UNIVERSIDADE ESTADUAL DE MARINGÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

LEANDRO LAGO DA SILVA

Análise automática de coerência textual em resumos científicos: avaliando quebras de linearidade

Maringá
2016

LEANDRO LAGO DA SILVA

Análise automática de coerência textual em resumos científicos: avaliando quebras de linearidade

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Informática, Centro de Tecnologia da Universidade Estadual de Maringá, como requisito parcial para obtenção do título de Mestre em Ciência da Computação

Orientadora: Profa. Dra. Valéria Delisandra Feltrim

Maringá
2016

Dados Internacionais de Catalogação na Publicação (CIP)
(Biblioteca Central - UEM, Maringá, PR, Brasil)

S586a

Silva, Leandro Lago da
Análise automática de coerência textual em resumos científicos: avaliando quebras de linearidade / Leandro Lago da Silva -- Maringá, 2016.
88 f. : il., color., figs., tabs., mapas.

Orientador: Profª. Drª. Valéria Delisandra Feltrim.
Dissertação (mestrado) - Universidade Estadual de Maringá, Centro de Tecnologia, Departamento de Informática, Programa de Pós-Graduação em Ciência da Computação, 2016.

1. Coerência. 2. Quebra de linearidade. 3. Grade de entidades. 4. Auxílio à escrita científica. SciPo. I. Feltrim, Valéria Delisandra, orient. II. Universidade Estadual de Maringá. Centro de Tecnologia. Departamento de Informática. Programa de Pós-Graduação em Ciência da Computação. III. Título.

CDD 21.ed. 006.35

AHS-002862

FOLHA DE APROVAÇÃO

LEANDRO LAGO DA SILVA

Análise automática de coerência textual em resumos científicos: avaliando quebras de linearidade

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Informática, Centro de Tecnologia da Universidade Estadual de Maringá, como requisito parcial para obtenção do título de Mestre em Ciência da Computação pela Banca Examinadora composta pelos membros:

BANCA EXAMINADORA

Profa. Dra. Valéria Delisandra Feltrim
Universidade Estadual de Maringá – DIN/UEM

Profa. Dra. Luciana Andréia Fondazzi Martimiano
Universidade Estadual de Maringá – DIN/UEM

Prof. Dr. Thiago Alexandre Salgueiro Pardo
Universidade de São Paulo – ICMC/USP

Aprovada em: 22 de fevereiro de 2016.
Local da defesa: Sala 120, Bloco C56, *campus* da Universidade Estadual de Maringá.

DEDICATÓRIA

À minha família, pelo incentivo e apoio em todas as minhas escolhas e decisões.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por ter me concedido saúde para poder realizar este trabalho. À minha esposa pelo amor, carinho, motivação, auxílio e compreensão nas horas em que não pude estar com ela. Agradeço também aos meus pais e familiares pelo incentivo e apoio em todos os momentos. À professora Valéria por me apresentar a área de pesquisa, demonstrando um enorme conhecimento e disposição em ensinar, pelas recomendações e dedicação empregada nas orientações e pelas inúmeras oportunidades concedidas diante dos meus erros. Aos colegas, amizades que jamais serão esquecidas. Ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), pelo auxílio financeiro que possibilitou a realização deste trabalho e à Universidade Estadual de Maringá e ao Departamento de Informática que permitiram a realização desta pesquisa.

Análise automática de coerência textual em resumos científicos: avaliando quebras de linearidade

RESUMO

A coerência faz com que uma sequência de palavras, sentenças ou parágrafos, se torne um texto, estabelecendo relações entre os elementos e dando sentido ao discurso. Escrever textos coerentes é uma tarefa que requer prática e habilidade em diversos aspectos linguísticos. Um método para se conseguir essas habilidades é solicitar auxílio para revisores ou ferramentas computacionais específicas para esse fim. O *Scientific Portuguese* - SciPo é um exemplo de ferramenta de auxílio à escrita para a língua portuguesa que, dentre outros recursos, inclui um módulo de análise de coerência (MAC) que detecta potenciais problemas de coerência semântica em resumos científicos. Baseado em análise de semântica latente (LSA), o MAC analisa o relacionamento semântico entre as sentenças de um resumo, de acordo com um conjunto pré-estabelecido de dimensões. Para uma das dimensões propostas para o MAC, chamada Quebra de Linearidade, os resultados obtidos por meio de LSA foram pouco satisfatórios, sugerindo a utilização de outros modelos de coerência. Nesse contexto, este trabalho teve por objetivo estender o MAC, pela adição da dimensão chamada Quebra de Linearidade. A abordagem proposta para essa dimensão é baseada na combinação do modelo grade de entidades com informações provenientes da estrutura retórica do resumo, permitindo que o módulo faça sugestões indicando possíveis quebras de linearidade em regiões específicas do resumo. Os experimentos realizados mostraram que a combinação proposta consegue capturar quebras de linearidade e também confirmaram que as sugestões geradas para essa dimensão são úteis, guiando os usuários na construção de um texto com um maior nível de coerência.

Palavras-chave: Coerência, Quebra de linearidade, Grade de entidades, Auxílio à escrita científica, SciPo

Automatic analysis of textual coherence in scientific abstracts: evaluating linearity breaks

ABSTRACT

Coherence makes a sequence of words, sentences or paragraphs, become a text, connecting the elements and giving meaning to the speech. To write coherent texts is a task that requires practice and skill in various linguistic aspects. One way to achieve these skills is to request aid for reviewers or for computational tools developed for this purpose. The Scientific Portuguese - SciPo is an example of writing tool for Portuguese that includes, among other features, a coherence analysis module (MAC) which detects potential problems of semantic coherence in scientific abstracts. Based on latent semantic analysis (LSA), MAC analyzes the semantic relationship between sentences of the abstract, in accordance with a predetermined set of dimensions. For one of the proposed dimensions for the MAC, called Linearity Break, the results obtained by LSA were unsatisfactory, suggesting the use of other coherence models. In this context, this project aimed at extending MAC by adding the Linearity Break dimension. The proposed approach for it combines the entity grid model with information from the abstract rhetorical structure, allowing MAC to generate suggestions pointing possible breaks linearity in specific regions of the abstract. Experimental results have shown that the proposed combination captures linearity breaks, and confirmed that the generated suggestions are useful, guiding users in writing texts with a higher level of coherence.

***Keywords*:** Coherence, Linearity Break, Entity Grid, scientific writing support, SciPo

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AMADEUS	*Amiable Article Development for User Support*
AZPort	*Argumentative Zoning Portuguese*
DIN	Departamento de Informática
DC	Departamento de Computação
EDU	*Elementary Discourse Units*
IEA	*Intelligent Essay Assessor*
INF	Departamento de Informática
LSA	*Latent Semantic Analysis*
MAC	Módulo de Análise de Coerência
PLN	Processamento de linguagem natural
SciPo	*Scientific Portuguese*
SNs	Sintagmas Nominais
SVD	*Singular Value Decomposition*
TCCs	Trabalhos de Conclusão de Curso
TF-IDF	Term Frequency – Inverse Document Frequency
UEM	Universidade Estadual de Maringá
UEL	Universidade Estadual de Londrina
UFPel	Universidade Federal de Pelotas
WEKA	*Waikato Environment for Knowledge Analysis*
W3C	*World Wide Consortium*
XML	*Extensible Markup Language*

SUMÁRIO

1. Introdução

Na linguística a coerência está intimamente relacionada ao processo de produção e compreensão do texto e, por isso, é uma das principais responsáveis para se alcançar a textualidade, fazendo com que o texto não seja um amontoado de palavras ou frases sem sentido (Koch e Travaglia, 2008). Segundo os autores, a coerência está diretamente ligada à possibilidade de estabelecer um sentido para o texto, sendo assim um princípio de interpretabilidade, ligada à inteligibilidade do texto numa situação de comunicação e à capacidade que o receptor tem para calcular o sentido desse texto.

A produção de bons textos é uma tarefa que requer habilidade do escritor em diversos aspectos como, o domínio do assunto abordado, o uso correto de recursos lexicais, gramaticais, estilísticos, coesão e coerência textual. Burstein *et al.* (2003) afirmam que a melhor maneira de se aprimorar as habilidades de um escritor é praticar ciclicamente os processos: escrita, revisão por um especialista e a correção dos pontos sugeridos. A grande dificuldade desses processos está em encontrar um revisor disponível sempre que for necessário.

Visando auxiliar professores e revisores de texto, assim como, iniciantes na escrita, pesquisadores têm buscado desenvolver aplicações que automatizem a avaliação de textos. Uma categoria dessas aplicações, que tem utilizado métodos de avaliação de coerência, é a das ferramentas de auxílio à escrita, em especial àquelas com propósito educacional.

Para a língua inglesa existem sistemas que consideram aspectos de coerência no processo de avaliação de redações, por exemplo, as ferramentas *Criterion* (Burstein et al., 2003), *Intelligent Essay Assessor* (Foltzet al., 2013) e *IntelliMetric* (Schultz, 2013). O auxílio fornecido por essas ferramentas ao usuário é dado por meio de críticas e sugestões referentes a vários aspectos linguísticos, incluindo desenvolvimento, organização, coerência, gramática, ortografia e estilo. Além disso, esses sistemas são classificados como *scoring system*, pois após a análise eles retornam uma nota a respeito da qualidade do texto, com base na avaliação de um conjunto de aspectos linguísticos.

Para a língua portuguesa existe a ferramenta de auxílio à escrita *SciPo - Scientific Portuguese* (Feltrim et al., 2006), que foi desenvolvida para ajudar escritores iniciantes na produção de textos científicos, em especial na escrita de resumos e introduções na área da Ciência da Computação. Além de outros recursos disponíveis, a ferramenta tem um módulo de análise de coerência (MAC) que detecta potenciais problemas de coerência textual em resumos. O MAC (Souza e Feltrim, 2013; Souza, 2011) é baseado na classificação de componentes retóricos e na similaridade semântica entre componentes medida por meio de Análise de Semântica Latente (LSA) (Landauer et al., 1998), que é um método para a aquisição e representação de conhecimento sobre o significado dos termos e documentos analisados. Especificamente, três tipos de relacionamentos semânticos, chamados de dimensões, são examinados: (1) a Dimensão Título verifica o relacionamento semântico entre o título do resumo e o componente Propósito; (2) a Dimensão Propósito verifica o relacionamento semântico entre o componente Propósito e os componentes Metodologia, Resultado e Conclusão; e (3) a Dimensão Lacuna-Contexto verifica o relacionamento semântico entre os componentes Lacuna e Contexto. O baixo relacionamento semântico detectado em qualquer uma das dimensões é interpretado pelo MAC como um indicativo de possíveis problemas na coerência no resumo, gerando assim sugestões de melhorias que são fornecidas ao usuário.

Ainda no contexto do MAC, uma quarta dimensão, chamada Quebra de Linearidade, foi proposta por Souza (2011), mas não chegou a ser automatizada. Essa dimensão busca identificar problemas de coerência local que se caracterizam pela dificuldade em se estabelecer uma ligação clara da sentença atual com a sentença anterior, demandando assim um maior esforço cognitivo para a interpretação do texto. Segundo o autor, os resultados obtidos em experimentos com LSA para essa dimensão foram pouco satisfatórios, sugerindo assim o uso de outros modelos de coerência, como o modelo de grade de entidades (Barzilay e Lapata, 2008).

O trabalho de Freitas (2013) propôs uma versão da grade de entidades para o português e mostrou que o modelo consegue capturar problemas locais de coerência como as quebras de linearidade. Para isso, o autor realizou experimentos com um *corpus* de resumos científicos contendo textos anotados manualmente. Os resultados obtidos por Freitas (2013) foram próximos dos resultados relatados por trabalhos relacionados para outras línguas (Filippova e Strube, 2007; Yokono e Okumura, 2010; Burstein et al., 2010) e também se aproximaram dos resultados obtidos por juízes humanos (Freitas, 2013), mostrando seu potencial de aplicação no contexto do MAC.

No entanto, a aplicação de um modelo como a grade de entidades no MAC não é direta, visto que é preciso gerar mensagens referentes à dimensão que sejam úteis aos usuários da

ferramenta SciPo. No contexto de uma ferramenta de auxílio à escrita voltada para escritores iniciantes, uma mensagem útil deve informar o usuário a respeito de possíveis quebras de linearidade, bem como dar uma indicação da localização dessa quebra no texto. Embora capture problemas locais de coerência, quando aplicado ao texto como um todo o modelo de grade de entidades não identifica a localização do possível problema.

Visando a automatização da dimensão Quebra de Linearidade como parte do MAC, este trabalho propõe utilizar informações provenientes da estrutura retórica em conjunto com o modelo de grade de entidades para gerar mensagens que indiquem possíveis problemas de coerência local em regiões específicas de resumos científicos. Nessa proposta, os componentes da estrutura retórica automaticamente detectada passam a servir de referência para a geração de mensagens ao usuário informando, por exemplo, que uma possível quebra de linearidade foi detectada em certo componente retórico. Para que isso seja possível, primeiramente são construídas grades de entidades para pequenos trechos do texto que correspondem a porções da estrutura retórica do resumo. Esses trechos vão sendo expandidos em trechos maiores até que se chegue ao texto do resumo como um todo. Como a análise da dimensão é feita por trechos, que correspondem a um ou mais componentes retóricos, no momento em que o sistema detecta uma possível quebra de linearidade ele é capaz de informar a sua localização aproximada.

Para verificar se o modelo de grade de entidades seria capaz de detectar problemas de quebra de linearidade tanto em textos completos quanto em trechos pequenos foram realizados experimentos com um *corpus* de resumos científicos, a partir dos quais foram extraídos trechos compostos de pares de componentes retóricos. Também foram realizados experimentos com textos originais anotados manualmente quanto à Dimensão Quebra de Linearidade. Os resultados experimentais mostraram que o modelo consegue detectar as quebras, mesmo em trechos compostos de poucas sentenças, permitindo o uso da abordagem proposta como parte do MAC da ferramenta SciPo. Tal fato também é comprovado pela avaliação positiva realizada com usuários reais, em que um protótipo do ambiente SciPo com novas sugestões relacionadas às quebras de linearidade adicionadas ao MAC é avaliado.

Esta dissertação está dividida em seis seções, sendo esta a primeira delas. Na Seção 2 são introduzidos conceitos relacionados à coerência textual, bem como teorias e métodos empregados por trabalhos da literatura na análise de coerência; na Seção 3 são apresentadas ferramentas computacionais que realizam a análise automática de textos e que possuem algum critério de análise que julgamos relacionado à coerência, entre as ferramentas está o ambiente de auxílio à escrita SciPo, no qual este trabalho está diretamente inserido; a Seção 4 descreve o *corpus* utilizado no desenvolvimento e avaliação desta proposta; na Seção 5 é apresentada a

implementação dos classificadores, também são abordadas as fases do desenvolvimento do protótipo, a avaliação intrínseca, bem como a avaliação do protótipo com usuários reais; na Seção 6 são apresentadas as conclusões deste trabalho. Também faz parte desta dissertação um apêndice contendo o questionário utilizado na avaliação do protótipo desenvolvido.

2. Coerência Textual

Coerência é derivada do latim *cohaerensentis*, que significa "o que está junto ou ligado". Ela está presente na produção textual, a qual é composta por diversos elementos que possuem significados individuais, mas que quando associados transmitem uma mensagem unificada. Nos casos em que as relações estejam desconexas, haverá falha na comunicação da mensagem. A coerência é, por assim dizer, o que faz com que uma sequência linguística qualquer seja considerada um texto, porque é ela, por meio de vários fatores, que permite ao receptor estabelecer as relações (sintático-gramaticais, semânticas e pragmáticas) entre os elementos da sequência (capítulos, parágrafos, frases, etc.).

Para Koch e Travaglia (2008), a coerência está relacionada à possibilidade de constituir um sentido ao texto, devendo, dessa forma, ser compreendida como um princípio de interpretabilidade, unida à inteligibilidade do texto em uma situação de comunicação e à capacidade que o receptor tem para avaliar o significado do texto. Ela é o instrumento para constituir, no texto, alguma forma de unidade ou relação. Essa unidade é sempre apresentada como uma unidade de sentido no texto, o que caracteriza a coerência como global, isto é, referente ao texto como um todo.

