fusão e linearização gere todas as sentenças possíveis a partir da floresta. Um modelo probabilístico de língua é utilizado, posteriormente, para auxiliar na seleção da melhor sentença gerada, como proposto no trabalho de Barzilay and Mckeown (2005).

Em um trabalho anterior (Seno and Nunes, 2008a) foi apresentada uma versão preliminar do alinhador para a identificação de informações comuns entre pares de sentenças comparáveis (os resultados obtidos são sumarizados na Seção 4). No presente trabalho, são apresentadas diversas modificações realizadas ao sistema, por exemplo, a possibilidade de alinhamento de um conjunto de sentenças (e não somente de pares de sentenças), mudanças na estratégia de alinhamento, a inclusão de novos conhecimentos lingüísticos e modificações no pré-processamento, que resultaram em um melhor desempenho do sistema (vide Seção 4).

O restante deste trabalho está organizado da seguinte forma: a Seção 2 apresenta alguns trabalhos correlatos de alinhamento de informações comuns; a Seção 3 apresenta o alinhador proposto; a Seção 4 mostra alguns experimentos realizados e, por fim, a Seção 5 apresenta as conclusões e algumas possibilidades de trabalhos futuros.

2. Trabalhos Relacionados

As abordagens de alinhamento de informações similares existentes na literatura se distinguem em dois aspectos principais: i) quanto ao tipo de sentença de entrada e ii) quanto ao tipo de conhecimento usado. Quanto aos tipos de sentenças têm-se as sentenças comparáveis, que se referem a um mesmo fato ou evento, porém são de fontes de informação diferentes, e as sentenças paralelas, que são traduções distintas de uma mesma fonte para uma mesma língua alvo. Em relação aos tipos de conhecimento utilizados para a identificação de conceitos similares, destacam-se as informações sintáticas, por exemplo, as relações de dependência entre os constituintes sintáticos, as relações semânticas, os léxicos de sinônimos e de paráfrases.

Pang et al. (2003) alinham árvores sintáticas de sentenças paralelas usando somente informações de *part-of-speech* (*POS*). As palavras com o mesmo POS são tratadas como paráfrases. Embora essa abordagem tenha se mostrado satisfatória para trabalhar com sentenças paralelas, somente informações de POS não são suficientes para o reconhecimento de conceitos comuns em sentenças comparáveis, uma vez que as estruturas sintáticas dessas sentenças nem sempre são similares, como é o caso das sentenças paralelas. Já Shen et al. (2006) consideram, além das informações de POS, os traços de dependência dos constituintes sintáticos. O alinhamento ocorre somente entre palavras lexicalmente similares que compartilham o mesmo POS e o mesmo traço de dependência.

Ao contrário desses trabalhos, outras abordagens ignoram completamente as informações de POS.

Em Marsi and Krahmer (2005), por exemplo, o alinhamento envolvendo sentenças paralelas é baseado apenas na similaridade de suas correspondentes estruturas de dependência sintática e em relações semânticas (por exemplo, *restates* e *intersects*). O alinhamento entre duas palavras só se realiza se houver uma relação semântica entre elas. Os autores relatam uma precisão de 86% e uma cobertura de 84% (isto é, 85% de Medida-f) do sistema[2]. Contudo, a principal limitação desse método está na dificuldade de se construir *parsers* semânticos. Outra limitação, que também se pode observar nos trabalhos de Pang et al. (2003) e Shen et al. (2006), é que essas abordagens não tratam o reconhecimento de paráfrases multipalavras. Como exemplos desse tipo de paráfrases tem-se *mercado moscovita* com *mercado Cherskisov de Moscou* e *capital da Rússia* com *capital russa*.

Já em Barzilay and Mckeown (2005), o alinhamento de informações similares entre sentenças comparáveis ocorre em nível de palavras e de *phrases*. Assim como em Marsi and Krahmer (2005), os autores também consideram a similaridade entre as estruturas de dependência sintática das sentenças. A similaridade entre palavras é obtida a partir de um conjunto de sinônimos, enquanto a similaridade entre multipalavras é determinada com o uso de um léxico de paráfrases, induzido automaticamente a partir de corpora. Entretanto, a construção de um léxico representativo de paráfrases requer um grande volume de dados de treinamento (ou seja, de sentenças parafrásticas), um recurso praticamente inexistente para a maioria das línguas.

As paráfrases multipalavras são as mais freqüentes, principalmente em sentenças comparáveis (vide Seção 3.2) e são muito difíceis de se tratar automaticamente.

O método descrito neste trabalho faz uso de regras de parafraseamento, identificadas a partir da análise de corpora (vide Seção 3), e de conhecimentos lexical, sintático (ou seja, POS e traços de dependência) e semântico (isto é, relações de sinonímia) que possibilitam a identificação de palavras e multipalavras que conduzem informações semanticamente similares em sentenças do português.

3. Reconhecimento de Informações Comuns

Esta seção apresenta o alinhador de informações comuns, destacando as melhorias realizadas em relação à primeira versão do sistema. Antes, porém, as subseções 3.1 e 3.2 descrevem a construção do corpus de trabalho e a formulação das regras de parafraseamento a partir da análise do corpus, respectivamente.

3.1 Construção do Corpus

Para a construção do corpus de sentenças comparáveis, foram coletadas manualmente 50 coleções de documentos a partir de diversas agências de notícias brasileiras disponíveis na web. O corpus compreende textos de diferentes domínios, tais como ciência, cotidiano, esporte, mundo e política. Cada coleção é composta por aproximadamente 4 documentos relacionados a um mesmo assunto, totalizando 71 documentos e 1.153 sentenças em todo o corpus. Todos os documentos de uma mesma coleção foram publicados em uma mesma data, o que assegura uma maior similaridade do conteúdo apresentado nesses documentos.

Após a coleta dos textos, cada coleção de documentos foi submetida a um processo de agrupamento de sentenças, para a identificação das sentenças comparáveis de cada coleção. Para esse processo foi desenvolvido o sistema SiSPI (Seno and Nunes, 2008b), baseado em um método de agrupamento incremental e não supervisionado conhecido por *Single-pass* (Van Rijsbergen**,** 1979). A abordagem incremental tem a vantagem de não ser baseada em treinamento e, portanto, não requer grandes conjuntos de dados.

O Single-pass, como o próprio nome sugere, requer um único passo seqüencial sob todo o conjunto de sentenças a ser agrupado. Dado um conjunto de documentos como entrada, o primeiro grupo é criado selecionando-se a primeira sentença do primeiro documento do conjunto. A cada iteração, o algoritmo verifica se a nova sentença de entrada deve pertencer a algum grupo já existente ou se um novo grupo

[2] Outros trabalhos reportados aqui não relatam resultados sobre o processo de alinhamento de informações em específico, já que esse é um processo intermediário.

deve ser criado para aquela sentença. Essa decisão é baseada em uma condição previamente estabelecida para a função de similaridade adotada, ou seja, um limiar de similaridade. Duas funções distintas foram implementadas no sistema, para calcular a distância semântica entre uma sentença e um grupo. A primeira função é baseada na medida *Word-Overlap* (*Wol*) (Radev et al., 2008), que calcula o número de palavras em comum entre uma sentença *S* e um grupo *C*, normalizado pelo total de palavras de *S* e *C* (Fórmula 1). O valor de similaridade da *Wol* varia de 0 a 0,5. Quanto mais próximo de 0,5, maior é a similaridade entre a sentença e o grupo.

(1)

$$Wol(S,C) = \frac{\#PalavrasComuns(S,C)}{(|S| + |C|)}$$

A segunda função de similaridade é baseada na distância do co-seno (Salton and Allan, 1994) aplicada entre o vetor de freqüência de termos de uma sentença e o vetor que representa os termos mais importantes de um grupo, denominado centróide. O valor de similaridade dessa função varia de 0 a 1. Quanto mais próximo de 1, maior é a similaridade entre a sentença e o grupo.

O centróide de um grupo de sentenças é determinado a partir de duas medidas estatísticas. A primeira medida é uma adaptação do *TF-IDF* (*Term Frequency Inverse Document Frequency*) (Salton and Allan, 1994). O valor do *TF-IDF* de uma palavra *w* pertencente a um grupo *c*, denotado por *TF-IDF(w,c)*, é dado pela Fórmula 2, onde *TF(w,c)* representa a freqüência da palavra *w* no grupo *c*. Quanto maior o valor de *TF*, mais representativa do grupo a palavra *w* é. A freqüência de documento inversa de *w*, denotada por *IDF(w)*, é dada pela Fórmula 3, onde |*C*| representa o total de sentenças de toda a coleção de documentos e *DF(w)* representa o total de sentenças da coleção que contem *w*.

(2)

$$TF\text{-}IDF(w,c) = TF(w,c) * IDF(w)$$

(3)

$$IDF(w) = 1 + \log (|C| / DF(w))$$

A segunda medida usada para calcular o centróide de um grupo é a *TF-ISF* (*Term Frequency Inverse Sentence Frequency*) (Larocca Neto et al., 2000). Essa medida é similar ao *TF-IDF*, exceto que ela calcula a freqüência de sentença inversa para um grupo em específico, ao invés de calcular para todos os documentos da coleção. A freqüência de sentença inversa de *w*, denotada por *ISF(w)*, é dada pela Fórmula 4, onde |*C*| representa o total de sentenças do grupo e *SF(w)* é o total de sentenças do grupo que contêm *w*.

(4)

$$ISF(w) = 1 + \log (|C| / SF(w))$$

Para que uma palavra seja representativa de um determinado grupo, ela deve ter um alto valor de *TF* e um alto valor de *ISF* (ou *IDF*) e, portanto, um alto valor de *TF-ISF* (ou *TF-IDF*).

Para avaliação do método foram selecionadas aleatoriamente 20 coleções de documentos do corpus. Visando construir um corpus de referência de sentenças similares, cada sentença de uma coleção foi manualmente classificada, isto é, associada a um nome de grupo (daqui a diante, a classificação manual será referenciada por *classes* e o agrupamento automático será referenciado por *grupos*). Para a classificação manual adotou-se o conceito de similaridade proposto por Hatzivassiloglou et al. (1999) para a mesma tarefa de identificação de sentenças semanticamente similares. De acordo com Hatzivassiloglou et al., duas sentenças são semanticamente similares se elas se referem a um mesmo objeto ou evento e i) o objeto realiza a mesma ação em ambas as sentenças, ou ii) é sujeito da mesma descrição. Considere, por exemplo, as sentenças (a), (b) e (c), extraídas do corpus. Apesar de todas as sentenças se referirem a explosão de uma bomba caseira, as sentenças (a) e (b) focam na explosão ocorrida no Ministério Público, enquanto que (c) se refere à explosão ocorrida na Secretaria de Estado da Fazenda. Nesse caso, somente (a) e (b) são consideradas similares.

(a) Uma bomba caseira foi atirada contra a sede do Ministério Público (MP).
(b) Uma bomba caseira foi jogada contra o prédio do Ministério Público, na capital do estado.
(c) Uma bomba caseira atingiu o prédio da Secretaria de Estado da Fazenda, localizado na avenida Rangel Pestana, ao lado do Poupatempo Sé.

O desempenho do método de agrupamento foi avaliado usando as medidas de Precisão, Cobertura e Medida-f, redefinidas no domínio de clustering (vide Funch et al., 2003 e Steinbach et al., 2000).

Seja N o número total de sentenças a serem agrupadas, K o conjunto de classes, C o conjunto de grupos e n_{ij} o número de sentenças da classe $k_i \in K$ que estão presentes no grupo $c_j \in C$. A Precisão e a Cobertura de k_i e c_j, denotada por $P(k_i,c_j)$ e $C(k_i,c_j)$, respectivamente, são dadas pelas fórmulas 5 e 6. A Precisão representa o número de sentenças do grupo c_j que pertence a classe k_i e indica o quão o grupo c_j é homogêneo em relação a classe k_i. Similarmente, a Cobertura é dada pelo total de sentenças da classe k_i que estão presentes no grupo c_j, representando, assim, a completude do grupo c_j em relação à classe k_i. Por fim, a Medida-f mede a qualidade do grupo c_j em descrever a classe k_i , calculando a média harmônica entre a Precisão e a Cobertura.

(5)

$$P(k_i, c_j) = n_{ij} / |c_j|$$

(6)

$$C(k_i, c_j) = n_{ij} / |k_i|$$

(7)

$$F(k_i, c_j) = \frac{(2 * C(k_i, c_j) * P(k_i, c_j))}{C(k_i, c_j) + P(k_i, c_j))}$$

A Medida-f para cada classe de todo o conjunto de dados se baseia no grupo que melhor descreve cada classe k_i, ou seja, no grupo que maximiza o valor de $F(k_i,c_j)$ para todo j. Assim, o valor de Medida-f global de uma solução de agrupamento S, denotado por $F(S)$, é dado pela Fórmula 8. O valor de $F(S)$ varia de 0 (pior) a 1 (melhor).

(8)

$$F(S) = \sum_{k_i \in K} \frac{|k_i|}{N} \max_{c_j \in C} \{F(k_i, c_j)\}$$

Além das medidas de desempenho apresentadas anteriormente, foram usadas ainda duas métricas para avaliar a qualidade dos grupos de sentenças similares obtidos automaticamente. A primeira métrica, chamada Entropia (Steinbach et al., 2000), mede a organização de cada grupo, ou seja, como as várias classes de sentenças estão distribuídas em cada grupo. A solução de agrupamento ideal será aquela na qual todos os grupos contêm sentenças de uma única classe. Nesse caso, o valor de Entropia será 0. O cálculo da Entropia é baseado na distribuição de classes de cada grupo e é exatamente o que é feito pela medida de Precisão. Em outras palavras, a Precisão representa a probabilidade de uma sentença escolhida aleatoriamente de um grupo c_j pertencer a classe k_i. Desse modo, a Entropia de um grupo c_j, denotada por $E(c_j)$, é dada pela Fórmula 9. A Entropia global de uma solução de agrupamento S, denotada por $E(S)$, é dada pela soma das entropias de cada grupo c_j ponderada pelo tamanho do grupo, conforme a Fórmula 10. Quanto menor o valor de $E(S)$, melhor é a solução de agrupamento.

(9)

$$E(c_j) = -\sum_{k_i} P(k_i, c_j) \log P(k_i, c_j)$$

(10)

$$E(S) = \sum_{c_j} \frac{|c_j|}{N} E(c_j)$$

A segunda métrica usada para medir a qualidade dos grupos é a Pureza (Rosell et al., 2004), que mede o quão puro cada grupo de sentença é. Em outras palavras, a Pureza representa o percentual da classe mais freqüente de cada grupo. Assim, a Pureza de um grupo *cj*, denotada por *P(cj)*, é definida pela classe k_i que maximiza a Precisão do grupo *cj* (Fórmula 11). A Pureza global de uma solução de agrupamento S, denotada por $P(S)$, é dada pela soma dos valores de Pureza de cada grupo *cj* ponderada pelo tamanho do grupo (Fórmula 12). O valor de $P(S)$ varia de 0 (pior) a 1 (melhor).

(11)

$$P(c_j) = \max_{k_i} \{P(k_i, c_j)\}$$

(12)

$$P(S) = \sum_{c_j \in C} \frac{|c_j|}{N} P(c_j)$$

A fim de identificar o limiar de similaridade que melhor define o corpus de trabalho, cada função de similaridade foi avaliada com diferentes configurações de limiares, variando de 0,1 a 1, com exceção da função *Word Overlap* que varia de 0,1 a 0,5. A Tabela 1 apresenta os resultados

Tabela 1. Resultados médios obtidos com cada medida de avaliação para diferentes limiares de similaridade

	Similaridade	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TF-IDF	**Entropia**	0,843	0,287	0,096	0,037	0,016	0,005	0,004	0,003	0,002	0,001
	Medida-f	0,603	0,814	**0,886**	**0,860**	0,841	0,828	0,812	0,799	0,775	0,736
	Pureza	0,549	0,808	0,907	**0,934**	**0,945**	0,945	0,942	0,940	0,941	0,938
TF-ISF	**Entropia**	1,759	0,900	0,319	0,101	0,043	0,013	0,004	0,003	0,002	0,002
	Medida-f	0,348	0,603	0,805	**0,864**	0,856	0,843	0,828	0,813	0,798	0,786
	Pureza	0,315	0,564	0,804	0,913	**1,000**	0,950	0,954	0,953	0,952	0,951
Word	**Entropia**	0,572	0,079	0,010	0,000	0,001	–	–	–	–	–
Overlap	**Medida-f**	0,695	**0,860**	0,838	0,809	0,786	–	–	–	–	–
	Pureza	0,654	0,908	**0,946**	0,943	0,941	–	–	–	–	–

obtidos por cada função de similaridade para cada medida de avaliação. No que diz respeito aos modelos *TF-IDF* e *TF-ISF*, os resultados foram gerados usando um centróide de tamanho 15, ou seja, considerando-se as 15 palavras mais importantes de cada grupo no cálculo da similaridade entre uma sentença e um grupo qualquer. O tamanho ideal do centróide foi obtido automaticamente a partir de experimentos com o corpus (Seno and Nunes, 2008b).