Na visão de Lima (2008), a coerência é a estruturação lógica do texto, de forma que as ideias principais e secundárias estejam relacionadas e subordinadas, estabelecendo princípio, meio e fim. Assim, para que o texto seja coerente e forneça ao leitor uma mensagem completa, é necessário que os elementos (frases, parágrafos, etc.) não possuam contradições entre eles.

Segundo Garcia (2006), a coerência ordena e interliga as ideias de maneira clara e lógica, de acordo com o plano estabelecido, e sem ela não se faz possível obter ao mesmo tempo unidade e clareza. Dessa forma, é preciso que o raciocínio seja lógico e não apresente lapsos, hiatos e deslocamentos abruptos das informações. Garcia (2006) ainda destaca a importância das palavras na composição textual:

> "Palavras desconexas são como fragmentos de um jarro de porcelana. É preciso colá-las, interligá-las para se

obter uma unidade de comunicação eficaz".

Para Van Dijk e Kintsch (1983), a coerência pode ser dividida em quatro diferentes tipos: a Semântica, a Sintática, a Estilística e a Pragmática, sendo cada uma delas explanada a seguir.

- Coerência Semântica: refere-se à relação de sentido entre as frases de um texto, ou no texto como um todo. Assim, pode-se dizer que, se o texto possui contradições de sentidos e mau uso dos termos em relação ao seu significado, ele possui incoerências semânticas.
- Coerência Sintática: refere-se aos meios sintáticos para expressar a coerência semântica, ou seja, está relacionada ao posicionamento e à escolha dos conectivos, pronomes, verbos, etc.
- Coerência Estilística: refere-se à utilização de elementos linguísticos pertencentes ou característicos de um mesmo estilo linguístico. A incoerência dar-se-á então pela utilização de mais de um estilo, sendo eles: formal e informal.
- Coerência Pragmática: refere-se ao texto visto como uma sequência de atos de fala. Estes são relacionados de modo que, para a sequência de atos ser percebida como apropriada, os atos de fala que a constituem devem satisfazer as mesmas condições presentes em uma dada situação comunicativa. Caso contrário, temos incoerência.

Ainda segundo Van Dijk (1981); Van Dijk e Kintsch(1983), existem dois níveis básicos de coerência: local e global, os quais são interdependentes, pois a coerência global depende de um conjunto de sentenças coerentes, assim como a coerência local pode ser constituída a partir da coerência global. Desta forma, os autores definem os níveis da seguinte maneira:

- Coerência local (ou microestrutural): refere-se a frases ou a sequências de frases dentro do texto. Está sujeita ao uso correto da língua para expressar sentidos que possibilitem a compreensão do texto. A incoerência local não impede o leitor de compreender o todo do texto, mas quando excessivamente presente pode tornar o texto incoerente.
- Coerência global (macroestrutural): diz respeito ao texto como um todo ou a fragmentos maiores, nos quais se estabelece uma conexão com o tema, ideia, essência ou finalidade do texto.

Dessa forma, a coerência é considerada como um dos principais elementos para a elaboração de textos, pois estabelece significado às partes e ao todo, dando sentido ao mesmo. Para tanto, faz-se necessário se manter uma apresentação lógica e harmônica entre as ideias, demonstrando-as de forma clara e ordenada, formando, assim, uma unidade na qual as partes e

o todo tenham sentido para o leitor.

2.1. Teorias e Modelos Relacionados à Coerência Textual

A seguir são apresentadas teorias e modelos relacionados à coerência textual que têm sido empregados em sistemas computacionais, sejam eles linguísticos ou estatísticos. Por ser utilizada por diversas ferramentas de auxílio à escrita (Foltz et al., 2013; Schultz, 2013), incluindo o MAC do ambiente SciPo (Souza e Feltrim, 2013), a LSA é apresentada na próxima subseção. A Teoria da Centralização é apresentada na Seção 2.1.2 por também ter sido empregada em diversos estudos relativos à coerência textual (Barzilay e Lapata, 2008; Miltsakaki e Kukich, 2004). Por ter sido empregado como parte deste trabalho, o modelo de grade de entidades, bem como a implementação feita por Freitas (2013) são apresentados nas subseções 2.1.3 e 2.1.4, respectivamente.

2.1.1. Análise de Semântica Latente (LSA)

Proposta nos anos 90 e extensivamente utilizada desde então, a Análise de Semântica Latente (*Latent Semantic Analysis* - LSA) é uma teoria e um método para a extração e representação da semântica de palavras em um contexto, obtida por meio de cálculos estatísticos aplicados a grandes *corpora* (Deerwester et al., 1990; Landauer e Dumais, 1997; Landauer et al., 1998).

Seu modelo de indexação semântica é baseado na coocorrência de palavras ou termos em um *corpus*. A suposição é que a ocorrência conjunta de termos dentro de um mesmo trecho do documento (parágrafos, frases ou textos) é representativa de similaridade semântica. Na prática, esse método é utilizado para a construção de um espaço semântico, no qual não só palavras, mas, sentenças, parágrafos, textos, podem ser representados por vetores.

A primeira etapa do processo é a seleção de um *corpus* que será utilizado para a criação do espaço semântico. É comum que a *corpus* sejam aplicadas técnicas de pré-processamento tradicionais como, remoção de caracteres diferentes de letras, conversão de letras maiúsculas em minúsculas e remoção de *stopwords* (palavras de uso muito frequente, como artigos, preposições e conjunções, que semanticamente não contribuem de forma relevante para um documento). Em seguida, é construída uma matriz de representação dessa coleção, com as linhas correspondendo aos termos selecionados e as colunas correspondendo aos documentos que constituem o *corpus*, como exemplificado na Tabela 2.1.

Tabela 2.1. Exemplo de matriz de coocorrência de termos

	Texto 1	**Texto 2**	**Texto 3**	**Texto N**
Termo 1	2	1	3	...
Termo 2	1	0	3	...
Termo 3	4	2	1	...
Termo K	...	...	...	...

Inicialmente, à cada célula dessa matriz é atribuído o valor da frequência absoluta do termo no texto. A frequência absoluta de cada termo, em cada uma de suas entradas na matriz, é transformada em seu logaritmo (*term frequency* - TF). Isto é feito baseando-se no fato de que um documento com, por exemplo, três ocorrências de uma mesma palavra, tende a ser mais importante do que um documento com apenas uma ocorrência, porém, não três vezes mais importante. O cálculo de TF atribui valores próximos a zero para os termos mais raros, porém, a teoria afirma que termos raros são considerados mais informativos do que termos frequentes. Para aumentar o valor dos termos raros, um outro cálculo chamado *inverse document frequency* - IDF é aplicado sobre todos os termos da matriz, a saber:

$$IDF_t = \log (N/df_t)$$

N = número de documentos do *corpus;*

df_t = número de documentos que contém o termo *t.*

Para a normalização dos valores dos termos, os dois cálculos são combinados (TF-IDF = TF X IDF).

O próximo passo é a aplicação da técnica conhecida como Decomposição de Valor Singular (SVD) (Golub e Reinsch, 1970). Por meio dessa técnica são geradas outras três matrizes a partir da matriz original, e uma nova matriz *M* é obtida a partir do produto destas três matrizes.

$$M = T \times S \times D$$

T = matriz de vetores singulares à esquerda;

S = matriz diagonal de valores singulares em ordem decrescente;

D = matriz de vetores singulares à direita.

A dimensão dessas matrizes é reduzida, eliminando as linhas e colunas correspondentes aos menores valores singulares da matriz *S*, assim como as colunas da matriz *T* e as linhas da matriz *D*.

A SVD é normalmente utilizada para localizar a informação semântica essencial em uma matriz de coocorrência de palavras. Com isso, a partir dessa decomposição, é possível, com a redução de dimensão das matrizes *T*, *S* e *D* (mantendo somente os maiores valores singulares), descartar as informações acidentais que geralmente estão presentes.

O objetivo com o produto dessas três novas matrizes reduzidas é obter um espaço semântico condensado que representa as melhores relações entre as palavras e os documentos. A proximidade entre duas palavras é obtida calculando-se o cosseno do ângulo entre seus vetores (linhas da matriz) correspondentes. Quanto maior o cosseno do ângulo entre os vetores de duas palavras, maior a proximidade entre elas, como representado na Figura 2.1, em que os termos {“casa”, “sacada”} são mais próximos (0,65) do que os outros pares de termos ({“casa”, “corrida”} e {“sacada”, “corrida”}), ambos com cosseno = 0,01.

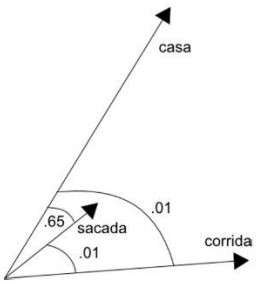

Figura 2.1. Comparação LSA entre três palavras (Kintsch, 2002)

Finalmente, o vetor de representação de um dado conjunto de palavras, como parágrafos ou textos, no espaço, pode ser obtido por meio da média de todos os vetores das palavras desse conjunto, permitindo, assim, a obtenção da proximidade entre uma palavra e um texto, e até mesmo entre dois textos.

Na Seção 3 são apresentadas algumas ferramentas categorizadas como *Automated Essay Scoring* que fazem o uso de LSA, por exemplo, para comparar o texto avaliado com um texto base escrito por um profissional e emitir uma nota, levando em consideração a similaridade semântica entre os textos.

2.1.2. Teoria da Centralização

Proposta inicialmente nos anos 80, a teoria da centralização (*Centering*) (Grosz et al., 1983; Grosz et al. 1995) também tem sido utilizada em diversos estudos relativos à coerência e a saliência das entidades no discurso (Althaus et al., 2004; Hasler, 2004; Karamanis et al., 2004; Miltsakaki e Kukich, 2004; Poesio et al., 2004).

Um dos princípios da teoria é que certos elementos em um dado enunciado são mais centrais que outros. Assim, cada enunciado tem um centro preferencial e a maneira como esse centro é tratado no enunciado seguinte tem um impacto na coerência do discurso. A ideia principal da teoria é relacionar o foco de atenção e a escolha da expressão referencial com a coerência entre

enunciados no mesmo segmento discursivo. Para tal, a teoria propõe um modelo do nível local do estado de atenção.

Um dos objetivos da teoria é analisar os níveis de complexidade inferencial de diferentes segmentos do discurso. O mecanismo criado pela teoria visa determinar os fatores de processamento discursivo responsáveis por diferenças na coerência de um dado segmento.

Para uma melhor explicação das definições e regras da teoria, um exemplo foi traduzido de Grosz et al.(1995) em que dois segmentos (S_1 e S_2) são apresentados, sendo que a diferença entre eles está somente no último enunciado (*utterance* U_4).

S_1: (U_1) João foi a sua loja de música favorita para comprar um piano.
(U_2) Ele frequentava a loja há vários anos. (Ele = João)
(U_3) Estava animado por finalmente poder comprar um piano. (Estava = João)
(U_4) Ele chegou justo quando a loja fechava. (Ele = João)

S_2: (U_1) João foi a sua loja de música favorita para comprar um piano.
(U_2) Ele frequentava a loja há vários anos. (Ele = João)
(U_3) Estava animado por finalmente poder comprar um piano. (Estava = João)
(U_4) Ela estava fechando assim que João chegou. (Ela = loja de música)

Embora ambos os segmentos apresentem a mesma informação, o segmento S_1 é intuitivamente mais coerente que o S_2. Isso se dá pela troca do centro de atenção (João e loja de música) que acontece no segmento S_2. Já no segmento S_1, por sua vez, todos os enunciados têm o mesmo foco de atenção, apresentando uma continuidade. O principal objetivo da teoria da centralização é capturar essas diferenças.

Os centros ou focos de atenção são objetos semânticos que servem para representar a ligação entre os enunciados $(U_i \ldots U_n)$ de um segmento do discurso. Cada enunciado pode ter mais de um centro de atenção. Esse conjunto de centros de atenção é chamado de *Forward-Looking-Centers* ou centros prospectivos (C_f) e representa os potenciais centros do próximo enunciado. Por exemplo, no enunciado U_1 do segmento S_1 apresentado anteriormente, é identificado o seguinte conjunto de entidades $C_f(U_1)$ = {João, loja de música, piano}. A ordenação do conjunto de entidades é feita conforme a função gramatical subcategorizada pelo verbo principal: sujeito > objeto direto > objeto indireto > outras subcategorizações > adjuntos. Essa ordenação é muito importante para a teoria, pois problemas na ordenação podem levar a escolha incorreta dos elementos de outros conjuntos. O primeiro elemento do conjunto $C_f(U_i)$ é o mais altamente classificado e denominado *Preferred Center* ou próximo centro preferencial,

representado por $C_p(U_i)$. Os autores afirmam que em um segmento coerente o (C_p) do enunciado (U_i) deve ser o centro de atenção do enunciado subsequente (U_{i+1}). Outro elemento da teoria é o *Backward-Looking Center* (C_b) ou centro retrospectivo, que também pertence ao C_f de um enunciado. O $C_b(U_i)$ é uma referência a um elemento de $C_f(U_{i-1})$, desde que i seja maior que um. Cada enunciado pode ter apenas um C_b e o $C_b(U_i)$ é o elemento melhor classificado em $C_f(U_{i-1})$ e que é realizado em (U_i).

Os elementos identificados no enunciado (U_4) e do seguimento (S_2) do exemplo apresentado anteriormente são:

$C_f(U_4)$ = {loja de música (Ela) e João}

$C_p(U_4)$ = {loja de música (Ela)}

$C_b(U_4)$ = {João (Estava)}

Grosz et al.(1995) afirmam que o grau de coerência do seguimento de um discurso é maior quando o elemento $C_b(U_i)$ é igual ao elemento $C_p(U_{i-1})$ e também quando o elemento $C_b(U_i)$ é igual ao elemento $C_p(U_i)$. Além dessa afirmação em relação à coerência, os autores estabelecem quatro transições que representam as relações entre os elementos identificados nos enunciados subsequentes, são elas: *Continue*, *Retain*, *Smooth-Shift* e *Rough-Shift*. A Tabela 2.2 apresenta as relações de equivalência entre os conjuntos C_b e C_p e as transições previstas pela teoria da centralização.

Tabela 2.2. Relações de equivalência entre os conjuntos C_b e C_p e transições (Poesio et al., 2004)

	$C_b(U_i) = C_b(U_{i-1})$ or $C_b(U_{i-1}) = NIL$	$C_b(U_i) \neq C_b(U_{i-1})$
$C_b(U_i) = C_p(U_i)$	*Continue*	*Smooth-Shift*
$C_b(U_i) \neq C_p(U_i)$	*Retain*	*Rough-Shift*

A teoria define uma ordem preferencial para as transições, sendo ela: *Continue* > *Retain* > *Smooth-Shift* >*Rough-Shift*. Sendo assim, um discurso composto por transições *Continue* será considerado mais coerente pela teoria do que um discurso composto por transições *Shift*.

No início desta seção foi afirmado que apesar dos segmentos S_1 e S_2 apresentarem a mesma informação, S_1 é mais coerente devido à continuidade apresentada entre os enunciados U_3 e U_4. Para verificar essa afirmação, nas Tabelas 2.3 e 2.4 é apresentada a identificação dos elementos dos enunciados U_3 e U_4 respectivamente, do seguimento S_1, e nas Tabelas 2.5 e 2.6 são apresentadas os elementos identificados nos enunciados U_3 e U_4 respectivamente, do seguimento S_2.

Tabela 2.3. Entidades identificadas no enunciado U_3 do seguimento S_1

Estava animado por finalmente poder comprar um piano.		
$C_f(U_3)$	$C_p(U_3)$	$C_b(U_3)$
{João, piano}	*{João}*	*{João}*

Tabela 2.4. Entidades identificadas no enunciado U_4 do seguimento S_1

Ele chegou justo quando a loja fechava.		
$C_f(U_4)$	$C_p(U_4)$	$C_b(U_4)$
{João, loja de música}	*{João}*	*{João}*

Observando as entidades identificadas nos enunciados U_3 e U_4 do seguimento S_1 e comparando com a Tabela 2.2 de equivalências e transições, a transição identificada é a *Continue*, pois $C_b(U_4) = C_b(U_3)$ *e* $C_b(U_4) = C_p(U_4)$.

Tabela 2.5. Entidades identificadas no enunciado U_3 do seguimento S_2

Estava animado por finalmente poder comprar um piano.		
$C_f(U_3)$	$C_p(U_3)$	$C_b(U_3)$
{João, piano}	*{João}*	*{João}*

Tabela 2.6. Entidades identificadas no enunciado U_4 do seguimento S_2

Ela estava fechando assim que João chegou.		
$C_f(U_4)$	$C_p(U_4)$	$C_b(U_4)$
{loja de música, João}	*{loja de música}*	*{João}*

Observando as Tabelas 2.5 e 2.6 de entidades identificadas nos enunciados U_3 e U_4 do seguimento S_2 e comparando com a Tabela 2.2 de equivalências e transições, a transição identificada é a *Retain*, pois $C_b(U_4) = C_b(U_3)$ *e* $C_b(U_4) \neq C_p(U_4)$. Assim, se confirma que a teoria considera o seguimento S_1 com um nível maior de coerência em relação à S_2.

2.1.3. Grade de Entidades

O modelo de grade de entidades (*Entity-Grid*), proposto por Barzilay e Lapata (2008), afirma que o requisito principal em sistemas de produção de texto é a coerência. O foco do modelo está sobre a coerência local e na captura do nível de relacionamento entre as sentenças adjacentes de um texto, como condição necessária para se atingir coerência global.

Assim como na teoria da centralização, a premissa fundamental da grade de entidades é que a distribuição das entidades (representadas por sintagmas nominais correferentes) em um texto coerente apresenta certa regularidade. Assim, seria possível extrair um padrão representativo de coerência a partir de *corpus* de treinamento.

No modelo cada texto é representado por uma grade de entidades, ou tabela, que captura a distribuição das entidades do discurso sobre as sentenças, na qual as linhas da grade representam as sentenças e as colunas correspondem às entidades do discurso. Para cada ocorrência de uma entidade, a célula correspondente na grade contém informação sobre a presença ou ausência daquela entidade na sentença. Além de marcar a ocorrência da entidade, a célula também pode conter informação a respeito da sua função sintática na sentença. Essa informação é representada por um caractere que representativo da função sintática mais proeminente. Em ordem de proeminência, as funções sintáticas consideradas são: Sujeito (S), Objeto (O) e Outro (X). Entidades ausentes em uma determinada sentença são representadas por uma lacuna (-). No caso de uma mesma entidade aparecer em uma sentença com diferentes funções sintáticas, a entidade é representada na grade pela função de maior proeminência.

A Figura 2.2, extraída de Barzilay e Lapata (2008), apresenta um fragmento de uma grade de entidades construída a partir do texto representado na Figura 2.3. A grade tem seis linhas, assim como o texto tem seis sentenças, e cada coluna representa uma entidade, bem como a presença e sua função sintática ou ausência dela na sequência de sentenças. Considere por exemplo a coluna da entidade *Suit* [– – – – O –], ela mostra que a entidade está presente somente na sentença 5 na função de objeto (O) e está ausente no restante das sentenças, configurando assim uma coluna esparsa. Em contrapartida, a coluna da entidade *Microsoft* [S O S S – S] é mais densa, pois essa entidade só não está presente na sentença 5, fato representado pela lacuna (–). Ainda sobre a entidade *Microsoft*, na primeira sentença ela aparece duas vezes, uma como *Microsoft Corp.*(X) e outra como *the company*(S). Ambas foram identificadas como sendo a mesma entidade, pois a grade de entidades foi construída utilizando-se resolução de correferência, mas somente a função Sujeito (S) foi representada na grade por ser a mais proeminente. Caso o sistema de resolução de correferência não consiga identificar dois sintagmas nominais como correferentes, duas entidades distintas serão representadas na grade.

	Department	Trial	Microsoft	Evidence	Competitors	Markets	Products	Brands	Case	Netscape	Software	Tactics	Government	Suit	Earnings	
1	S	O	S	X	O	–	–	–	–	–	–	–	–	–	–	1
2	–	–	O	–	–	X	S	O	–	–	–	–	–	–	–	2
3	–	–	S	O	–	–	–	–	S	O	O	–	–	–	–	3
4	–	–	S	–	–	–	–	–	–	–	–	S	–	–	–	4
5	–	–	–	–	–	–	–	–	–	–	–	–	S	O	–	5
6	–	X	S	–	–	–	–	–	–	–	–	–	–	–	O	6

Figura 2.2. Fragmento de uma grade de entidades (Barzilay e Lapata, 2008).

1 [The Justice Department]$_S$ is conducting an [anti-trust trial]$_O$ against [Microsoft Corp.]$_X$ with [evidence]$_X$ that [the company]$_S$ is increasingly attempting to crush [competitors]$_O$.
2 [Microsoft]$_O$ is accused of trying to forcefully buy into [markets]$_X$ where [its own products]$_S$ are not competitive enough to unseat [established brands]$_O$.
3 [The case]$_S$ revolves around [evidence]$_O$ of [Microsoft]$_S$ aggressively pressuring [Netscape]$_O$ into merging [browser software]$_O$.
4 [Microsoft]$_S$ claims [its tactics]$_S$ are commonplace and good economically.
5 [The government]$_S$ may file [a civil suit]$_O$ ruling that [conspiracy]$_S$ to curb [competition]$_O$ through [collusion]$_X$ is [a violation of the Sherman Act]$_O$.
6 [Microsoft]$_S$ continues to show [increased earnings]$_O$ despite [the trial]$_X$.

Figura 2.3. Texto base para a grade de entidades (Figura 2.2) (Barzilay e Lapata, 2008).

As autoras consideram ainda, que as grades de entidades extraídas a partir de textos coerentes são suscetíveis a possuírem duas características: algumas colunas densas, em que a maior parte de seus elementos são diferentes de lacunas (-) e em que as funções sintáticas Sujeito (S) e Objeto (O) apareçam em maior número, como é o caso da coluna da entidade *Microsoft*, e muitas colunas esparsas, constituídas principalmente de lacunas, como é caso das entidades *Department*, *Brands* e *Earnings*.

Somente com a construção da grade de entidades ainda não é possível fazer a tradução para um modelo computacional, sendo ainda necessário extrair padrões da grade que possam ser tratados computacionalmente. Isso é feito mapeando as transições locais de um tamanho pré-definido *n*. Uma transição de entidade local é uma sequência $\{S, O, X, -\}^n$ que representa a ocorrência da entidade e suas funções sintáticas em *n* sentenças adjacentes. Essas transições podem ser obtidas a partir de um conjunto subsequente de colunas da grade. Cada transição terá certa probabilidade para uma determinada grade. Por exemplo, a probabilidade da sequência *{O –}* na grade representada pela Figura 2.2 é de 0,09, já que essa transição ocorre sete vezes em um total de 75 possíveis transições de tamanho dois (7 ÷ 75 = 0,09). Assim, cada texto pode ser representado por um vetor de características, no qual cada valor corresponde à probabilidade de certa transição.

Existe um grande número de transições passíveis de serem incluídas no vetor de características. Uma forma de limitar o tamanho do vetor de características é considerar apenas transições de um dado tamanho, normalmente tamanho dois ou três, ou verificar apenas as transições mais frequentes nos documentos. Isso impede que haja uma explosão do número de características, o que causaria um aumento excessivo da quantidade necessária de dados de treinamento e elevando o custo de processamento computacional.

Dessa forma, um documento d_i pode ser representado por uma grade de entidades x_{ij}, que por sua vez pode ser representada por um conjunto fixo de sequência de transições locais e suas probabilidades, constituindo um vetor de características $\Phi(x_{ij})$:

$$\Phi(x_{ij}) = (p_1(x_{ij}), p_2(x_{ij}), ..., p_m(x_{ij}))$$

em que $_m$ é o número de todas as transições locais pré-definidas e $p_t(x_{ij})$ é a probabilidade da transição $_t$ na grade x_{ij}. Essa representação por meio de vetor de características é bastante útil, pois permite a utilização dos dados por algoritmos de aprendizagem de máquina, que por sua vez podem descobrir padrões relevantes de distribuição de entidades a serem utilizados na avaliação de coerência.

A construção da grade de entidades pode ser feita levando em conta três diferentes tipos de dimensões linguísticas e essas dimensões influenciam diretamente nas probabilidades de transições e na extração do vetor de características. São elas:

(i) identificação de entidades correferentes: são identificados sintagmas nominais que se referem à mesma entidade em um texto. O uso dessa dimensão pode ou não ser considerada durante a extração de entidades (correferência (+/-)), alterando as probabilidades das transições locais.

(ii) função sintática: quando essa dimensão é considerada (função sintática (+)) a grade de entidades é construída e as células são preenchidas com os caracteres {S, O, X, -} que representam a presença e a função sintática de uma entidade ou a ausência da entidade em uma determinada sentença; caso contrário, se essa dimensão não for considerada (função sintática (-)) a grade marcará apenas a presença {P} ou a ausência {-} da entidade na sentença.

(iii) saliência, que também pode ou não ser considerada no processo de identificação das entidades (saliência (+/-)): são consideradas salientes entidades que aparecem no texto com certa frequência. Em seus experimentos, as autoras consideraram salientes as entidades mencionadas duas ou mais vezes no texto. Quando a dimensão de saliência não é considerada (saliência (-)), a grade é composta por todas as entidades do texto; caso contrário (saliência (+)), duas grades diferentes são construídas, uma delas contendo as entidades salientes e outra contendo o restante das entidades. Nesse caso, as probabilidades são calculadas separadamente para cada grade e depois são combinadas em um único vetor de características.

Dessa forma, considerando a presença (+) ou a ausência (-) desses três tipos de informação, oito configurações diferentes da grade de entidades podem ser obtidas fazendo-se as combinações de correferência (+/-), função sintática (+/-) e saliência (+/-).

Segundo relatado pelas autoras, a grade de entidades obteve bom desempenho em três experimentos: (1) ordenação de sentenças, (2) avaliação de coerência em resumos gerados automaticamente e (3) avaliação de legibilidade. Para todos os experimentos foi utilizado um

corpus formado por textos jornalísticos e também todas as três dimensões linguísticas foram testadas e avaliadas. O experimento (1) consistiu em ordenar diferentes sequências de sentenças de um mesmo texto, esperando que as sequências mais coerentes ficassem nas primeiras posições da ordenação. A partir das dimensões linguísticas (Correferência, Função Sintática e Saliência) foram gerados oito modelos para treinamento e teste. O modelo que utiliza todas as dimensões (Correferência+ Sintática+ Saliência+) obteve o melhor desempenho.

O experimento (2) tratou a tarefa de avaliação de coerência em resumos automaticamente gerados por diferentes sistemas de sumarização automática e resumos produzidos por escritores humanos, todos provenientes da conferencia DUC 2003. Para essa tarefa, o melhor desempenho foi alcançado utilizando a configuração Correferência- Sintática+ Saliência+.

Por fim, o experimento (3) buscou avaliar se o modelo seria útil para avaliar a legibilidade de textos (facilidade de leitura e compreensão de um documento (Barzilay e Lapata, 2008)). O melhor desempenho foi obtido pelo modelo com a configuração Correferência- Sintática+ Saliência+.

Considerando os experimentos, as autoras observaram a importância do uso das dimensões Função Sintática e Saliência na construção da grade de entidades, uma vez que os melhores resultados foram obtidos quando essas duas dimensões linguísticas foram utilizadas. A descrição completa do modelo e o detalhamento dos resultados dos experimentos estão disponíveis em Barzilay e Lapata (2008).

2.1.4. Grade de Entidades Para o Português

Dados os resultados encorajadores do modelo de grade de entidades em diferentes tarefas de avaliação de coerência, Freitas (2013) verificou a sua aplicabilidade na tarefa de avaliação de coerência de resumos científicos escritos em português por nativos da língua portuguesa. Além disso, o autor buscou avaliar se tal modelo poderia ser empregado na implementação de um classificador capaz de detectar problemas locais de coerência, semelhantes aos descritos na dimensão Quebra de Linearidade proposta por Souza (2011), visando a futura inclusão de tal classificador no módulo de análise de coerência da ferramenta SciPo.

A implementação de Freitas foi feita segundo a proposta original de Barzilay e Lapata (2008) e usando ferramentas de PLN disponíveis para o português. A identificação dos sintagmas nominais e de suas funções sintáticas foi feita utilizando a versão *online*[1] do *parser*

[1] Disponível em http://beta.visl.sdu.dk/visl/pt/parsing/automatic/trees.php. Acessado em 21/01/2016.

PALAVRAS (Bick, 2000). Das três dimensões linguísticas propostas no modelo original, Freitas (2013) implementou as dimensões (ii) função sintática e (iii) saliência. A dimensão (i) não foi implementada pela indisponibilidade de uma ferramenta robusta de resolução automática de correferência para o português. Para minimizar os efeitos da falta dessa dimensão, o autor lematizou (conversão das palavras em suas formas canônicas) os sintagmas nominais (SNs), visando reduzir a duplicação de entidades na grade. Essa abordagem é similar à adotada por Elsner e Charniak (2011), na qual sintagmas com o mesmo núcleo são considerados correferentes. Para núcleos compostos, como "linguagem de programação", foi verificado se um dos componentes do núcleo já consta na grade, por exemplo, "linguagem" e, nesse caso, "linguagem de programação" e "linguagem" foram considerados correferentes e representados na grade como uma única entidade.

Após a construção da grade de entidades, a extração do vetor de características foi feita para quatro configurações de grade: função sintática (+/-) e saliência (+/-).

Assim como Barzilay e Lapata (2008), Freitas considerou, para a construção do vetor de características, as probabilidades de todas as transições possíveis de tamanho dois. Dessa forma, quando a função sintática é levada em conta, o vetor de características contém as probabilidades de 16 possíveis transições, a saber: SS, SO, SX, S-, OS, OO, OX, O-, XS, XO, XX, X-, -S, -O, -X, --. Quando a função sintática não é considerada (-) o vetor de características contém as probabilidades das quatro transições possíveis: XX, X-, -X, --.

Com relação à dimensão saliência, Freitas novamente utilizou os mesmos parâmetros utilizados nos experimentos de Barzilay e Lapata (2008): são consideradas duas classes de saliência, uma classe contém as entidades que aparecem duas ou mais vezes no discurso e outra classe contém as entidades que aparecem apenas uma vez no texto.

Para a avaliação do modelo grade de entidades implementado para o português foram feitos dois experimentos comumente realizados em trabalhos relacionados (Barzilay e Lapata, 2008; Elsner e Charniak, 2011; Filippova e Strube, 2007; Yokono e Okumura, 2010), são eles: (1) ordenação de sentenças e (2) classificação de coerência de textos baseado em julgamento de juízes humanos.

Para o experimento (1) foi utilizado um *corpus* de 286 textos jornalísticos coletados a partir de três outros *corpora*, são eles: CSTNews (Cardoso et al., 2011), Summit (Collovini et al., 2007) e Temário (Rino e Pardo, 2007). Para cada texto do *corpus* foram criadas 20 versões sintéticas diferentes nas quais as sentenças foram desordenadas de forma aleatória. Nesse experimento, o texto é visto como um conjunto de sentenças e o algoritmo de ordenação busca encontrar uma ordem que maximize a coerência do texto de acordo com alguns critérios, por

exemplo, a probabilidade de uma ordem. No experimento de Freitas essa tarefa foi simplificada, de modo que o texto original, considerado coerente, foi comparado com suas versões sintéticas, consideradas incoerentes. Em vez de buscar encontrar a melhor ordem, o algoritmo deve escolher, entre duas ordens apresentadas, aquela que julgar a mais coerente. O experimento foi realizado utilizando o sistema SVM^{rank} (Joachims, 2006), que implementa o algoritmo SVM para problemas de ordenação. O percentual de acerto para as quatro configurações da grade de entidades, bem como o resultado de uma *baseline* baseada em LSA, podem ser vistos na Tabela 2.7.

Tabela 2.7. Percentual de acerto para o experimento (1) de Freitas (2013).