De acordo com a Tabela 1, os valores de Entropia melhoram consideravelmente na medida em que se aumenta o limiar de similaridade para todos os casos. O mesmo ocorre para os valores de Medida-f e Pureza, mas até certo ponto. A Medida-f alcança o seu valor máximo com um limiar de 0,2, 0,3 e 0,4 para *Word Overlap*, *TF-IDF* e *TF-ISF*, respectivamente. Em relação à Pureza, os valores melhoram até um limiar de 0,3 para *Word Overlap*, e um limiar de 0,5 para os modelos *TF-IDF* e *TF-ISF*.

Especificamente em relação aos valores de Entropia e de Pureza, esses se justificam pelo fato de que o número de grupos cresce na proporção em que se aumenta o limiar de similaridade, de modo que eles se tornam mais homogêneos, ou seja, a variedade de classes em cada grupo tende a diminuir. Além disso, como há muitas sentenças não similares no corpus, a tendência é de que esses valores melhorem ainda mais, uma vez que muitos grupos contêm somente uma sentença.

Em relação aos valores de Medida-f, apesar da tendência dos grupos de se tornarem mais homogêneos (aumentando a Precisão), à medida que o limiar de similaridade aumenta, torna-se mais difícil identificar sentenças semanticamente equivalentes, mas lexicalmente muito distintas. Dessa forma, os valores de Cobertura tendem a diminuir, prejudicando o desempenho global.

Em termos de bom desempenho do método de agrupamento e qualidade dos grupos de sentenças, o modelo *TF-IDF* com similaridade 0,4 (daqui a diante *TF-IDF-0,4*) se mostrou mais apropriado para o propósito deste trabalho. Além de obter uma Medida-f de 86% (a melhor Medida-f foi de 88,6% (*TF-IDF-0,3*)), ele obteve bons valores de Entropia (isto é, 0,037) e de Pureza (isto é, 93,4%), principalmente se comparado aos valores obtidos pelo *TF-IDF-0,3*, *TF-ISF-0,4* e *Word-Overlap-0,2*. Além do mais, o desvio padrão obtido pelo *TF-IDF-0,4* (0,07 para Medida-f, 0,06 para Pureza e 0,05 para Entropia) foi menor do que o obtido para o *TF-IDF-0,3* (0,08 para Medida-f, 0,07 para Pureza e 0,10 para Entropia), *TF-ISF-0,4* (0,09 para Medida-f, 0,08 para Pureza e 0,09 para Entropia) e *Word-Overlap-0,2* (0,08 para Medida-f, 0,06 para Pureza e 0,07 para Entropia). Portanto, para a construção do corpus de sentenças comparáveis utilizou-se o modelo *TF-IDF-0,4*.

Visando facilitar a formulação das regras de parafraseamento (Seção 3,2), para cada grupo identificado foram obtidas todas as possíveis combinações de pares de sentenças comparáveis, resultando aproximadamente em 670 pares em todo corpus.

3.2 Formulação de Regras de Parafraseamento

Para a formulação das regras de parafraseamento foram selecionados aleatoriamente 30 pares de sentenças comparáveis do corpus. Cada par foi

analisado e um total de 81 paráfrases foram identificadas manualmente em todo conjunto. A definição de paráfrases adotada nessa análise segue aquela proposta por Hoey (1991) em que duas seqüências distintas de palavras são ditas paráfrases se uma delas puder ser substituída pela outra, em um dado contexto, sem alterar significativamente o sentido do texto.

A Tabela 2 mostra alguns exemplos de ocorrência de paráfrases no corpus. Aproximadamente 26% dos casos identificados são paráfrases lexicais (isto é, ocorrem entre palavras), por exemplo, (a), (g) e (h). Os outros 74% das paráfrases são multipalavras (por exemplo, (b), (c), (d), (f) e (j)) ou ocorrem entre uma palavra e um segmento multipalavras (por exemplo, (e), (i)).

a. colisão ⇔ choque
b. tucano Geraldo Alckmin ⇔ candidato tucano Geraldo Alckmin
c. capital russa ⇔ capital da Rússia
d. direção da Câmara ⇔ Mesa Diretora da Câmara
e. acordo ⇔ acordo financeiro
f. mercado moscovita ⇔ mercado Cherskisov de Moscou
g. membro ⇔ integrante
h. arrasou ⇔ venceu
i. grupo ⇔ grupo criminoso
j. liderança do Grupo B ⇔ liderança do Grupo B da Liga
l. não chegaram a obter ⇔ não alcançaram

Tabela 2: Exemplo de paráfrases

27 regras de parafraseamento foram formuladas a partir da análise de corpus. Alguns exemplos de regras são apresentados na Tabela 3 (onde ADJ: adjetivo; ART: artigo; ADV: advérbio; V: verbo; N: substantivo; PRP: preposição; PROP: nome próprio; ?: indica zero ou uma ocorrência; |: indica alternativa (operador ou) e os números indicam as unidades lexicalmente similares). A regra R1 cobre os exemplos (c) e (f) da Tabela 2; R2 cobre os exemplos (e) e (i); R3 cobre o exemplo (b); R4 cobre o exemplo (l) e R5 cobre os exemplos (d) e (j). Para os exemplos (a), (g) e (h) não há regras, uma vez que são paráfrases lexicais. É importante observar que todas as regras preveem ao menos uma ocorrência de palavras similares em ambos os segmentos, conforme indicam os números subscritos em cada regra.

No caso de R5, por exemplo, dois segmentos S_1 e S_2 são considerados paráfrases se S_1 iniciar com um substantivo (N) e uma preposição (PRP), acompanhada ou não de um artigo (ART?), e finalizar com um nome próprio ou um substantivo ($PROP_1|N_1$) e S_2 iniciar com um nome próprio ou um substantivo (PROP|N) e uma preposição (PRP), que pode ser acompanhada ou não de um artigo (ART?), seguido de um outro nome próprio (similar ao de S_1, se existir) ou de um outro substantivo ($PROP_1|N_1$) que, por sua vez, pode ou não ser acompanhado por uma preposição (PRP?), um artigo (ART?) e mais um nome próprio ou substantivo ((PROP|N)?). A paráfrase (d) da Tabela 2, por exemplo, inicia-se com um substantivo (*direção*), seguido de um artigo e uma preposição (*de* + *a* = *da*), e termina com um nome próprio (*Câmara*). A sua correspondente, por sua vez, é iniciada por um nome próprio (*Mesa Diretora*), acompanhado de um artigo mais uma preposição (*de* + *a* = *da*), e finalizado por outro nome próprio (*Câmara*).

R1. N_1 ADJ ; N_1 PROP? PRP ART? PROP
R2. N_1 ; N_1 ADJ
R3. N $PROP_1$; N ADJ $PROP_1$
R4. ADV? V PRP V_1 ; ADV? V_1
R5. N PRP ART? ($PROP_1|N_1$) ; (PROP|N) PRP ART? ($PROP_1|N_1$) PRP? ART? (PROP|N)?

Tabela 3: Exemplo de regras de parafraseamento

3.3 Alinhamento

O alinhador de conceitos comuns é baseado em informações de *part-of-speech* (*POS*) e em relações de dependência sintática fornecidas pelo *parser* Palavras (Bick, 2000). Dessa maneira, as sentenças comparáveis são primeiramente processadas pelo *parser*, de modo a obter todo o conhecimento sintático necessário de entrada para o alinhador (vide Figura 2). Durante o processo de alinhamento, o sistema também faz uso da base de sinônimos Tep[3] (Maziero et al., 2008), desenvolvida no contexto do projeto Wordnet-Br (Dias-da-Silva et al., 2006), de um

[3] Disponível em: http://www.nilc.icmc.usp.br/tep2/download.htm (último acesso em 13/01/2009)

conjunto de regras de parafraseamento (Seção 3.2) e de uma *stoplist,* que permite a identificação das palavras irrelevantes ao alinhamento (vide Subseção 3.3.2). Como saída tem-se um conjunto de alinhamentos que representam as informações em comum entre as sentenças de entrada.

A versão preliminar do alinhador, descrita em Seno and Nunes (2008a), trabalha somente com pares de sentenças comparáveis. No atual sistema, é possível alinhar duas ou mais sentenças de entrada, conforme ilustra a Figura 2.