Modelo	**Acerto**
LSA	67,00%
Sintático+ Saliência-	62,11%
Sintático+ Saliência+	58,11%
Sintático- Saliência-	**68,58%**
Sintático- Saliência+	67,37%

O experimento (2) verificou o desempenho do modelo grade de entidades na avaliação de resumos científicos, distinguindo entre resumos "com problemas" ou "sem problemas" de coerência. Para isso foi utilizado um *corpus* composto por 139 resumos científicos manualmente anotados por dois juízes humanos. A concordância entre os juízes medida em um subconjunto de 40 resumos por meio da medida *Kappa* foi de 0,70. O experimento foi realizado no ambiente *WEKA* (Witten e Frank, 2005) com os algoritmos SMO, J48 e *Naive Bayes*. Os melhores resultados foram obtidos com o algoritmo J48 e com o modelo na configuração Sintático- Saliência+, sendo eles: *Kappa* = 0,65 e *F-Measure*= 0,91.

O detalhamento dos experimentos realizados com os *corpora* em português, bem como dos resultados obtidos, pode ser encontrado em Freitas (2013).

2.1.5. Entidades Semanticamente Relacionadas

Filippova e Strube (2007) realizaram um estudo para verificar a aplicabilidade de um modelo de conhecimento semântico para o agrupamento de entidades em oposição a resolução de correferência originalmente utilizada no modelo grade de entidades de Barzilay e Lapata (2008).

Especificamente, o trabalho teve dois objetivos: (1) verificar se a integração de conhecimento semântico de fato melhora os resultados atingidos usando-se correferência e (2)

verificar se o uso do relacionamento semântico é confiável para agrupamento de entidades no caso de não haver disponibilidade de resolução de correferência.

Filippova e Strube (2007) utilizaram a API *WikiRelate!* (Strube e Ponzetto, 2006) para calcular relações semânticas entre entidades. A escolha pela *WikiRelate!* como fonte de conhecimento semântico se deu pelo fato do *corpus* utilizado ser de textos jornalísticos, que em geral contêm entidades que são nomes próprios (pessoas, locais, empresas) e a *Wikipédia* seria então a melhor escolha.

Os experimentos foram realizados nos mesmos moldes do experimento (1) de Barzilay e Lapata (2008), com a diferença do *corpus* utilizado para gerar as versões permutadas – 100 textos jornalísticos em alemão com anotação manual de funções sintáticas e resolução manual de correferência. Utilizando o processo original para a identificação e agrupamento das entidades, o melhor desempenho foi alcançado com o modelo Correferência+ Sintático-Saliência+, com acurácia de 75%. Os resultados utilizando a *WikiRelate!* para agrupar entidades relacionadas ficaram em torno de 5% abaixo dos modelos que usam correferência, mostrando que o uso da *WikiRelate!* no processo de agrupamento de entidades é melhor do que não usar informação nenhuma (Correferência-), mas não é tão bom quanto o uso de informação de correferência.

2.1.6. Teoria da Estrutura Retórica

Lin et al., (2011) propuseram um novo modelo, que combina o modelo grade de entidades com relações discursivas semelhantes as da Teroria da Estrutura Retórica RST (Mann e Thompson, 1988) que é uma teoria discursiva cujo objetivo é descrever a organização do texto por meio de relações entre suas partes funcionais. Essas partes são porções de texto chamadas de *text spans*, constituídas de intervalos lineares e ininterruptos de texto, e o conjunto de relações definidas entre essas partes forma a estrutura retórica do texto.

Esse novo modelo também usa uma grade de entidades, mas em vez das entidades serem representadas na grade por seus papeis sintáticos (S;O;X), são representadas pelo tipo de relação retórica em que aparecem (*Temporal*, *Contingency*, *Comparison* e *Expansion*). Desse modo, a grade de entidades é utilizada para se calcular as probabilidades de transições entre relações retóricas em vez de transições entre funções sintáticas. Para a identificação das relações retóricas, Lin et al. (2011) utilizaram a D-LTAG (*Discourse Lexicalized Tree Adjoining Grammar*) (Webber, 2004).

Os experimentos foram realizados no mesmo formato do experimento (1) de Barzilay e Lapata (2008), utilizando, inclusive, os mesmos *corpora* (Acidentes e Terremotos). Experimentos adicionais utilizando um *corpus* de 1040 artigos do *Wall Street Journal* WSJ também foram realizados. Os resultados do modelo original de Barzilay e Lapata (2008) na configuração Sintático+ Saliência+ foram utilizados como *baseline*. Os resultados mostraram que o novo modelo superou o modelo original para os *corpora* WSJ e Terremotos, mas manteve o mesmo desempenho para o *corpus* Acidentes. Para o *corpus* WSJ, o novo modelo obteve acurácia de 88%, enquanto a *baseline* obteve acurácia de 85,7%. Para o *corpus* Terremotos, o novo modelo obteve acurácia de 86,5%, enquanto a *baseline* obteve acurácia de 83,6%. Para o *corpus* Acidentes, o novo modelo obteve acurácia de 89,9%, enquanto a *baseline* obteve acurácia de 89,4%.

2.1.7. Modelo de Coerência Local Baseado em Grafo

Guinaudeau e Strube (2013) identificaram algumas desvantagens do modelo grade de entidades, como: esparsidade de dados, dependência de um domínio e complexidade computacional. Em busca de solução para esses problemas, os autores propuseram representar as entidades de um texto em grafos e então aplicar medidas de centralidade nos nós desse grafo para modelar coerência local.

A Figura 2.4 apresenta um exemplo de grafo bipartido $G = (V_s, V_e, L, w)$ que é definido por dois conjuntos independentes de nós – que corresponde ao conjunto de sentenças V_s e o conjunto de entidades V_e do texto – e um conjunto de arestas L associadas aos pesos w. Um peso $w(e_j, s_i)$ depende do papel gramatical da entidade e_j na sentença s_i (Sujeito (S) = 3, Objeto (O) = 2 e Outro (X) = 1). Em contraste com o modelo grade de entidades que contém informações sobre entidades ausentes, o modelo baseado em grafo contém somente informações das entidades que estão presentes nas sentenças.

Os autores afirmam que com essa estrutura, as informações de transições das entidades são suficientes para computar coerência local, tornando o vetor de características e a etapa de treinamento desnecessários.

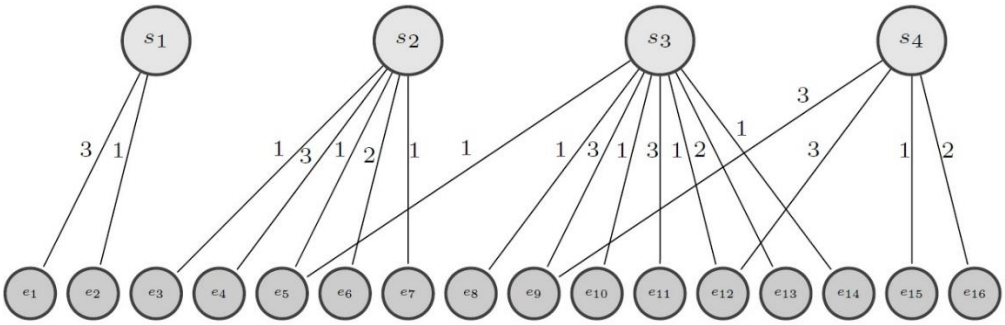

Figura 2.4. Exemplo de grafo bipartido (Extraido de Guinaudeau e Strube (2013))

A partir do grafo bipartido, diferentes tipos de projeções chamadas de "*one-mode*" podem ser aplicadas para o conjunto de nós de sentença V_s para representar as conexões existentes entre as sentenças do grafo. Essas projeções resultam em grafos em que os nós correspondem as sentenças e uma aresta é criada entre dois nós se a sentença correspondente tiver ao menos uma entidade em comum. Ao contrário do grafo bipartido, as projeções *one-mode* são direcionadas seguindo a ordem das sentenças do texto, portanto, uma aresta pode existir partindo da primeira para a segunda sentença, enquanto o inverso não é possível.

Neste modelo, os autores definiram três tipos de grafos projetados (P_U, P_W e P_{Acc}) dependendo de como o cálculo de peso associado às arestas é feito. No grafo P_U o peso é um valor binário, o valor 1 é atribuído à aresta se duas sentenças têm pelo menos uma entidade em comum (Figura 2.5). Em P_W, os pesos são atribuídos conforme o número de entidades compartilhadas por duas sentenças (Figura 2.6). Em PAcc a informação sintática é considerada para o cálculo do peso das arestas, os pesos são iguais a:

$$W_{ik} = \sum_{e \in E_{ik}} \omega(e, s_i) \cdot \omega(e, s_k) \,,$$

onde E_{ik} é o conjunto de entidades compartilhadas por s_i e s_k. A distância entre as sentenças s_i e s_k também podem ser adicionada no cálculo do peso para diminuir a importância das ligações que existem entre as sentenças não adjacentes. Nesse caso, os pesos dos grafos projetados são divididos por $k - i$.

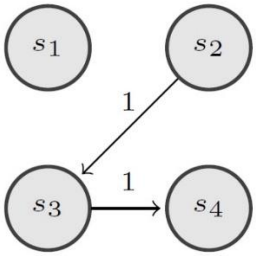

Figura 2.5. Projeção one-mode P_U

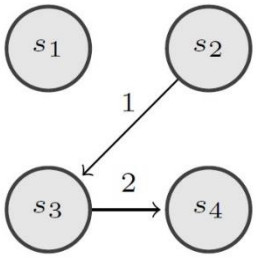

Figura 2.6. Projeção one-mode P_W

A coerência local de um texto T pode ser mensurada por meio da aplicação da medida de centralidade *average outdegree* em uma projeção P. Essa medida foi escolhida pelos autores por ter uma complexidade computacional baixa ($\theta\left(\frac{N*(N-1)}{2}\right)$ para um documento composto por *N* sentenças) comparado com outras medidas. Também foi escolhida por permitir avaliar a forma em que uma sentença está ligada, em termos de entidades do discurso, com as outras sentenças do texto. Formalmente coerência local de um texto *T* é igual a:

$$LocalCoherence(T) = AvgOutDegree(P) = \frac{1}{N}\sum_{i=1..N} OutDegree(s_i),$$

onde *OutDegree(s_i)* é a soma dos pesos das conexões feitas a partir de s_i e *N* é o número de sentenças de um texto *T*.

Para a avaliação do modelo baseado em grafo foram feitos três experimentos: (1) ordenação de sentenças, (2) classificação de coerência de textos e (3) avaliação de legibilidade. O experimento (1) consiste em ranquear diferentes ordenações de sentenças de um documento, como proposto por Barzilay e Lapata (2008) e Elsner e Charniak (2011). Duas tarefas foram feitas nesse experimento: (a) discriminação, que consiste em comparar um documento com uma permutação aleatória de suas sentenças, onde um valor de coerência local é atribuído para o documento e para a versão com sentenças permutadas, e o modelo é considerado correto se a nota atribuída para o documento for maior que a nota da permutação. (b) inserção: o modelo é avaliado quanto a habilidade em recuperar a posição original de uma sentença previamente removida de um documento, para isso, cada sentença é removida e uma nota de coerência local é atribuída para todas as possíveis reinserções da sentença no documento. A saída do modelo é considerada correta se a maior nota de coerência local for atribuída ao documento que teve a sentença inserida na posição correta. Para o experimento (1) foi utilizado um *corpus* formado por textos jornalísticos e também as três projeções (P_U, P_W, P_{Acc}) de grafos foram avaliadas. O melhor resultado para a tarefa de (1) discriminação foi com o grafo P_{Acc}, superando os resultados do modelo de Barzilay e Lapata (2008) e se aproximando do modelo de Elsner e Charniak (2011). Na tarefa de (2) inserção o melhor resultado foi obtido com o grafo P_W, superando os resultados do modelo de Elsner e Charniak (2011).

Para o experimento (2) foi utilizado o *corpus* composto de textos jornalísticos proposto por Barzilay e Lapata (2008) e o melhor resultado foi alcançado com o grafo P_U.

Assim como Barzilay e Lapata (2008), o experimento (3) buscou avaliar a legibilidade comparando um texto com sua versão para crianças. O *corpus* utilizado foi o mesmo do modelo grade de entidades e o melhor resultado obtido foi com o grafo P_{Acc}.

A descrição completa da construção dos grafos e o detalhamento dos resultados dos experimentos estão disponíveis em Guinaudeau e Strube (2013).

2.1.8. Modelo Combinado de Estrutura Retórica e Grade de Entidades

Dias et al., (2014) propuseram um modelo que combina a Teoria de Estrutura Retórica (RST) (Mann e Thompson, 1988) com a Grade de Entidades (Barzilay e Lapata, 2008), esse modelo buscou aprender os padrões de distribuição das entidades e as informações organizacionais de um discurso para ser capaz de distinguir entre textos coerentes e incoerentes.

Na teoria de estrutura retórica as Unidades Elementares do Discurso (EDUs) são conectadas por relações retóricas, visando discursos coerentemente organizados. O papel dos núcleos (N) ou satélites (S) é atribuído a cada EDU. Os núcleos contêm as mais importantes partes da informação nas relações e são consideradas mais relevantes do que os satélites. Por outro lado, os satélites contem informação adicional que ajuda o leitor na intepretação dos núcleos.

A Tabela 2.8 apresenta parte de um texto dividido em EDUs. Esse trecho é usado para exemplificar a análise RST, mostrada na Figura 2.7.

Tabela 2.8. Parte de um texto dividido em EDUs, adaptado de Dias et al., (2014).

(1) Muitas das atitudes “corajosas” de Almir – o Pernambuquinho – foram ditadas pelo medo. (2) Poucas pessoas sabem disso, (3) mas isso é verdade.

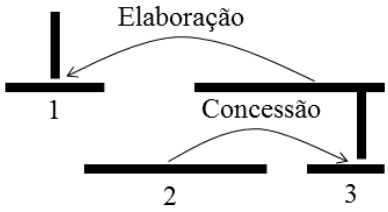

Figura 2.7. Diagrama que representa as relações RST do texto da Tabela 2.8, adaptado de Dias et al., (2014).

EDU (1) apresenta a ideia central do discurso, que é o medo que influencia na maneira de Almir agir. Entretanto, EDUs (2) e (3) indica que poucas pessoas sabem sobre a característica do personagem e o fato que essa característica é real. A relação entre as EDUs (1), (2) e (3) ocorrem como elas são reconhecidas no discurso. EDUs (2) e (3) são identificadas como componentes de uma relação de ELABORAÇÃO da afirmação (1). EDU (1) corresponde com o núcleo da relação ELABORAÇÃO (núcleo é indicado por uma linha vertical e pela ponta da

seta), enquanto EDUs (2) e (3) constituem o satélite da relação. EDU (3) é o núcleo e EDU (2) é o satélite da relação CONCESSÃO.

O modelo de Dias et al., (2014) seguiu a abordagem de Barzilay e Lapata (2008), porém não foi utilizado a informação de correferência. A grade é formada por sentenças (linhas) e entidades (colunas), em que cada célula é preenchida com as relações RST que a entidade tem com as sentenças, também é especificado a nuclearidade correspondente. A Tabela 2.9 mostra a grade extraída do trecho apresentado na Tabela 2.8 com as relações RST dadas no diagrama da Figura 2.7.

Tabela 2.9. Grade de Relações RST para o texto da Tabela 2.8. Adaptado de Dias et al., (2014).

	atitudes	Almir	Pernambuquinho	medo	pessoas	verdade
S1	elab.Nuc	elab.Nuc	elab.Nuc	elab.Nuc	-	-
S2	-	-	-	-	conces.Sat elab.Sat	conces.Nuc elab.Sat

Assim como em outros modelo que utilizam grade de entidades (Barzilay e Lapata, 2008; Freitas, 2013; LIN et al., 2011) um vetor de características é extraído da grade para representar o texto e também é utilizado para calcular as probabilidades de transições entre relações retóricas agregadas a nuclearidade por meio de algoritmos de aprendizagem de máquina.

O modelo foi avaliado por meio de um experimento realizado em formato parecido com o experimento (1 – ordenação de sentenças) de Barzilay e Lapata (2008), utilizando, um *corpus* jornalístico (CSNews) composto por 140 textos escritos em português. Os resultados de alguns modelos como os de Barzilay e Lapata (2008) e Lin et al., (2011) foram utilizados como *baseline*. Os resultados mostraram que o modelo de Dias et al., (2014) superou todos os modelos.

3. Ferramentas de Auxílio à Escrita

Considerando que o contexto de desenvolvimento deste trabalho é a análise de aspectos linguísticos relacionados à coerência textual por uma ferramenta de auxílio à escrita, nesta seção são apresentados alguns sistemas disponíveis na literatura que de alguma maneira consideram a coerência como parte da análise textual. Especificamente para língua inglesa são apresentados: o sistema *Criterion*^{TM}, composto pelas ferramentas *Critique* e *E-Rater*, apresentado na Subseção 3.1; o *Intelligent Essay Assessor* e as ferramentas *WriteToLearn* e *Writing Coach*, apresentados na Subseção 3.2; e o sistema *IntelliMetric*^{TM}, apresentado na Subseção 3.3. Para a língua portuguesa, na Subseção 3.4, é apresentado o sistema SciPo, incluindo o Módulo de Análise de Coerência (MAC), no qual a proposta deste trabalho está diretamente inserida.

3.1. Criterion^{TM}

Ao observar que o melhor modo que o aluno possui para melhorar sua escrita é escrever um texto, submetê-lo para análise de seu professor, receber a crítica, analisar os pontos destacados e repetir todo o processo até que não se tenha mais melhorias a serem realizadas, a *Education Testing Service* - ETS[2] desenvolveu o sistema *Criterion*^{TM} *Online Essay Evaluation Service* (Burstein et al., 2004), que tem como finalidade aliviar a carga dos professores, permitindo que os mesmos possuam mais tempo para instruir e ensinar os alunos, constituindo assim um auxílio ao aprendizado e não um substituto do profissional.

Esse sistema fornece *feedback* ao aluno com o diagnóstico da análise realizada e, com base nesses resultados, calcula uma pontuação que, no geral, é semelhante à análise e avaliação realizada pelos professores de sala de aula. Para que isso ocorra, o *Criterion* dispõe de duas ferramentas baseadas em Processamento de Linguagem Natural (PLN): a *Critique Writing Analysis Tools* e o *E-rater*^{TM} *score engine*, descritas a seguir.

[2] http://www.ets.org/

3.1.1. Critique Writing Analysis Tools

Critique é uma ferramenta que utiliza PLN e técnicas de aprendizagem de máquina para avaliar e fornecer *feedback* ao aluno sobre as seguintes características indesejáveis do texto:

- Gramática: identifica erros de concordância, erros de formação verbal, uso de palavras erradas, falta de pontuação e erros ortográficos.
- Palavras confusas: identifica o uso incorreto de palavras homófonas (palavras que possuem a mesma pronúncia, mas que são escritas de forma diferente), identificando a melhor opção de acordo com o contexto de uso. Um exemplo de palavra homófona é apresentado na Figura 3.1, onde a palavra *right* na língua inglesa tem a mesma pronúncia de *write* (usada incorretamente no exemplo).
- Estilo: identifica aspectos de estilo que podem ser revistos pelo escritor para melhorar o texto, tais como uso de frases na voz passiva, frases muito longas ou muito curtas e repetição excessiva de palavras.

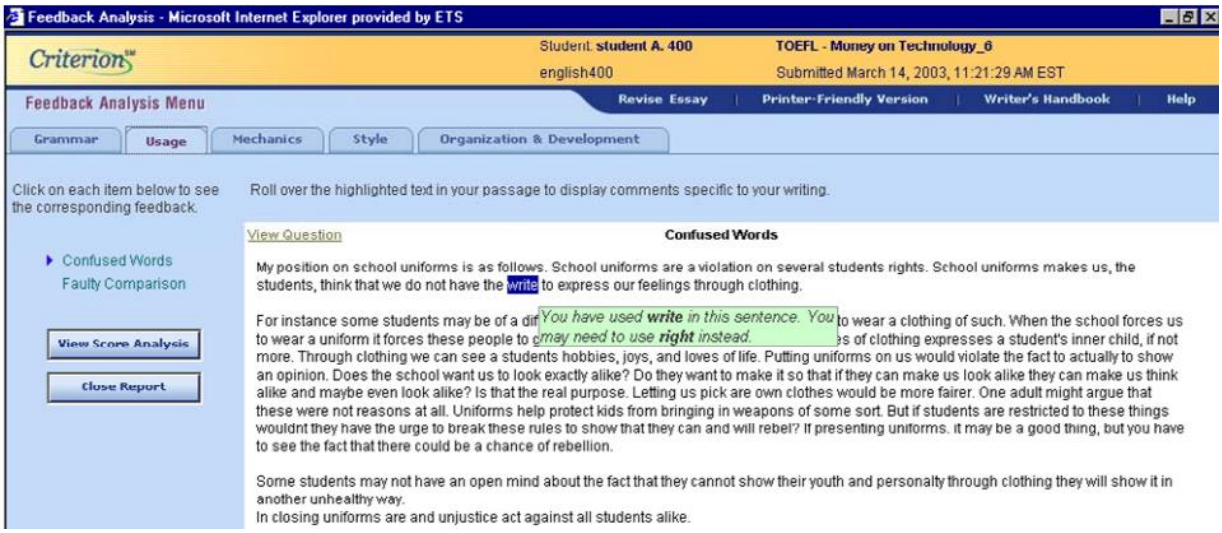

Figura 3.1. Exemplo da identificação de Palavras Confusas.[3]

3.1.2. The E-rater™ Score Engine

O *E-rater* (Burstein et al., 1998; Attali e Burstein, 2006) é classificado como um sistema de pontuação (*scoring*) que extrai características linguísticas a partir de redações curtas (*essays*) e calcula uma nota, buscando simular a nota que seria atribuída por um avaliador humano. Para tanto, o sistema utiliza aprendizagem de máquina sobre dez características automaticamente extraídas com auxílio de diferentes ferramentas de PLN. As características são divididas em

[3] Disponível em http://www.toeic.co.nz/pdfs/HowtheCriterionServiceWorks.pdf. Acessado em 21/01/2016

quatro grupos, a saber: (1) uso correto da gramática, aspectos mecânicos (pontuação, capitalização e ortografia) e estilísticos do texto, (2) organização e desenvolvimento, (3) complexidade lexical e (4) uso do vocabulário em cada categoria da pontuação (Burstein et al., 2004).

As características do Grupo (1) são construídas basicamente pela contagem de erros observados em cada uma das categorias do grupo. O *Criterion* utiliza esse grupo para fornecer *feedback* e comentários sobre cada uma das categorias de erros para os usuários.

Para as características do Grupo (2), Attali e Burstein(2006) afirmam que há muitas maneiras possíveis de usar os elementos discursivos identificados pelas ferramentas de análise de textos, dependendo do tipo de indicação e da estratégia de discurso que é procurado pelo professor ou avaliador. Os textos desenvolvidos em sala de aula muitas vezes são padronizados (persuasivos ou informativos). Ambos os gêneros costumam seguir uma estratégia de discurso que exige pelo menos uma "*thesis statement*" (uma sentença ou parágrafo que responde a pergunta que é explorada na redação), várias ideias principais e de apoio, e uma conclusão. A característica de organização geral de redações foi projetada para esses dois gêneros de escrita. Ela assume uma estratégia de escrita que inclui: um parágrafo introdutório, pelo menos três parágrafos de desenvolvimento do assunto, e um parágrafo de conclusão. A característica de organização mede a diferença entre esse modelo estrutural ideal e os componentes discursivos reais encontrados na redação. Elementos em falta afetam diretamente a pontuação atribuída pela ferramenta à redação, por outro lado, a identificação de elementos além do mínimo não contribui para a pontuação. A segunda característica do grupo 2, desenvolvimento, é derivada do módulo de organização e desenvolvimento da ferramenta *Criterion*, que mede o desenvolvimento dos elementos do discurso baseado em seus tamanhos médio.

Duas características fazem parte do Grupo (3) de complexidade lexical. A primeira é a medida de nível de vocabulário, que é um índice de frequência normalizada de todas as palavras da redação. A segunda característica é chamada de "tamanho da palavra" e é baseada na média de quantidade de caracteres das palavras da redação.

No Grupo 4, o conteúdo lexical da redação é avaliado comparando-se as palavras do texto com as palavras encontradas em uma amostra de redações divididas em seis categorias de pontuação. Para isso, o vocabulário da redação é convertido em um vetor cujos elementos representam a frequência de cada palavra na amostra de redações. A análise do vetor é feita como segue. A redação e um conjunto de redações de treinamento são convertidos em vetores. Os elementos desses vetores correspondem aos pesos de cada palavra de uma redação. O peso W de cada palavra *i* na redação é dado por:

$$W_i = (F_i / MaxF) * \log(N / N_i)$$

Na qual F_i é a frequência da palavra *i* na redação, MaxF é a frequência máxima de uma palavra da redação, N é o número total de redações do conjunto de treinamento e N_i é o total de redações do conjunto de treinamento que contém a palavra *i*.

Para cada redação, seis medidas de cosseno são computadas para o vetor de pesos das palavras. Os seis valores indicam o grau de similaridade entre as palavras usadas na redação e as palavras usadas na amostra de redações de acordo com cada uma das categorias de pontuação.

3.2. Intelligent Essay Assessor

O *Intelligent Essay Assessor* (*IEA*) (Foltz et al., 2013) foi desenvolvido pela empresa *Pearson Knowledge Technologies*, e visa analisar alguns aspectos de textos dissertativos, tais como semântica, erros de ortografia e gramaticais, entre outros, atribuindo uma pontuação final a fim de auxiliar o escritor no processo de escrita.

O modelo de pontuação do IEA utiliza técnicas de aprendizagem de máquina para identificar como os anotadores humanos realizam a análise e a pontuação dos textos. A ferramenta foi treinada a partir de um conjunto pré-selecionado de textos corrigidos, a partir do qual foram extraídas as características referentes ao desempenho dos elaboradores, especialmente quanto ao conhecimento e domínio do vocabulário e recursos linguísticos.

Partindo do pressuposto que a qualidade da redação de um aluno pode ser caracterizada por uma série de propriedades que medem seu conhecimento sobre o domínio do conteúdo, gramática, escrita e qualidade de raciocínio, o IEA identifica essas características utilizando um conjunto de recursos e as combina para obter uma a pontuação. A Figura 3.2, extraída de Foltz et al. (2013), mostra alguns dos recursos utilizados pelo IEA e como eles se relacionam para medir o desempenho da escrita dos alunos.

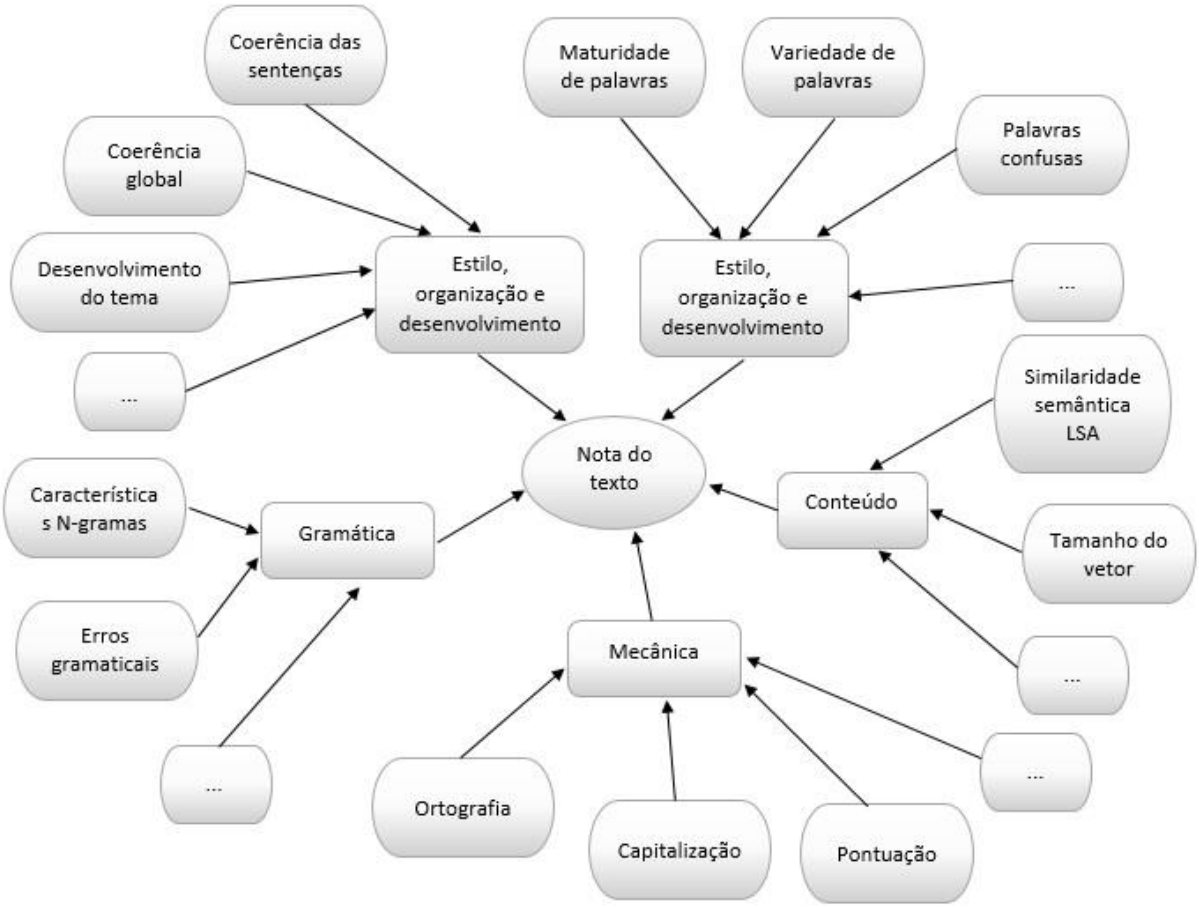

Figura 3.2. Recursos utilizados pelo Intelligent Essay Assessor adaptado de Foltz et al. (2013)

Baseado em LSA (Landauer et al., 1998), o IEA utiliza essa tecnologia para obter medidas de conteúdo, organização e recursos baseados no desenvolvimento da escrita, ou seja, ele compara a semelhança semântica do texto do aluno com um *corpus* de qualidade já conhecida. A pontuação é atribuída ao texto com base na pontuação dos textos mais semanticamente relacionados. Segundo os autores, a pontuação atribuída pelo sistema é muito próxima à pontuação atribuída por anotadores humanos. Outra aplicação baseada em LSA é usada para comparar o conteúdo de frases individuais e com o intuito de avaliar a coerência, bem como a semelhança de construção semântica do conteúdo de frases e parágrafos com as amostras previamente anotadas.

Além da avaliação em redações dissertativas, o IEA é utilizado para a atribuição de notas em questionários, por meio da avaliação de respostas discursivas. Para isso, o IEA utiliza uma variedade de controles estatísticos e probabilísticos para determinar com precisão a pontuação do aluno com base nas características das respostas em que o sistema foi treinado. A

resposta que parece estar fora do tópico ou ser incomum é direcionada para um avaliador humano, para que este realize as anotações necessárias.

Respostas muito curtas, com uma ou duas frases, são consideradas um desafio para o IEA, pois possui pouca informação para analisar o conhecimento e a habilidade do aluno, o que também pode ser agravado nos casos em que a resposta possua muitos erros ortográficos.

Outro ponto importante, é que o IEA permite que os alunos pratiquem a compreensão da leitura e o aprimoramento da escrita por meio da avaliação automática de resumos. Essa ferramenta permite que o aluno elabore um resumo a partir de um texto sugerido e, após análise, o sistema aponta as partes do texto que o aluno pode não ter compreendido totalmente, solicitando ao aluno que releia o texto, repense e reescreva o resumo.

3.2.1. Aplicações do *Intelligent Essay Assessor*

WriteToLearn é uma ferramenta que além de fornecer a análise instantânea dos elementos textuais propostos, serve para melhorar continuamente a escrita dos estudantes, por meio da análise automática de textos e constantes *feedbacks*, a fim de que o texto tenha a melhor redação e, consequentemente, obtenha a maior nota possível.