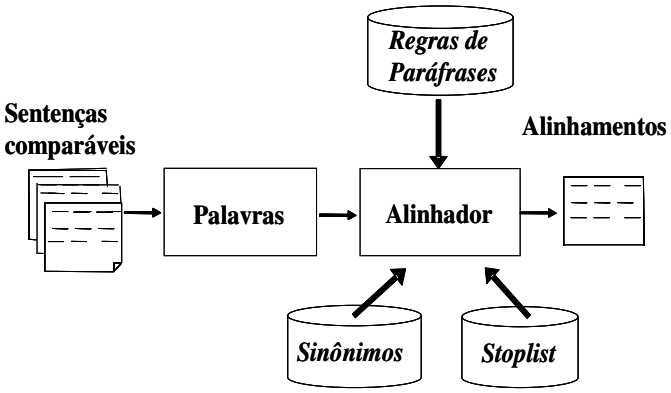

Figura 2: Ilustração do processo de alinhamento

A subseção a seguir descreve a etapa de pré-processamento das sentenças feita pelo Palavras para, então, apresentar o processo de alinhamento propriamente dito na Subseção 3.3.2.

3.3.1 Pré-processamento

O parser Palavras permite análises em diferentes formatos de saída, por exemplo, *Visl* e *TigerXML*, sendo que as informações de dependência sintáticas são obtidas apenas com o formato *Visl* (Bick, 2000). A Figura 3 apresenta um exemplo de análise de dependência sintática realizada pelo *parser* para a sentença "*O Airbus A320, vôo JJ 3054, partiu de Porto Alegre, às 17h16 da terça-feira e chegou a São Paulo às 18h45,*" (sentença [1] da Figura 1). Os traços de dependência se realizam entre *tokens* e incluem relações entre sujeito e verbo, objeto e verbo, etc, No exemplo da figura, *Airbus A320* (*token* #2) é o sujeito (@SUBJ) do verbo (V) *partiu* (*token* #7) e #2->7 indica que o *token* #2 é dependente do *token* #7 (isto é, dependência entre sujeito e verbo)[4]. O *parser* também inclui o processo de lematização (os lemas de cada palavra estão apresentados entre colchetes).

```
O [o] <artd> DET M S @>N #1->2
Airbus=A320 [Airbus=A320] <V> PROP M S
@SUBJ> #2->7
$, #3->0
Vôo [vôo] <activity><np-close> N M S
@N<PRED #4->2
JJ=3054 [JJ=3054] <top> PROP M/F S
@APP #5->4
$, #6->0
partiu [partir] <predco><cjt-
head><fmc> <mv> V PS 3S IND VFIN @FS-
STA #7->0
de [de] PRP @<ADVL  #8->7
Porto=Alegre [Porto=Alegre] <civ> PROP
M S @P< #9->8
$, #10->0
a [a] <sam-> PRP @<ADVL #11->7
as [o] <-sam><artd> DET F P @>N #12-
>13
17h16 [17h16] <temp> N F P @P< #13->11
de [de] <sam-><np-close> PRP @N< #14-
>13
a [o] <artd><-sam> DET F S @>N #15->16
terça-feira [terça-feira] <temp> N F S
@P< #16->14
e [e] <co-fin><co-fmc><co-fin> KC @CO
#17->7
chegou [chegar] <nosubj><cjt><fmc><mv>
V PS 3S IND VFIN @FS-STA #18->7
a [a] PRP @<SA #19->18
São=Paulo [São=Paulo] <civ> PROP M S
@P< #20->19
a [a] <sam-> PRP @<ADVL  #21->18
as [o] <-sam><artd> DET F P @>N #22-
>23
18h45 [18h45] <temp> N F P @P< #23->21
$. #24->0
```

Figura 3: Análise de dependência sintática fornecida pelo Palavras (formato *Visl*)

Apesar de fornecer os traços de dependência entre os constituintes sintáticos, o formato *Visl* não fornece informações sobre os segmentos das sentenças como os sintagmas nominais e os sintagmas verbais, entre outros, Dessa forma, para recuperar as relações de dependência entre sintagmas, na versão preliminar do sistema (Seno and Nunes, 2008a) foram utilizadas algumas expressões regulares definidas com base nos traços de dependência entre os *tokens* (vide Figura 3).

[4] Vide http://beta.visl.sdu.dk/visl/pt/info/portsymbol.html, para maiores informações sobre as etiquetas do Palavras (último acesso em 13/01/2009).

```
...
<terminals>
  <t id="s1_1" word="O" lemma="o" pos="art" morph="M S" sem="--" extra="--"/>
  <t id="s1_2" word="Airbus_A320" lemma="Airbus_A320" pos="prop" morph="M S"
sem="V" extra="--"/>
  <t id="s1_3" word="," lemma="--" pos="pu" morph="--" sem="--" extra="--"/>
  <t id="s1_4" word="vôo" lemma="vôo" pos="n" morph="M S" sem="activity"
extra="np-close"/>
  <t id="s1_5" word="JJ_3054" lemma="JJ_3054" pos="prop" morph="M/F S" sem="--
" extra="top"/>
  <t id="s1_6" word="," lemma="--" pos="pu" morph="--" sem="--" extra="--"/>
  <t id="s1_7" word="partiu" lemma="partir" pos="v-fin" morph="PS 3S IND VFIN"
sem="--" extra="predco predco  fmc mv"/>
   ...
</terminals>
<nonterminals>
  ...
  <nt id="s1_502" cat="np">
     <edge label="DN" idref="s1_1"/>
     <edge label="H" idref="s1_2"/>
     <edge label="DNc" idref="s1_503"/>
  </nt>
  <nt id="s1_503" cat="np">
     <edge label="H" idref="s1_4"/>
     <edge label="DNapp" idref="s1_5"/>
  </nt>
  ...
```

Figura 4: Exemplo de saída do Palavras no formato *TigerXML*

```
...
<tokens>
  <t id="1" word="O" lemma="o" pos="art" morph="M S" sem="--" extra="--"
traco="@>N " dep="2"/>
  <t id="2" word="Airbus_A320" lemma="Airbus_A320" pos="prop" morph="M S"
sem="V" extra="--" traco="@SUBJ> " dep="7"/>
  <t id="3" word="," lemma="--" pos="pu" morph="--" sem="--" extra="--"
traco="--" dep="--"/>
  <t id="4" word="vôo" lemma="vôo" pos="n" morph="M S" sem="activity"
extra="np-close" traco="@N<PRED  " dep="2"/>
  <t id="5" word="JJ_3054" lemma="JJ_3054" pos="prop" morph="M/F S" sem="--"
extra="top" traco="@APP " dep="4"/>
  <t id="6" word="," lemma="--" pos="pu" morph="--" sem="--" extra="--"
traco="--" dep="--"/>
  <t id="7" word="partiu" lemma="partir" pos="v-fin" morph="PS 3S IND VFIN"
sem="--" extra="predco predco  fmc mv" traco="@FS-STA " dep="0"/>
  ...
<phrases>
  <p id="502" phrase="1_2_3_4_5" pos-ph="S"/>
  ...
<dependencies>
  <d id="0" type="S-Verb" son="502" father="7"/>
   ...
```

Figura 5: Formato de entrada atual do alinhador com as relações de dependência entre *phrases*

Na versão atual do sistema, optou-se por modificar o formato dos arquivos de entrada, de modo a representar as relações de dependência entre os sintagmas. O novo formato, ilustrado na Figura 5, foi construído a partir de informações extraídas de duas saídas distintas do *parser* para a mesma sentença de entrada (sentença [1] da Figura 1), São eles: o *Visl* (Figura 3) e o *TigerXML* (Figura 4).

Enquanto o *Visl* fornece os traços de dependência entre os *tokens*, o *TigerXML* fornece as informações sobre os sintagmas das sentenças. No exemplo da Figura 4, o nó não-terminal *s1_502* (`nt id="s1_502"`) é um sintagma nominal (`cat="np"`) composto pelos *tokens* 1 e 2 (`idref="s1_1" e idref="s1_2"`), ou seja, "*o*" e "*Airbus_A320*", e por outro sintagma nominal (`id="s1_503"`) o qual é composto, por sua vez, pelos *tokens* 4 e 5 (`idref="s1_4" e idref="s1_5"`), ou seja, "*vôo*" e "*JJ_3054*". A partir do traço de dependência de cada *token* e da informação sobre qual sintagma ele pertence, é possível obter as relações de dependência entre sintagmas, como mostra o exemplo da Figura 5. Nesse exemplo, o sintagma nominal 502 (`id="502"`), que é composto pelos *tokens* de 1 a 5 (`phrase="1_2_3_4_5"`), ou seja, "*o Airbus_A320 , vôo JJ_3054*", é o sujeito (`pos-ph="S"`) da sentença e estabelece uma relação com o *token* 7 (`son="502" father="7"`), ou seja, "*partiu*", configurando a dependência entre sujeito e verbo (`type="S-Verb"`).

3.3.2 Estratégia de Alinhamento

Dado um conjunto de sentenças comparáveis como entrada (mínimo de duas sentenças), previamente processadas (conforme Figura 5), o algoritmo inicialmente identifica todos os alinhamentos possíveis entre as duas primeiras sentenças do conjunto. Então, as sentenças alinhadas são unidas em uma única estrutura de dependência sintática, denominada floresta. As demais sentenças são alinhadas uma a uma com a floresta e, incrementalmente, também são unidas a ela (isto é, ao término de cada alinhamento entre uma sentença e a floresta). Como resultado final, tem-se uma única estrutura de dependência sintática representando todas as sentenças do conjunto e as intersecções entre elas. A Figura 6 ilustra a floresta construída a partir da união de duas árvores de dependências sintáticas, correspondentes às sentenças "*O Airbus A320, vôo JJ 3054, partiu de Porto Alegre, às 17h16 da terça-feira e chegou a São Paulo às 18h45,"* e "*A aeronave da TAM Airbus A320, vôo JJ 3054, partiu de Porto Alegre, às 17h16 com destino a Congonhas,*" (sentenças [1] e [2] da Figura 1). As setas indicam as dependências entre cada nó terminal e seu nó pai. Por exemplo, o nó terminal *Porto Alegre* (Árvores 1 e 2) é dependente do nó não terminal *partir* e representa uma relação de dependência entre verbo (ver) e objeto (obj). As caixas de textos e as setas não tracejadas representam os nós alinhados, enquanto que as setas tracejadas indicam os nós sem alinhamento.

O alinhamento realizado é do tipo um-para-um, ou seja, cada segmento de uma sentença tem no máximo um segmento correspondente na outra sentença. É válido dizer que o processo de alinhamento descrito neste trabalho difere consideravelmente daquele realizado em outras tarefas do PLN (por exemplo, na Tradução Automática), pois algumas informações não estão presentes em ambas as sentenças, mas em apenas uma delas e, nesses casos, elas não são alinhadas. Além do mais, somente as palavras de classes abertas como os substantivos, os verbos, os advérbios e os adjetivos são alinhados. As palavras de classes fechadas (por exemplo, artigos, preposições e conjunções) participam somente dos alinhamentos envolvendo paráfrases multipalavras (por exemplo, *capital russa* e *capital da Rússia*) e por esse motivo elas foram omitidas da Figura 6.

Algoritmo incremental de alinhamento

Passo 1 (inicial): *Alinhamento de duas sentenças*

Dadas duas sentenças do conjunto de entrada (aqui denominadas de sentença fonte e sentença alvo), o algoritmo tenta encontrar o melhor alinhamento entre segmentos que compartilham a mesma informação semântica. Ao invés de analisar exaustivamente todo o espaço de busca dos alinhamentos possíveis, para cada palavra da sentença fonte, o algoritmo procura por possíveis candidatas ao alinhamento na sentença alvo. Para isso, são usadas como âncoras palavras sinônimas, cognatas ou que possuem o mesmo lema da palavra alvo. Além do mais, as palavras candidatas têm que ter o mesmo *POS* da palavra fonte, de modo a garantir um alinhamento mais confiável. As relações de sinonímia são obtidas

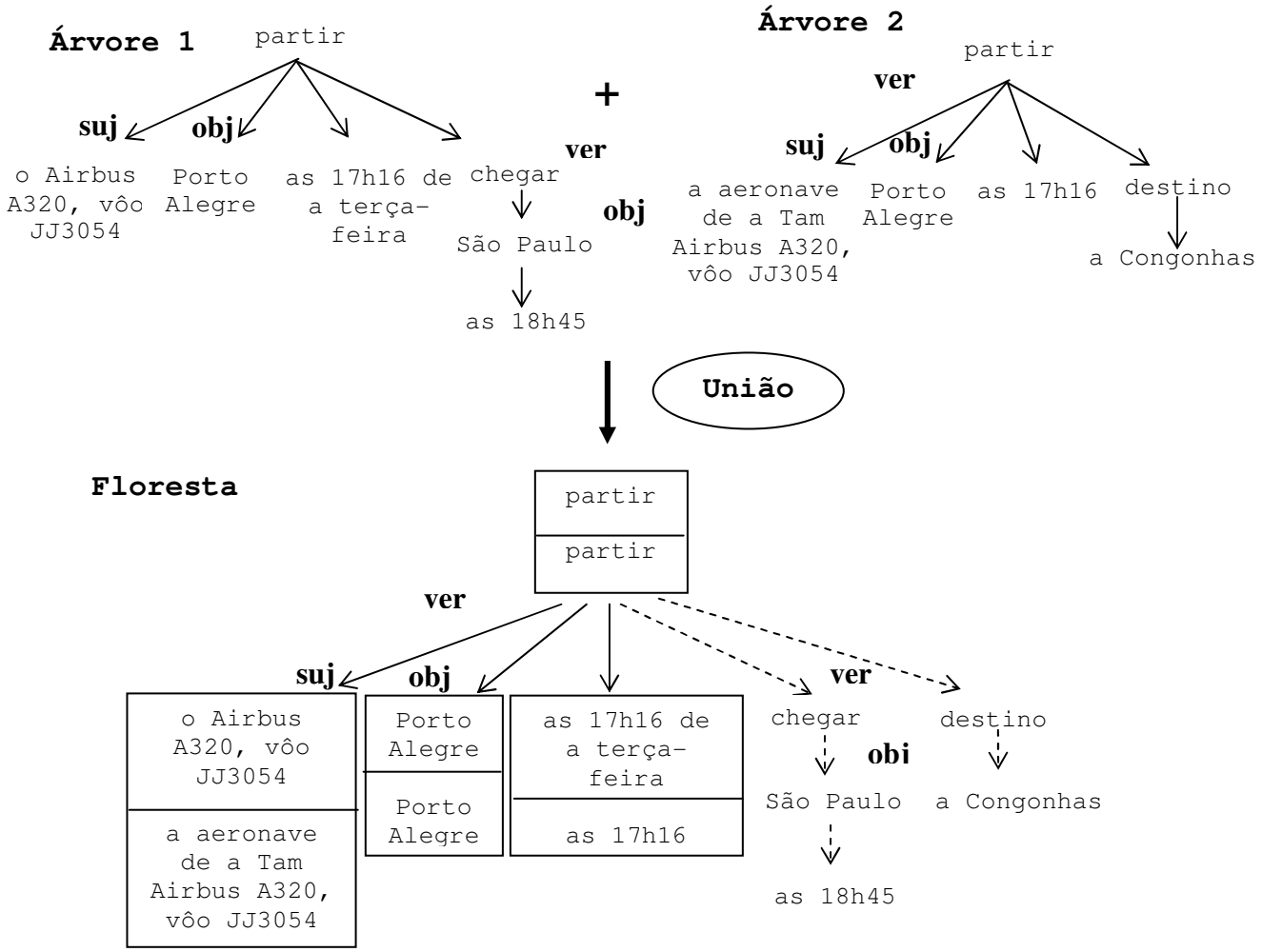

Figura 6: Exemplo de floresta obtida a partir do alinhamento de um par de árvores de dependências sintática

por meio de consultas à base Tep, enquanto que as palavras cognatas são identificadas com o uso de uma medida de similaridade conhecida como LCSR (*Longest Common Subsequence Ratio*, em inglês). O LCSR de duas palavras é calculado dividindo-se o comprimento da maior subseqüência de caracteres em comum entre elas pelo comprimento da maior palavra. Essa medida permite a identificação de palavras com algumas alterações de grafia (por exemplo, *Hezbolla* e *Hisbola*) e também o reconhecimento de diferentes formas de um mesmo nome próprio (por exemplo, *Rui Pimenta* e *Rui Costa Pimenta*). A LCSR só não é usada para os verbos, a fim de evitar casos como *correr* e *morrer* que, apesar do alto valor de LCSR (0,84), têm significados completamente distintos.