A ferramenta foi aplicada em diversas pesquisas que identificaram melhora significativa na média geral dos alunos. Entre elas, pode-se destacar o exame realizado no estado americano Dakota do Sul com os alunos da quinta, sétima e décima série, no qual a ferramenta foi disponibilizada para prática textual e, devido ao maior número de *feedbacks* do sistema se comparado a prática normal da sala de aula, foi constado um aumento médio de 1 ponto na avaliação dos textos dos alunos (Foltz et al., 2013).

Na apresentação do resultado das análises, o sistema fornece ao aluno uma pontuação global, que é a composição da nota atribuída a seis aspectos comuns da escrita. São eles: ideias, organização, convenções, fluência sentencial, vocabulário e voz. O aluno consegue visualizar sua pontuação (de 1 a 6) na tela do sistema (Figura 3.3), e identificar os aspectos que precisam ser melhorados. Clicando sobre cada aspecto, como *ideas* e *organization*, o aluno obterá informações detalhadas sobre como melhorar o seu desempenho nesse aspecto.

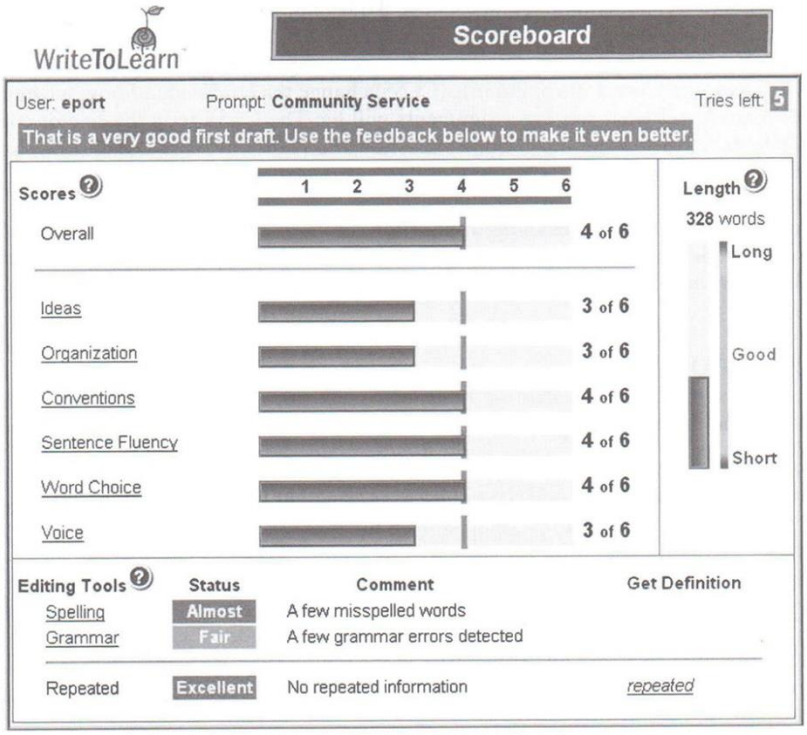

Figura 3.3. Exemplo de feedback apresentado pelo WriteToLearn. (Foltz et al., 2013)

A ferramenta *WriteToLearn* também estimula a capacidade de leitura, por meio da avaliação dos resumos elaborados pelos alunos. Assim como o IEA, a ferramenta disponibiliza um texto para leitura e solicita que o estudante realize um resumo. Após o resumo ser submetido para análise, o sistema fornece, além da pontuação atingida, a sua avaliação sobre o quão bem o aluno cobriu o conteúdo de cada seção da leitura proposta, realizando sugestões de melhorias, comentários sobre frases desnecessárias e redundantes e trechos identificados como cópia do texto original. Dessa forma, os alunos são incentivados a observar os pontos destacados e refazê-los para que possam aumentar sua pontuação.

Outra ferramenta de auxílio à escrita baseada no IEA é a *Writing Coach* (Foltz et al., 2013). Como pode ser observado na Figura 3.4, a ferramenta possibilita que o aluno visualize a avaliação do texto por parágrafos, o que facilita o entendimento e a correção por parte do aluno. As seguintes características são avaliadas pelo sistema:

- *Topic Focus*: analisa e identifica as sentenças que têm e que não têm relação semântica com o tema.
- *Topic Development*: analisa o desenvolvimento de ideias no parágrafo.
- Tamanho e estrutura da sentença.
- Vocabulário: uso de adjetivos vagos, palavras repetidas, uso incorreto de pronomes, erros de ortografia e gramática, e redundância.

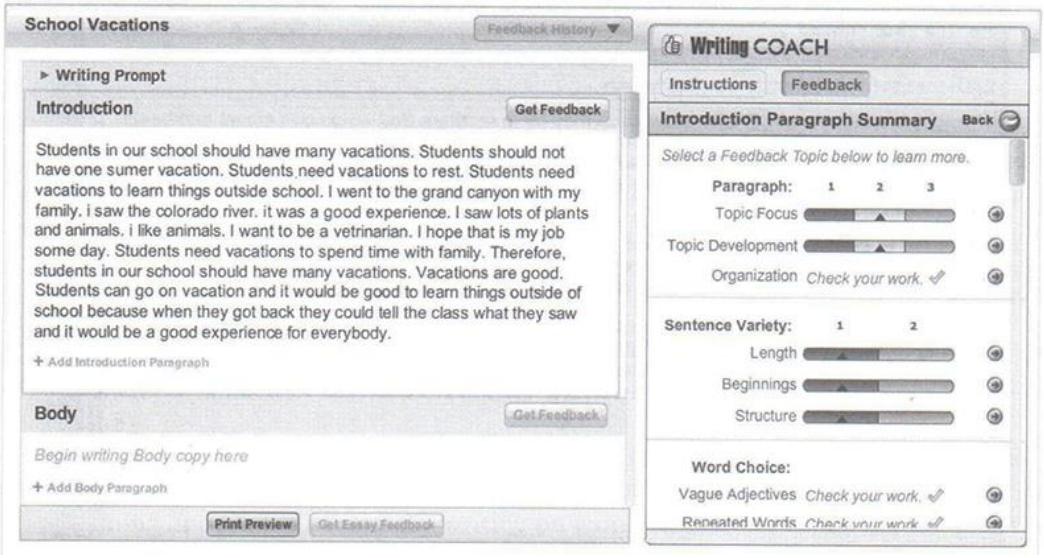

Figura 3.4. Exemplo de feedback apresentado pela ferramenta Writing Coach. (Foltz et al., 2013)

3.3. IntelliMetricTM

IntelliMetric é considerada uma ferramenta de auxílio à escrita, pois estimula a capacidade do autor na construção de redações, por meio da análise textual e do *feedback* instantâneo. Denominado como um sistema inteligente de pontuação, a ferramenta realiza o processo de análise e avaliação textual, instruindo o autor a melhor nos pontos destacados. Assim, pode-se dizer que a ferramenta contribui, principalmente, para a redução do custo e do tempo de análise do texto, se comparada à análise convencional realizada por um especialista humano.

Segundo Schultz (2013), o *IntelliMetric* é teoricamente fundamentado em um modelo cognitivo de processamento de informações e compreensão, chamado de "modelo baseado em cérebro", que busca simular o processo de avaliação de redações feito por seres humanos. Para isso, o modelo é treinado com as características identificadas de textos que receberam altas pontuações de avaliadores humanos. O *IntelliMetric* captura características ou funções das respostas pertinentes aos pontos destacados pelos avaliadores e aplica essa inteligência nas análises.

IntelliMetric analisa mais de 400 características sintáticas, semânticas e discursivas. Para isso utiliza técnicas de inteligência artificial, processamento de linguagem natural e estatísticas, conforme representado na Figura 3.5:

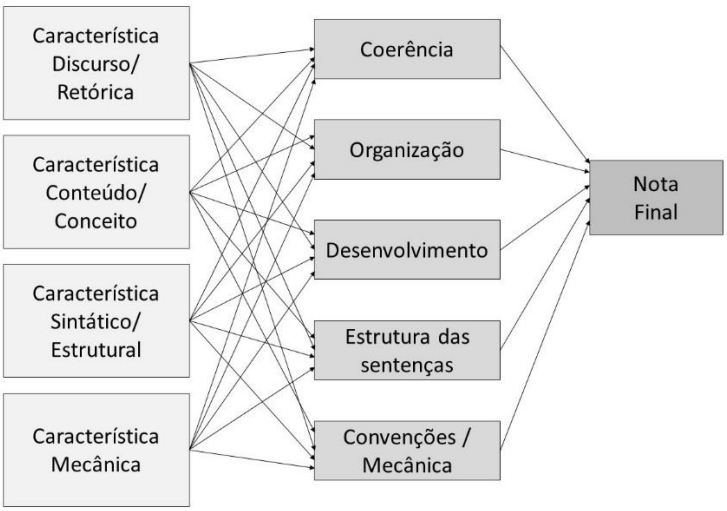

Figura 3.5. Características e aspectos analisados pelo IntelliMetric adaptado de Foltz et al. (2013)

Assim como a ferramenta *WriteToLearn*, a *IntelliMetric* fornece uma pontuação global, bem como as pontuações em cinco aspectos principais: Coerência, Organização, Conteúdo e Desenvolvimento, Uso de linguagem e estilo, e Mecânica e Convenções. Esses recursos se dividem em outras duas características:

- Conteúdo: avaliação do tema abordado e a amplitude do conteúdo. Exemplo: análise de vocabulário, conceitos, coesão, coerência, lógica do discurso, fluidez de transição e relação entre as partes da resposta, entre outros.
- Estrutura: avaliação da gramática, ortografia, sentença completa, pontuação, entre outros.

Ainda de acordo com Schultz (2013), o *IntelliMetric* utiliza um processo multifase para avaliar as redações e obter a pontuação final, conforme mostrado na Figura 3.6 e nos subitens a seguir:

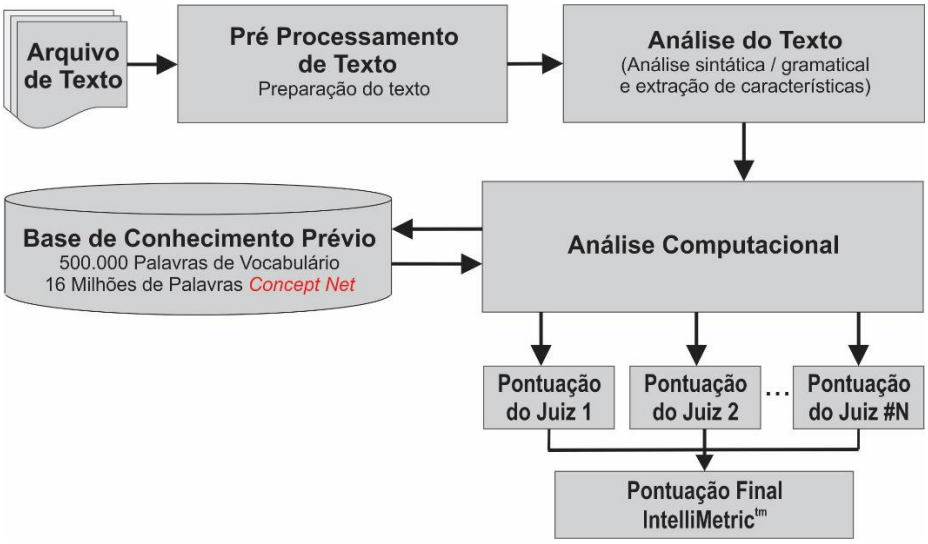

Figura 3.6. Diagrama do processo de avaliação das redações adaptado de Foltz et al. (2013)

- Etapa Arquivo de Texto: as respostas das perguntas podem estar escritas à mão ou em meio eletrônico. Nos dois casos, é necessário criar um arquivo na extensão "txt", para que assim o sistema *IntelliMetric* o aceite como entrada.
- Etapa Pré-Processamento de Texto: nessa etapa o sistema realiza a leitura do arquivo criado na extensão "txt", e prepara as informações para a próxima fase, removendo caracteres desconhecidos e corrigindo a formatação.
- Etapa Análise de Texto: nessa fase é feita a análise da estrutura sintática do texto. O sistema possui um vocabulário de 500.000 palavras e 16 milhões de conceitos, que são utilizados para formar a compreensão do texto.
- Etapa Análise Computacional (Cálculo de Informações): transformação dos dados extraídos do texto para a forma numérica, afim de suportar os modelos matemáticos (LSA, análise linear e bayesiana) empregados no sistema.
- Etapa Avaliação com Base nos Juízes Virtuais: os juízes virtuais, criados a partir de métodos estatísticos, de inteligência artificial e de aprendizagem de máquina, procuram associar as características extraídas do texto com a pontuação atribuída no conjunto de treinamento, a fim de gerar notas para textos não avaliados.
- Etapa Pontuação Final *IntelliMetric*: cálculo final da pontuação dada pelos juízes, de forma a produzir um único resultado.

Em resumo, o sistema *IntelliMetric* foi constituído buscando simular processos do cérebro humano e como ele processa o texto para desenvolver significado. Pode-se dizer que são quatro os princípios básicos que norteiam o sistema, sendo eles:

- É modelado com base no cérebro humano e sua capacidade de sintetizar informações. O sistema reproduz os processos mentais usados por especialistas humanos para anotar e avaliar o texto escrito (respostas de perguntas).
- É um motor de aprendizagem, pois aprende a avaliar com base em textos que foram analisados pelos avaliadores humano e aplica o conhecimento aprendido em novas situações.
- É indutivo, uma vez que a avaliação é feita com base em inferências e não em regras pré-definidas.
- É baseado em modelos matemáticos e estatísticos. *IntelliMetric* usa uma combinação de várias técnicas, incluindo análise linear, *bayesiana* e LSA.

3.4. SciPo – Scientific Portuguese

Esta subseção apresenta a ferramenta de auxílio à escrita científica SciPo (Feltrim, 2004), bem como seus recursos, destacando o módulo de análise de coerência – MAC (Souza, 2011) e o sistema de detecção automática de estrutura retórica para resumos acadêmicos AZPort, uma vez que essa ferramenta e seus recursos serviram de motivação para este trabalho.

Assim como outros autores (Aluísio et al., 2001; Aluísio et al., 2005; Kinnunen et al., 2012), Feltrim (2004) observou as dificuldades dos escritores em produzir bons textos científicos (por exemplo, artigos, dissertações e teses). O processo de escrita, de uma forma geral, demanda do escritor a articulação adequada de diferentes aspectos do texto, como o tipo de informação a ser incluída, a forma de apresentação dessas informações, o uso de padrões lexicais adequados, o uso correto de tempos verbais, e a organização textual. Essa dificuldade é aumentada quando escritor é iniciante na escrita científica, pois ele não estará familiarizado com as convenções desse gênero.

O SciPo foi pioneiro como ferramenta computacional de auxílio à escrita científica em língua portuguesa, e foi inspirado na abordagem baseada em *corpus* proposta por Aluísio (1995) para a ferramenta AMADEUS, que é voltada para a escrita científica em inglês.

O AMADEUS, projetado para auxiliar escritores não nativos da língua inglesa, enfatiza a definição de estratégias para a organização do texto escrito, bem como a reutilização de exemplos, apoiado na recuperação de casos similares a partir da especificação da estrutura retórica (ou esquemática) desejada e de uma base de exemplos manualmente anotados. Essa mesma abordagem é adotada pelo SciPo, porém, ajustada e incrementada para auxiliar escritores nativos do português na escrita de teses e dissertações.

O auxílio fornecido pelo SciPo se dá por meio de dois processos distintos de interação, sendo eles:

1. Processo *top-down*, no qual o escritor parte do planejamento da estrutura para a escrita do texto. Esse modo de interação é similar ao que ocorre no AMADEUS.
2. Processo *bottom-up*, no qual o usuário submete um texto já escrito para sistema, que então detecta automaticamente a estrutura retórica utilizada.

Cabe destacar que os processos 1 e 2 são apenas duas formas distintas de se iniciar o processo de crítica da estrutura, visando dirigir o usuário na produção de um texto com uma estrutura adequada. Após esse processo, o usuário pode se beneficiar de outros recursos da ferramenta, que inclui funções de suporte à escrita do texto de acordo com a estrutura refinada pelo usuário.

As funções de suporte ao reúso de *corpus* e de crítica à estrutura do texto disponíveis no SciPo foram adaptadas do ambiente AMADEUS, sendo elas: (a) navegação das bases de exemplos, (b) pesquisa das ocorrências de determinado componente ou estratégia retórica, (c) apoio à composição de estruturas retóricas, (d) crítica da estrutura construída e (e) recuperação dos exemplos com estruturas similares à estrutura construída.

Essas funções utilizam quatro fontes de conhecimento para auxiliar o usuário com modelos, exemplos e regras. As funções (a), (b) e (e) contam com uma base de exemplos de Resumos e Introduções que foram obtidas a partir de um processo de anotação manual de um *corpus* contendo 52 instâncias de estruturas retóricas de Resumos autênticos, contendo o texto do resumo anotado com suas características retóricas (componentes, estratégias, expressões padrão e marcadores discursivos) e, ainda, observações críticas sobre sua estrutura, e outro *corpus* abrangendo 48 instâncias de estruturas retóricas de Introduções autênticas, representadas de forma similar à descrita para os Resumos. A função (c) faz uso dos modelos estruturais apresentados nas Figuras 3.7 e 3.8 e extraídas de Feltrim (2004). A função (d) se beneficia de um conjunto de regras de críticas, que são distribuídas em duas categorias: regras que consideram desvios de conteúdo (falta de componentes na estrutura) e desvio de ordem (ordem de ocorrência dos componentes). Em ambas as categorias as regras se diferenciam em desvios graves, que são apresentados ao usuário como críticas, e desvios leves, que são apresentados como sugestões. Além da função (e) utilizar a base de exemplos, ela também conta com uma base de regras e medidas de similaridade que são baseadas na medida de similaridade *nearest neighbors matching* (Kriegsman e Barletta, 1993) e em regras de similaridade entre listas (casamento de padrões). Essas regras são usadas para recuperar exemplos com estrutura retórica similar à especificada pelo usuário.

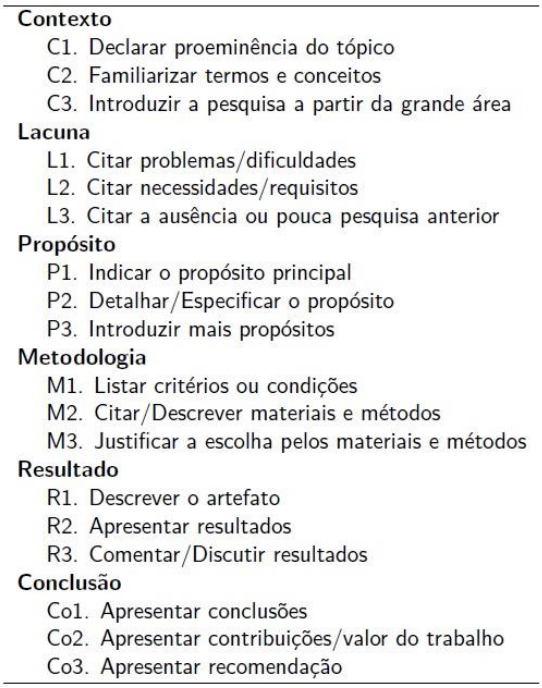

Figura 3.7. Modelo de estrutura retórica de resumos (Feltrim, 2004).

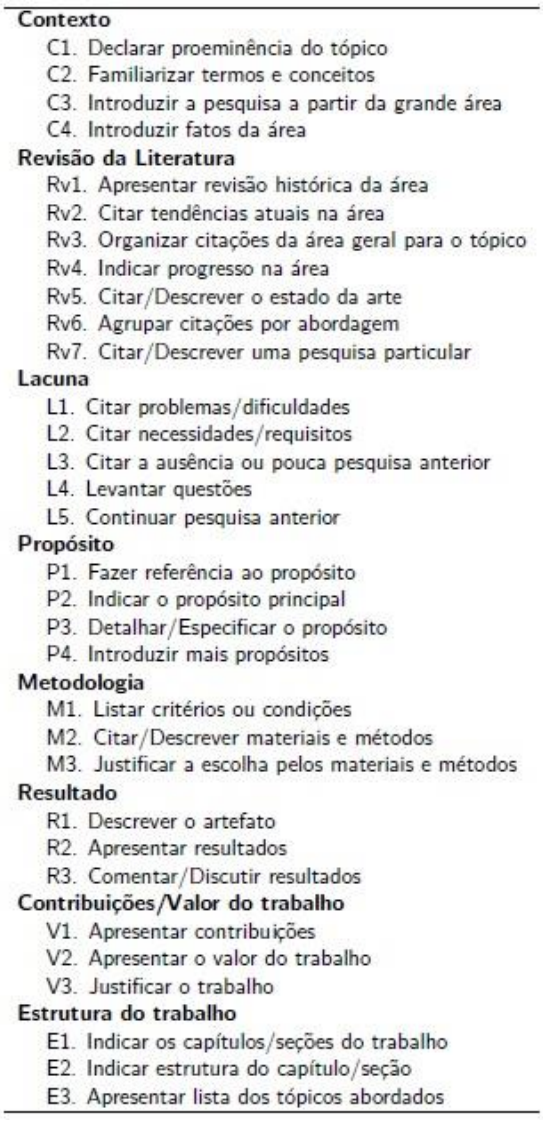

Figura 3.8. Modelo de estrutura retórica de introduções (Feltrim, 2004).

Para a implementação do processo *bottom-up* foi criado o classificador textual AZPort, que detecta automaticamente a estrutura retórica de resumos de acordo com os componentes mostrados na Figura 3.7. A seção Resumo foi escolhida por constituir um estudo de caso suficiente para a avaliação inicial da viabilidade do uso desse tipo de classificador em um sistema como o SciPo, além do fato do Resumo conter componentes semelhantes aos da Introdução, possibilitando, dessa forma, que a extensão do classificador para introduções pudesse ocorrer de forma escalável. O AZPort foi construído seguindo a abordagem de Teufel e Moens (2002), que propõem a segmentação automática de um texto científico em "zonas argumentativas", de modo que partes do texto possuem certos papéis retóricos dentro do texto como um todo. A detecção automática das zonas argumentativas é realizada por meio de um classificador estatístico que implementa o algoritmo *NaiveBayes*.

Após o usuário ter definido a estrutura do texto, seja ela construída a partir dos recursos disponíveis no processo *top-down* ou detectada automaticamente pelo processo *bottom-up*, ele receberá críticas e/ou sugestões a respeito do conteúdo (componentes presentes/ausentes) e da ordem dos componentes, até que, pelas regras do sistema, tenha-se uma estrutura aceitável.

Uma vez que a construção e refinamento da estrutura do Resumo tenha sido finalizada, o usuário pode se beneficiar de outros recursos disponíveis no SciPo, como o revisor ortográfico e gramatical, que pode ser utilizado no ambiente da mesma forma que é feito em um editor de texto comum, e do MAC, que é responsável por gerar sugestões que auxiliem o escritor a melhorar a coerência do resumo.

A Figura 3.9 apresenta uma visão geral da arquitetura do SciPo, mostrando como os diferentes processos se relacionam com as bases de conhecimento.

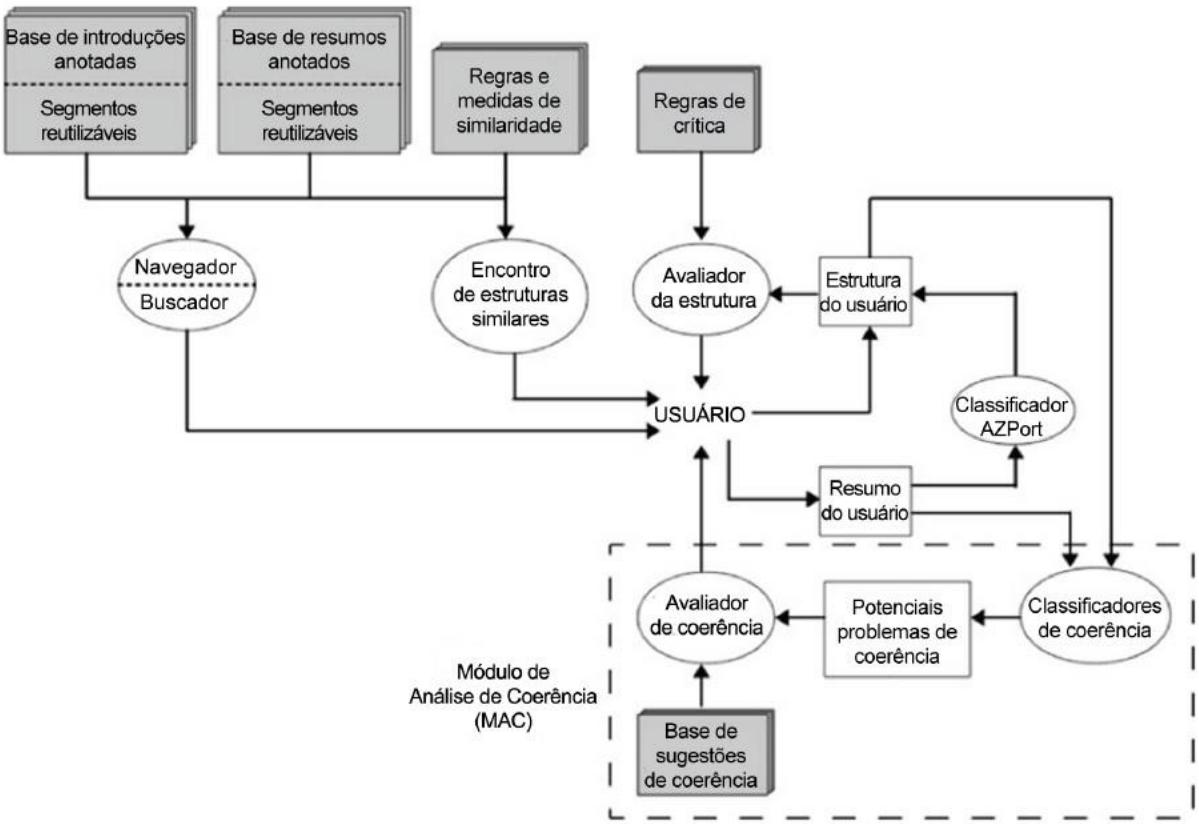

Figura 3.9. Arquitetura do SciPo com o módulo de análise de coerência (MAC).

3.4.1. MAC – Módulo de Análise de Coerência

O MAC (Souza, 2011; Souza e Feltrim, 2013) foi proposto e desenvolvido como um módulo adicional da ferramenta *SciPo*, com o objetivo de auxiliar o usuário na escrita em relação a aspectos relacionado à coerência semântica. No MAC isso é feito por meio de análise de relacionamento semântico entre as sentenças do resumo e de sugestões referentes à coerência, tendo como base o texto do resumo e sua estrutura retórica. A análise é elaborada sobre três

tipos diferentes de relacionamento semântico entre as sentenças, que são denominados como dimensões, sendo elas:

(i) Dimensão Título, que verifica o relacionamento semântico entre as sentenças do componente Propósito com o Título do resumo;

(ii) Dimensão Propósito, que verifica o relacionamento semântico dos componentes Metodologia, Resultado e Conclusão com o componente Propósito; e

(iii) Dimensão Lacuna-Contexto, que verifica o relacionamento semântico entre os componentes Lacuna e Contexto.

Uma quarta dimensão, chamada Quebra de Linearidade, foi proposta por Souza (2011), mas não chegou a ser incluída no MAC. Essa dimensão busca identificar problemas de coerência local que se caracterizam pela dificuldade em se estabelecer uma ligação clara da sentença atual com a sentença anterior, demandando assim maior esforço cognitivo para a interpretação do texto. Segundo o autor, os resultados obtidos na implementação dessa dimensão foram pouco satisfatórios e isso poderia se dever tanto à dificuldade intrínseca da tarefa, quanto ao método utilizado pelo MAC para análise desse aspecto de coerência. Assim, para essa dimensão, foi sugerida a experimentação de outros modelos de coerência, tal como o modelo grade de entidades. Essa a lacuna motivou o trabalho de Freitas (2013), bem como este trabalho.

Para medir a força do relacionamento semântico entre as sentenças, o MAC utiliza a LSA. Assim, a implementação das três dimensões foi feita com classificadores que utilizam atributos extraídos da superfície do texto e da LSA. Ao todo foram gerados cinco classificadores: um para a dimensão (i), três para a dimensão (ii) (Metodologia-Propósito, Resultado-Propósito, Conclusão-Propósito) e um para a dimensão (iii). A média da medida Macro-F dos classificadores foi de 0,875 e o desvio padrão foi de 0,039, portanto esses resultados mostraram que os classificadores foram capazes de desempenhar a tarefa proposta por cada uma das dimensões. A descrição dos atributos utilizados, a forma de seleção de atributos, bem como os resultados detalhados das avaliações podem ser encontrados em Souza (2011).

O processo de análise de coerência do MAC se inicia após a identificação e correção da estrutura retórica do resumo. O texto e a estrutura retórica são enviados para a extração de atributos e o processamento da LSA. Após a análise dos cinco classificadores, os resultados são enviados para o avaliador de coerência. Caso algum problema tenha sido detectado, a sugestão será selecionada conforme a dimensão identificada, e será apresentada ao usuário como no exemplo da Figura 3.10, que apresenta uma sugestão referente à (i) dimensão Título.

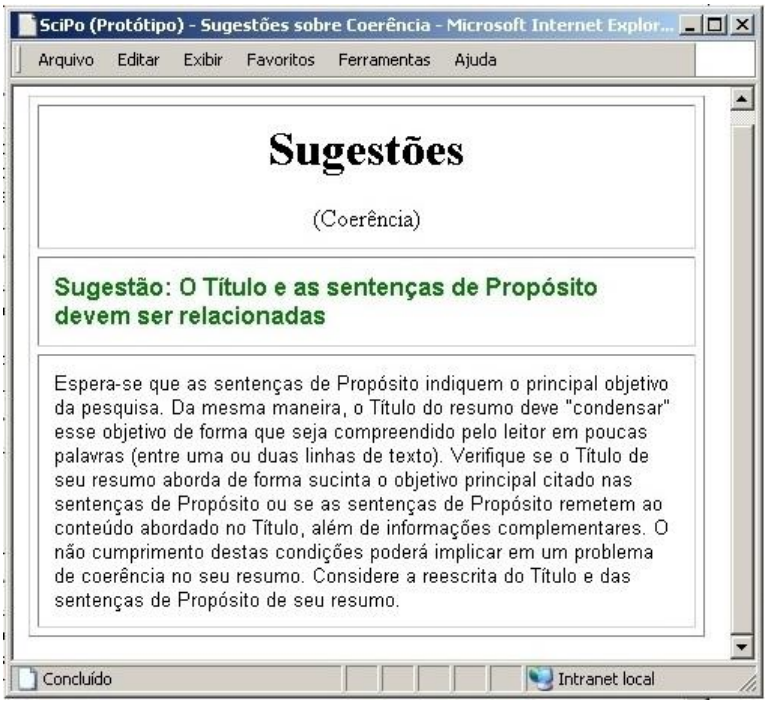

Figura 3.10. Captura de tela do MAC com sugestão a respeito da dimensão Título (Souza, 2011).

O processo de análise de coerência continua enquanto possíveis problemas de coerência são identificados e enquanto o usuário aceita as sugestões apresentadas, constituindo assim um processo cíclico. Em qualquer momento, o escritor pode rejeitar as sugestões oferecidas pelo sistema, pois, assim como dois especialistas em escrita podem discordar em questões referentes à coerência, o usuário e o MAC também podem, dado que aspectos semânticos são ambíguos por natureza (Muller et al., 2003).

Além dos experimentos de avaliação intrínseca dos classificadores, o MAC foi avaliado extrinsecamente em um experimento com usuários reais, no qual se buscou medir a eficácia das sugestões na escrita dos resumos. Primeiramente, os usuários – oito estudantes de mestrado em Ciência da Computação da Universidade Estadual de Maringá, os quais não tinham familiaridade com os conceitos dos componentes e das estruturas retóricas – realizaram as correções da estrutura retórica do resumo conforme as sugestões feitas pelo *SciPo*. Depois dessa etapa, os alunos foram apresentados ao processo de refinamento do texto com as sugestões relativas à coerência. Após utilizarem o MAC, todos os usuários responderam um questionário com questões sobre suas impressões sobre o MAC, tais como facilidade de uso, relevância das sugestões apresentadas, melhoria proporcionada pelas alterações realizadas, entre outras. A maioria dos usuários concordaram que as sugestões do MAC são relevantes e ajudam a

melhorar a coerência do resumo. Todos os resultados referentes a avaliação extrínseca do MAC podem ser encontrados em Souza (2011).