Após encontrar todas as candidatas, o algoritmo recupera os sintagmas correspondentes da palavra fonte e de cada palavra candidata, caso a palavra pertença a algum sintagma (por exemplo, *Airbus A320* pertence ao sintagma nominal *o Airbus A320, vôo JJ3054* (sentença [1] da Figura 1)). O sistema então calcula a probabilidade de alinhamento de cada palavra candidata e aquela que apresentar a maior probabilidade é alinhada com a palavra fonte.

Na versão preliminar do sistema (Seno and Nunes (2008a)), a probabilidade de alinhamento é igual a 1, em caso de segmentos idênticos, 0,5 em casos de paráfrases e 0,3 em casos de sinônimos ou cognatos. Esses valores foram determinados empiricamente e priorizam os alinhamentos de palavras e multipalavras literalmente idênticas. Os traços de dependência sintática são considerados somente no alinhamento de verbos. Ou seja, para os casos em que os sujeitos correspondentes aos verbos são similares (isto é, se eles foram previamente alinhados) a probabilidade de alinhamento dos

verbos é acrescida de 0,1, ou penalizada em 0,1, caso contrário. Portanto, nas primeiras iterações, o algoritmo prioriza o alinhamento de nomes próprios e substantivos, visando encontrar as correspondências entre os sujeitos. Por fim, o algoritmo tenta alinhar as palavras e multipalavras restantes ainda não alinhadas, para as quais nenhuma regra de parafraseamento pôde ser aplicada. Esses alinhamentos são realizados somente para os verbos e os sujeitos e se baseiam apenas nos seus traços de dependência sintática, Nos casos em que os sujeitos das sentenças fonte e alvo foram previamente alinhados e os verbos correspondentes ainda não foram alinhados, alinham-se os verbos, assumindo-se que há uma paráfrase entre eles. De maneira similar, se dois verbos foram previamente alinhados e os sujeitos correspondentes nas sentenças não foram, então eles também são alinhados.

No atual sistema, o cálculo da probabilidade de alinhamento foi modificado de modo a considerar não apenas a similaridade entre palavras e multipalavras (isto é, se eles são idênticos, sinônimos, cognatos ou paráfrases), mas também o papel sintático que cada um desempenha na sentença (por exemplo, sujeito, objeto direto, objeto indireto, etc,) e a similaridade entre seus dependentes (para todos os casos, e não somente para os verbos). Nos casos em que a palavra candidata e a palavra fonte têm a mesma função sintática, o sistema adiciona um bônus de 0,3 na probabilidade de alinhamento entre elas. A similaridade entre os dependentes sintáticos é verificada tanto para os verbos, quanto para os sujeitos e objetos das sentenças. Porém, como os verbos são alinhados por último, ao alinhar sujeitos e objetos, o algoritmo verifica se os verbos correspondentes são sinônimos ou paráfrases e, em caso positivo, aumenta a probabilidade de alinhamento em 0,3.

Outra modificação realizada ao sistema diz respeito aos valores de similaridade entre palavras e multipalavras. Para os casos de identidade e de paráfrases, a similaridade é 1, e para os cognatos e sinônimos, a similaridade é 0,5. Esses valores foram ajustados manualmente com base no corpus usado para a identificação das regras de parafraseamento (vide Seção 3.2).

Para que o alinhamento entre duas palavras (ou dois segmentos multipalavras) se concretize, a probabilidade máxima deve ser maior ou igual a 0,5. Esse limite foi estabelecido de modo a permitir também o alinhamento de segmentos que têm funções sintáticas e dependentes em comum, mas para os quais nenhuma regra de parafraseamento pôde ser aplicada.

Passo 2 (incremental): *Alinhamento entre uma sentença e a floresta*

O alinhamento entre uma sentença qualquer e a floresta é realizado de maneira similar ao alinhamento de duas sentenças. Assim, para cada palavra de uma sentença fonte, o algoritmo procura por possíveis candidatas ao alinhamento na floresta. A floresta é armazenada em um vetor associativo cujas chaves correspondem ao identificador de cada sentença do conjunto já alinhada à ela. Para cada chave de uma sentença, é mantido outro vetor associativo contendo cada palavra da sentença e, para cada palavra, por sua vez, são guardadas informações sobre o sintagma ao qual pertence e sobre o alinhamento, ou seja, as palavras (ou sintagmas) de outras sentenças que estão alinhadas a ela, em caso da palavra já ter sido alinhada anteriormente. Desse modo, a palavra fonte é comparada a cada palavra de uma sentença da floresta. Ao encontrar possíveis candidatas ao alinhamento, o algoritmo recupera os sintagmas correspondentes a cada uma delas (se houver) e, então, calcula a probabilidade de a probabilidade de alinhamento, conforme descrito anteriormente (Passo 1). Caso haja alguma candidata com probabilidade >= 0,5, ela é alinhada à palavra fonte (e a todas as outras que já foram previamente alinhadas a ela, se existir alguma) e a busca por novas candidatas é finalizada. Caso contrário, a busca procede na próxima sentença da floresta.

Para fins de ilustração, considere o alinhamento entre a floresta apresentada na Figura 6 e a sentença "*Um Airbus A320 com capacidade para 170 passageiros partiu de Porto Alegre (RS) às 17h16 com destino a Congonhas,*" (sentença [3] da Figura 1). Ao buscar na floresta possíveis candidatos ao alinhamento para o nome "*Airbus A320",* por exemplo, o algoritmo inicialmente analisa todas as palavras de uma determinada sentença da floresta. As sentenças são ordenadas de acordo com seu identificador, isto é, sua chave no vetor associativo, e são selecionadas em ordem.

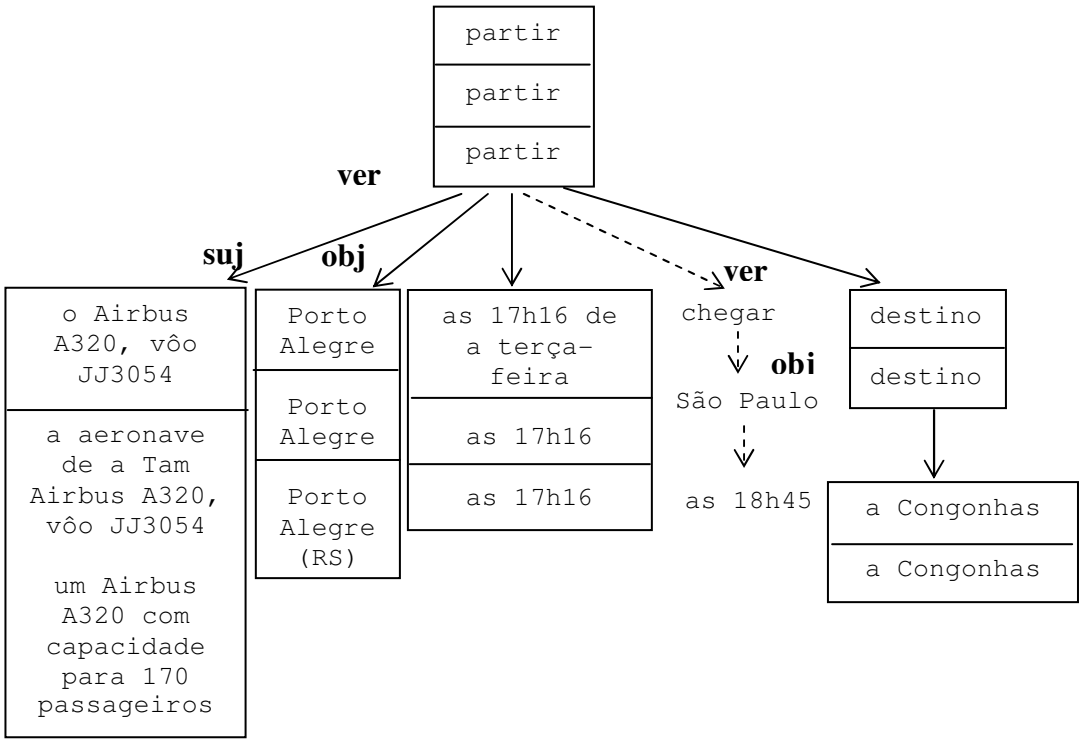

Figura 7: Exemplo de floresta obtida a partir do alinhamento de 3 árvores de dependências sintáticas

Supondo que a sentença da floresta em foco seja "*A aeronave da TAM Airbus A320, vôo JJ 3054, partiu de Porto Alegre, às 17h16 com destino a Congonhas,*", somente um nome candidato será encontrado, "*Airbus A320*". O algoritmo então recupera os sintagmas correspondentes da sentença fonte e da sentença da floresta, ou seja, "*um Airbus A320 com capacidade para 170 passageiros*" e "*a aeronave da TAM Airbus A320, vôo JJ 3054*", respectivamente. Após recuperar os sintagmas, o sistema calcula a probabilidade de alinhá-los. Para esse exemplo, em particular, não há regras de parafraseamento. Desse modo, a probabilidade de alinhamento é igual a 0,6, uma vez que ambos os segmentos desempenham o papel de sujeito e os verbos correspondentes (*partir*) são similares. Portanto, eles são alinhados e a busca por novos candidatos em outras sentenças da floresta é finalizada. Como o sintagma da floresta ("*a aeronave da TAM Airbus A320, vôo JJ 3054*") já havia sido alinhado a outro sintagma ("*o Airbus A320, vôo JJ 3054*") (vide Figura 6), o novo correspondente "*um Airbus A320 com capacidade para 170 passageiros*" é adicionado ao mesmo alinhamento. A Figura 7 ilustra a floresta resultante do alinhamento entre a sentença [3] (Figura 1) e a floresta da Figura 6.

4. Experimentos

Com o propósito de verificar se as mudanças no pré-processamento das sentenças de entrada e na estratégia de alinhamento de fato contribuem para um melhor desempenho do sistema, foram avaliados somente os alinhamentos produzidos entre pares de sentenças comparáveis (e não a partir de um conjunto de sentenças). Uma vez que o alinhamento entre uma sentença qualquer e a floresta é similar ao alinhamento de um par de sentenças (vide Seção 3.3.2), acredita-se que o desempenho do sistema tanto no alinhamento de duas sentenças como no alinhamento de um conjunto de sentenças será equivalente.

A qualidade dos alinhamentos automáticos foi verificada com base em um corpus de referência composto por 20 pares de sentenças extraídos aleatoriamente do corpus comparável (Seção 3.1). É válido dizer que esse subcorpus é diferente daquele usado para a formulação das regras de parafraseamento.

Os 20 pares de sentenças foram manualmente alinhados por dois anotadores. Posteriormente, a concordância entre eles foi calculada com base no total de alinhamentos em comum dividido pelo total de alinhamentos produzidos pelos dois anotadores. Uma taxa de concordância de 87% foi obtida, indicando que os alinhamentos de referência são razoavelmente confiáveis.

Para a avaliação do sistema, foram usadas as medidas de Precisão, Cobertura e Medida-f. Seja *R* o conjunto de alinhamentos de referência, *A* o conjunto de alinhamentos produzidos automaticamente e $|A \cap R|$ o conjunto de alinhamentos automáticos corretamente produzidos. A Precisão representa a fração dos alinhamentos automáticos identificados corretamente, em relação a todos os alinhamentos automáticos produzidos (Fórmula 13). A Cobertura representa a fração dos alinhamentos automáticos identificados corretamente, em relação a todos os alinhamentos previstos no conjunto de referência (Fórmula 14). A Medida-f, por sua vez, representa a média harmônica entre a Precisão e a Cobertura (Fórmula 15).

$$\text{Precisão} = \frac{|A \cap R|}{|A|} \quad (13)$$

$$\text{Cobertura} = \frac{|A \cap R|}{|R|} \quad (14)$$

$$\text{Medida-f} = \frac{2 \times \textit{Precisão} \times \textit{Cobertura}}{\textit{Precisão} + \textit{Cobertura}} \quad (15)$$

O sistema proposto foi comparado com outros dois sistemas *baselines*. O *baseline* 1, que é baseado somente na similaridade lexical e semântica, alinha apenas segmentos idênticos, cognatos e sinônimos. O *baseline* 2 é uma extensão do *baseline* 1 que inclui, além dos sinônimos e cognatos, os traços de dependência sintática. O primeiro *baseline* tem como propósito avaliar a contribuição das regras de parafraseamento e das relações de dependência sintática para o processo de alinhamento, enquanto que o *baseline* 2 visa apenas verificar a contribuição das regras de parafraseamento.

A Tabela 4 apresenta os valores médios obtidos pelo alinhador proposto (versão 2,0) e por cada *baseline* para Precisão, Cobertura e Medida-f. Para fins de comparação, a tabela também resume os resultados obtidos com a versão preliminar do sistema (versão 1,0), apresentados em Seno and Nunes (2008a). Os *baselines* usados na versão 1,0 são equivalentes aos *baselines* descritos neste trabalho.

Sistema	Precisão	Cobertura	Medida-f
Versão 2,0			
Baseline 1	0,81	0,76	0,78
Baseline 2	0,81	0,75	0,78
Alinhador Proposto	0,87	0,83	0,85
Versão 1,0			
Baseline 1	0,77	0,72	0,74
Baseline 2	0,77	0,72	0,74
Alinhador Proposto	0,86	0,81	0,83

Tabela 4: Resultados do alinhamento automático obtidos para Precisão, Cobertura e Medida-f

Conforme os resultados apresentados na Tabela 4, o atual sistema obteve uma melhora de 2,4% no desempenho global em relação à sua primeira versão (isto é, 85% de Medida-f contra 83% de Medida-f) e um ganho de 9% comparado aos seus *baselines*. Os *baselines*, por sua vez, já obtiveram um desempenho bem elevado (isto é, 78% de Medida-f), o que era esperado devido às características do corpus (aproximadamente 72% dos alinhamentos identificados ocorrem entre segmentos literalmente idênticos).

É importante observar que os *baselines* atuais também apresentaram um desempenho de cerca de 5% melhor em relação aos *baselines* usados na avaliação do sistema anterior (ou seja, 78% de Medida-f contra 74% de Medida-f). Isso se deve principalmente às modificações no pré-processamento das sentenças que permitem recuperar de forma mais abrangente e confiável as dependências sintáticas entre os sintagmas.

Outro ponto importante a ser notado é que o *baseline* 2 não apresentou ganho de desempenho comparado ao *baseline* 1 (em ambas as versões), quando foram incluídos os traços de dependência entre os constituintes sintáticos. O ganho de desempenho apenas foi verificado ao se incluir

as regras de parafraseamento nos sistemas propostos (conforme mostrado na Tabela 4).

Com propósito de verificar a contribuição do sistema proposto para o alinhamento de paráfrases apenas (tanto lexicais, isto é, sinônimos e cognatos, quanto sintáticas), a Precisão, a Cobertura e a Medida-f foram calculadas considerando-se somente esses casos. Os resultados obtidos são mostrados na Tabela 5. Para fins de comparação, os resultados alcançados com a versão 1,0 do sistema também são mostrados na tabela.

De acordo com a Tabela 5, a segunda versão do alinhador apresentou um ganho de aproximadamente 21% em comparação a sua primeira versão (ou seja, 64% de Medida-f contra 53% de Medida-f). É válido notar que o ganho de Precisão e de Cobertura foi de 9,5% e 33,3%, respectivamente. Além do mais, o sistema obteve uma melhora substancial de desempenho em relação aos *baselines* (isto é, um aumento de 94% e de 178% comparado ao *baseline* 2 e ao *baseline* 1, respectivamente), quando considerados apenas os casos de paráfrases.

Sistema	Precisão	Cobertura	Medida-f
Versão 2,0			
Baseline 1	0,55	0,14	0,23
Baseline 2	0,53	0,24	0,33
Alinhador Proposto	0,69	0,60	0,64
Versão 1,0			
Baseline 1	0,63	0,12	0,20
Baseline 2	0,50	0,17	0,25
Alinhador Proposto	0,63	0,45	0,53

Tabela 5: Resultados do alinhamento automático obtidos para Precisão, Cobertura e Medida-f, considerando-se somente os casos de paráfrases

O uso das relações de dependência sintática no *baseline* 2 (versão 2,0) contribuiu para um aumento de cerca de 43% no desempenho global, em relação ao *baseline* 1 (sem relações de dependências), quando considerados apenas os alinhamentos de paráfrases. No entanto, como dito anterior, nenhuma melhora foi observada entre os *baselines* 1 e 2, quando considerados todos alinhamentos em ambas as versões dos sistemas (vide Tabela 4).

Esses resultados comprovam que a similaridade lexical, as relações de sinonímia e as relações sintáticas auxiliam no alinhamento de informações comuns, porém não são suficientes para tratar os casos mais complexos de paráfrases como é o caso das paráfrases sintáticas, parcialmente tratadas pelas regras de parafraseamento.

A Figura 8 mostra alguns exemplos de alinhamentos produzidos pelo algoritmo. A maioria deles foi identificado com o auxílio das regras de parafraseamento, como os exemplos (a), (b), (c), (d), (e), (f), (h) e (i). Alguns casos de paráfrases que não foram cobertos pelas regras são ilustrados na Figura 9.

(a) 44% das intenções de voto ⇔ 44% dos votos
(b) março ⇔ março de o ano que vem
(c) a agência Itar-Tass ⇔ a agência oficial russa Itar-Tass
(d) Luiz Inácio Lula da Silva ⇔ o presidente Luiz Inácio da Silva ⇔ Lula
(e) a cidade de Tampere ⇔ Tampere (FIN)
(f) o chefe de polícia do campus ⇔ o chefe de polícia da universidade
(g) afirmou ⇔ disse
(h) aconteceu ⇔ foi registrada
(i) bujão de gás ⇔ botijão de gás

Figura 8: Exemplos de alinhamentos automáticos

(a) os 69 deputados acusados pela CPI dos Sanguessugas de envolvimento ⇔ os deputados envolvidos
(b) os quatro menores ⇔ os quatro com menos de 18 anos
(c) o prédio de carga e descarga da companhia aérea ⇔ o prédio da TAM Express
(d) 23 pessoas ⇔ o grupo

Figura 9: Exemplos de paráfrases não identificadas pelas regras de parafraseamento

5. Conclusões e Trabalhos Futuros

Este trabalho apresentou uma nova versão do alinhador descrito em Seno and Nunes (2008a), para a identificação de segmentos que conduzem a mesma informação semântica entre sentenças comparáveis do português.

Diversas melhorias realizadas ao sistema, como alterações no pré-processamento das sentenças de entrada, modificações na estratégia de alinhamento e a inclusão de novas relações sintáticas, resultaram em um aumento de desempenho de aproximadamente 21%, comparado com a primeira versão do sistema, quando avaliados somente os alinhamentos entre paráfrases (tanto lexical, quanto sintática). Quando considerados todos os alinhamentos (incluindo os casos de segmentos literalmente idênticos), o ganho no desempenho foi de 2,4%,

O resultado alcançado neste trabalho, ou seja, um desempenho de 85% de Medida-f considerando todos os alinhamentos, representa um ganho de 9% em relação aos *baselines* de comparação e está de acordo com outros resultados reportados na literatura (vide Seção 2).

Com relação ao alinhamento de paráfrases somente (isto é, excluindo-se os casos de segmentos idênticos), o método apresentou um ganho de até 178% no desempenho global, comparado aos *baselines*. Os trabalhos encontrados na literatura não reportam resultados para os casos de paráfrases apenas.

Os experimentos apresentados na seção anterior são preliminares e se referem apenas aos alinhamentos produzidos a partir de pares de sentenças. Entretanto, como a estratégia de alinhamento é independente do número de sentenças de entrada, acredita-se que o sistema obterá um desempenho similar no alinhamento de um conjunto de sentenças. Novos experimentos deverão ser realizados para comprovar essa hipótese. Além disso, estão previstos experimentos com corpora maiores e a indução automática de paráfrases a partir de corpus.

É importante notar que o alinhador foi projetado para trabalhar com sentenças semanticamente muito similares (ou seja, comparáveis ou paralelas monolíngües). Portanto, é natural que haja uma queda de desempenho do sistema ao tentar alinhar sentenças com pouca similaridade semântica.

Como continuação deste trabalho, os próximos passos incluem a implementação de um módulo de fusão e linearização, para a geração de novas sentenças a partir da fusão de informações comuns previamente alinhadas. Esse módulo já está em desenvolvimento atualmente e poderá ser usado em um futuro próximo para validar o processo de alinhamento de informações comuns, inclusive no que se refere ao alinhamento envolvendo mais de duas sentenças.

Agradecimento

Agradecemos ao CNPq (Conselho Nacional de Pesquisa e Desenvolvimento) pelo suporte financeiro.

Referências

Barzilay, R. 2003. *Information Fusion for Multidocument Summarization: Paraphrasing and Generation*, Phd, Thesis, Columbia University, New York, 221 p.

Barzilay, R, and McKeown, K. 2005. Sentence Fusion for Multi-document News Summarization, *Computational Linguistics*, Vol, 31, nº 3, pp, 297-327.

Bick, Eckhard. 2000. *The Parsing System "Palavras" - Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*, Aarhus University Press.

Dias-da-Silva, B.C., Di Felippo, A., and Hasegawa, R. 2006. Methods and Tools for Encoding the WordNet,Br Sentences, Concept Glosses and Conceptual-Semantic Relations. In: *Proceedings of the 7th Workshop on Computational Processing of the Portuguese Language - Written and Spoken -*

PROPOR (Lecture Notes in Artificial Intelligence, 3960), pp, 120-130.

Fung, B.C.M., Wang, K., Ester, M. 2003. Hierarchical Document Clustering using Frequent Itemsets. In: Barbará, D, Kamath, C, eds, *3rd SIAM International Conference on Data Mining*, pp, 59-70.

Hatzivassiloglou, V., Klavans, J. L., Eskin, E. 1999. Detecting Text Similarity over Short Passages: Exploring Linguistic Feature Combinations via Machine Learning. In: *Proceedings of the Empirical Methods in Natural Language Processing and Very Large Corpora – EMNLP*, pp, 203-212.

Hoey, M. 1991. *Patterns of Lexis in Text*, Oxford: Oxford University Press,

Krahmer, E., Marsi, E. and van Pelt, P. 2008. Query-based sentence fusion is better defined and leads to more preferred results than generic sentence fusion, In: *Proceedings of the Human Language Technology Conference – HLT/ACL*, pp, 193-196.

Larocca Neto, J., Santos, A.D., Kaestner, C.A.A., Freitas, A.A. 2000. Document Clustering and Text Summarization. In: *4th International Conference Practical Applications of Knowledge Discovery and Data Mining – PAAD*, pp, 41-55.

Marsi, E. and Krahmer, E. 2005. Explorations in Sentence Fusion. In: *Proceedings of the 10th European Workshop on Natural Language Generation – ENLG*, pp, 109-117.

Maziero, E.G., Pardo, T.A.S., Di Felippo, A., Dias-da-Silva, B.C. 2008. A Base de Dados Lexical e a Interface Web do TeP 2,0 - Thesaurus Eletrônico para o Português do Brasil. *VI Workshop em Tecnologia da Informação e da Linguagem Humana (TIL)*, pp, 390-392.

Pang, B., Knight, K. and Marcu, D. 2003. Syntax-based Alignment of Multiple Translations: Extracting Paraphrases and Generating New Sentences. In: *Proceedings of the Human Language Technology Conference – HLT/NAACL*, pp, 102-109.

Radev, D., Otterbacher, J., Zhang, Zhu. 2008. Cross-document Relationship Classification for Text Summarization. Disponível em: tangra,si,umich,edu/~radev/papers/progress/p1,ps (último acesso: 13/04/2009).

Rosell, M., Kann, V., Litton, J. 2004. Comparing Comparisons: Document Clustering Evaluation Using Two Manual Classifications. In: Sangal R, Bendre SM, eds, *International Conference on Natural Language Processing*, Allied Publishers Private Limited, pp, 207-216.

Salton, G. and Allan, J. 1994. Text Retrieval Using the Vector Processing Model. In: *Proceedings of the 3rd Symposium on Document Analysis and Information Retrieval*, University of Nevada, Las Vegas.

Seno, E.R.M. and Nunes, M.G.V. 2008a. Automatic Alignment of Common Information in Comparable Sentences of Portuguese. In: *Anais do VI Workshop em Tecnologia da Informação e da Linguagem Humana – TIL*, pp, 331-335.

Seno, E.R.M. and Nunes, M.G.V. 2008b. Some Experiments on Clustering Similar Sentences of Texts in Brazilian Portuguese. In: *Proceedings of the International Conference on Computational Processing of Portuguese Language - PROPOR* (Lecture Notes in Artificial Intelligence, 5190), pp, 133-144.

Shen, S., Radev, D. R., Patel, A. and Erkan, G. 2006. Adding Syntax to Dynamic Programming for Aligning Comparable Texts for the Generation of Paraphrases. In: *Proceedings of the COLING/ACL*, pp, 747-754.

Steinbach, M., Karypis, G., Kumar, V. 2000. A Comparison of Document Clustering Techniques. In: *International Conference on Knowledge Discovery & Data Mining – KDD*.

Van Rijsbergen, C.J. 1979. *Information Retrieval*, 2nd edition, Butterworths, Massachusetts.