4. *Corpus* de Desenvolvimento e Avaliação

Como mencionado na Introdução, este trabalho propõe utilizar informações provenientes da estrutura retórica de resumos em conjunto com a grade de entidades para gerar sugestões que indiquem possíveis problemas de coerência local em regiões específicas do resumo. Para isso, a abordagem proposta extrai conhecimento a partir de textos já anotados, uma vez que o modelo grade de entidades faz uso de aprendizagem de máquina supervisionada. Além disso, a abordagem utiliza informações referentes à estrutura retórica. Dessa forma, um *corpus* anotado com esses dois tipos de informação é necessário tanto para o treinamento quanto para a avaliação da abordagem.

Visto que o contexto de aplicação deste trabalho é semelhante ao de Souza (2011) e ao de Freitas (2013), que é a escrita de resumos científicos por estudantes nativos do português, o *CorpusTCC,* criado por Souza e Feltrim (2011), e o *corpus* utilizado por Freitas (2013) foram empregados como *corpus* de desenvolvimento e avaliação. A seguir, são descritos o *CorpusTCC* e o *corpus* de Freitas (2013), respectivamente.

4.1. CorpusTCC

O *CorpusTCC* é formado por 385 resumos de monografias de conclusão de curso de Ciência da Computação. Os resumos foram extraídos manualmente das monografias, que por sua vez foram coletadas em formato digital, a partir de três fontes, sendo 205 trabalhos pertencentes a monografias defendidas no Departamento de Informática da Universidade Estadual de Maringá (DIN-UEM), 98 no Departamento de Computação da Universidade Estadual de Londrina (DC-UEL) e 82 no Departamento de Informática do Instituto de Física e Matemática da Universidade Federal de Pelotas (Inf-UFPel).

Todos os resumos estão anotados com etiquetas *XML (Extensible Markup Language)*, como mostrado no exemplo da Figura 4.1. Na figura, as quatro etiquetas utilizadas estão destacadas em azul enquanto os quatro atributos estão destacados em verde. A etiqueta <RESUMO>...</RESUMO> tem a função de estabelecer o início e fim do resumo, e o atributo "id" tem o valor (nesse caso "3") atribuído conforme o nome do arquivo em questão. A etiqueta <Titulo> ... </Titulo> estabelece o início e o fim do texto do título do resumo. A etiqueta <P> ... </P> tem a função de estabelecer o início e o fim de cada parágrafo do resumo e o atributo "id" correspondente se refere a posição do parágrafo, iniciando a contagem em zero. De forma similar, a etiqueta <S> ... </S> estabelece o início e o fim de cada sentença do resumo e possui os atributos "id" que recebe o valor do "id" do resumo seguido da posição da sentença e o atributo "AZ" se refere a classificação retórica atribuída à sentença de acordo com um dos seis componentes que compõem a estrutura retórica de um resumo, como apresentado na Figura 3.7.

```
<RESUMO id="3">
<Titulo>Caracterização de Tráfego de Rede</Titulo>
<P id="0">
<S id="3-0" AZ="B">A caracterização de tráfego de rede tem como objetivos
principais: garantia de qualidade de serviço (QoS), planejar e modelar o
tráfego, analisar o desempenho de redes, analisar a perda de pacotes e estudar
o comportamento do usuário.</S>
<S id="3-1" AZ="R">Neste trabalho, serão mostradas algumas técnicas de
caracterização de tráfego como: processos de renovação, processos de Markov,
séries temporais e auto-similaridade.</S>
<S id="3-2" AZ="R">Além destes processos, podem ser utilizados os sketches.</S>
<S id="3-3" AZ="C">Estes possuem a característica de fazer a análise dinâmica e
em tempo real de fluxo de dados, o que é muito importante para detectar
anomalias.</S>
</P>
</RESUMO>
```

Figura 4.1. Exemplo de resumo previamente anotado

A estrutura ideal de um resumo acadêmico deveria conter todos os componentes retóricos sugeridos por Feltrim (2004), porém, a frequência de cada componente identificada no *CorpusTCC* é a seguinte: o componente Proposta está presente em quase todos os resumos analisados com 97,4%, seguido dos componentes Contexto (68,05%), Resultados (55,32%), Lacuna (40,51%), Metodologia (37,66%) e Conclusão (23,11%).

Ao todo, os 385 resumos do *corpus* contabilizam 2.293 sentenças anotadas, sendo que a distribuição dos componentes por sentenças e a média de sentenças por resumo podem ser observadas na Tabela 4.1 adaptada de Souza (2011).

Tabela 4.1. Distribuição dos componentes nas sentenças identificadas. Souza (2011)

Componente	Nº Resumos	Nº Sentenças	Distribuição	Sentenças/Resumo
Contexto	262	808	35,23%	3,08
Lacuna	156	215	9,38%	1,38
Proposta	375	426	18,58%	1,13
Metodologia	145	273	11,90%	1,88
Resultados	213	451	19,67%	2,12
Conclusão	89	120	5,24%	1,35
Total	**385**	**2.293**	**100%**	**5,95**

Na tabela nota-se que embora 97,4% dos resumos apresentem sentenças classificadas como Proposta, esse tipo de sentença não é a de maior quantidade no *corpus*. Isso se deve ao fato das 426 sentenças classificadas como Proposta estarem distribuídas em 375 resumos, o que leva a uma média de 1,13 sentença de Proposta nesses textos. Já as sentenças classificadas como Contexto estão presentes em maior número no *corpus* (808 sentenças), representando 35,23% do total de sentenças, e estão presentes em 262 resumos, resultando a média de 3,08 sentenças.

Além das anotações referentes à estrutura retórica, o *CorpusTCC* também está anotado com informações referentes as quatro dimensões de coerência propostas por Souza (2011): dimensão Título, dimensão Propósito, dimensão Lacuna-Contexto e dimensão Quebra de Linearidade, detalhadas na Subseção 3.4.1. Dos 385 resumos do *corpus*, 101 foram classificados com a dimensão Quebra de Linearidade igual a "sim" pelo modelo (LSA) desenvolvido por Souza (2011). Por considerar que esses resumos têm maior potencial de apresentar quebras de linearidade, eles foram separados e incluídos no *corpus* utilizado por Freitas (2013), que é descrito a seguir.

4.2. Corpus de Freitas (2013)

Além dos resumos do *CorpusTCC*, Freitas (2013) também coletou e anotou 40 resumos experimentais escritos por uma turma do último ano do curso de Ciência da Computação do DIN-UEM. Em sala de aula, foi solicitado aos alunos que escrevessem resumos sobre os seus trabalhos de graduação, que no momento estavam em andamento. Uma vez que esses resumos não passaram pela correção dos respectivos professores orientadores, estariam mais propensos a apresentar problemas de coerência semelhantes aos detectados pela dimensão Quebra de Linearidade. Seguindo a mesma metodologia utilizada por Souza (2011), Freitas anotou os resumos com informações relativas à estrutura retórica e à dimensão Quebra de Linearidade.

Do total de resumos coletados por Freitas, dois foram eliminados do conjunto por terem apenas uma sentença cada não estando assim aptos à avaliação de coerência com a grade de entidades. Desse modo, o *corpus* de Freitas (2013) é composto por 139 resumos.

5. Análise Automática de Quebra de Linearidade

O principal objetivo deste trabalho foi o desenvolvimento de uma extensão para o Módulo de Análise de Coerência (MAC), que é parte da ferramenta de auxílio à escrita científica *SciPo*. Essa extensão deve ser capaz de identificar possíveis quebras de linearidade, um dos aspectos relacionados à coerência semântica e modelados pelas quatro dimensões de coerência propostas por Souza (2011) para resumos científicos. Além da identificação de possíveis quebras, a extensão também deve fornecer sugestões relacionadas a essa dimensão, identificando, ainda que de forma aproximada, a região do resumo que foi considerada como potencialmente problemática.

Para atingir esse objetivo, este trabalho propõe utilizar informações provenientes da estrutura retórica do resumo em conjunto com o modelo grade de entidades implementado por Freitas (2013) para gerar mensagens que indiquem possíveis problemas de coerência local, especificamente, problemas de quebra de linearidade em regiões específicas do resumo. Assim, uma vez que uma possível quebra é identificada, uma sugestão de revisão é apresentada para o usuário, de modo que ele pode acatar a sugestão, revisar o resumo e submetê-lo para uma nova análise, em um processo cíclico de refinamento. Ou, ainda, o usuário pode ignorar as sugestões dadas e continuar ou não a sua interação com a ferramenta. Deixar o usuário livre para escolher entre acatar ou não as sugestões é importante nesse contexto, pois tanto a identificação da estrutura retórica quanto a análise dos aspectos de coerência são tarefas subjetivas e, dessa forma, o usuário pode não concordar com as saídas da ferramenta. Cabe destacar que mesmo especialistas em escrita científica podem discordar quando realizam esses tipos de tarefas. Isso fica evidente nos trabalhos referentes à anotação de *corpus*, em que a concordância entre juízes

não é perfeita. O modelo grade de entidades foi escolhido por apresentar ótimos resultados, principalmente na dimensão quebra de linearidade.

Para que a abordagem proposta pudesse ser avaliada como uma extensão do MAC, um protótipo foi implementado. Esse protótipo utiliza a implementação do modelo grade de entidades de Freitas (2013) e o classificador de estruturas retóricas AZPort (Feltrim et al., 2004), e foi integrado a implementação do MAC feita por Souza (2011). Vale ressaltar que, por se tratar de um protótipo, o MAC está integrado a uma versão *off-line* do sistema SciPo, que tem sido utilizada na avaliação do módulo por usuários. A sua integração à versão *online* do SciPo está prevista como trabalho futuro.

Esta seção apresenta a metodologia empregada neste trabalho, bem como os resultados alcançados. Na Subeção 5.1 é apresentada a etapa de pré-processamento do *corpus*, necessária para o processo de identificação das configurações da grade de entidades. Na Subeção 5.2 é apresentado o processo de construção e avaliação dos modelos de classificação utilizados no protótipo desenvolvido. Por fim, o protótipo e suas avaliações são apresentadas na Seção 5.3, 5.4 e 5.5, respectivamente.

5.1. Pré-processamento

Para o treinamento de modelos classificadores é necessário que se tenha exemplos de resumos pertencentes as classes que esses modelos serão capazes de atribuir após sua criação. Neste trabalho as classes utilizadas foram duas: “com problema de quebra de linearidade” e “sem problema de quebra de linearidade”. Devido ao pequeno número de resumos do *CorpusTCC* anotados como tendo problemas de quebra de linearidade, o que deixa o *corpus* altamente desbalanceado, foi feita a geração de versões artificiais dos textos para o treinamento dos classificadores, permitindo a ampliação e balanceamento do número de exemplos do *corpus*.

Para a implementação do protótipo, foi proposta a criação de dois classificadores, a saber: **(1) classificador de componentes** e **(2) classificador de resumos completos**. A proposta do classificador (1) surgiu a partir do objetivo principal do trabalho, que é, apresentar sugestões sobre coerência apontando a possível localização do problema de quebra de linearidade para o usuário. Para isso, a grade de entidades é aplicada a trechos do resumo, compostos de um ou mais componentes retóricos. Caso o classificador de componentes não encontre problemas de coerência em partes específicas do resumo, então o classificador (2) faz a análise do texto completo para dar um *feedback* sobre a coerência do texto como um todo. Nesse caso, a sugestão apresentada indica a possibilidade de quebras de linearidade, mas não a

sua localização. Os detalhes da construção dos modelos classificadores são apresentados na Seção 5.2.

Cinco experimentos foram feitos para gerar e escolher as melhores versões artificiais dos resumos para serem usadas no treinamento e teste dos classificadores. Cabe ressaltar que, nesse processo, o objetivo foi gerar textos com problemas de coerência, mas não completamente incoerentes, por essa ser uma característica dos textos com problemas de coerência encontrados tanto no *CorpusTCC* quanto em outros *corpora* de textos científicos (Feltrim et al., 2006; Schuster et al., 2005).

Para o treinamento e teste do **(1) classificador de componentes,** versões sintéticas dos textos do *CorpusTCC* foram geradas de duas maneiras diferentes. São elas:

a. Inversão das sentenças de fronteira com no mínimo quatro sentenças: nessa tarefa, pares de componentes retóricos adjacentes são extraídos de cada resumo original, sendo que, cada componente deve ser composto de no mínimo duas sentenças. Após a extração do par, um novo arquivo de texto é gerado contendo o par de componentes com as posições das sentenças de fronteira dos componentes invertidas. Mais de uma versão sintética do resumo pode ser gerada se o texto apresentar dois ou mais pares de componentes adjacentes com pelo menos duas sentenças cada um. Esse processo é exemplificado na Figura 5.1.

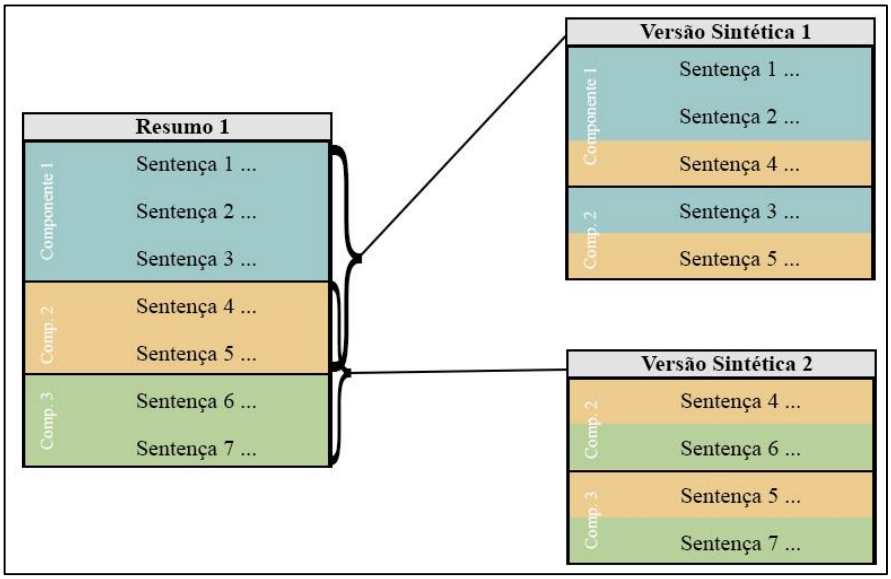

Figura 5.1. Exemplo de resumos gerados sinteticamente pela inversão de sentenças na fronteira dos componentes retóricos

b. Inversão das sentenças de fronteira com no mínimo três sentenças: essa tarefa é similar à tarefa (**a**). Pares de componentes retóricos adjacentes também são extraídos de cada resumo original. A diferença é que, nesse caso, a restrição de possuir no

mínimo duas sentenças se aplica apenas a um dos componentes do par. Assim, são geradas versões sintéticas pela inversão de sentenças da fronteira do par de componentes, sendo que um deles é composto por uma única sentença.

A partir dos 385 resumos do *Corpus*TCC foram gerados na tarefa (**a**) 268 exemplos, sendo 134 pares originais e 134 pares sintéticos, com uma média de 5,67 sentenças por texto. Como a tarefa (**b**) tem uma regra menos restritiva, nessa tarefa foram gerados 1.160 exemplos, sendo 580 pares na ordem original e 580 pares sintéticos, com a média de 4,30 sentenças por texto.

Para ambas as tarefas (**a** e **b**) os textos compostos pelos pares de componentes com as sentenças na ordem original são considerados "sem problema" de quebra de linearidade, enquanto os textos compostos pelos pares de componentes com a ordem das sentenças de fronteira invertida são considerados "com problema" de quebra de linearidade.

Para o treinamento e teste do **(2) classificador de resumos completos**, foram geradas, de três formas diferentes, versões sintéticas a partir do *Corpus*TCC:

a. Inversão de ordem de dois componentes: nessa tarefa, dois componentes retóricos completos (com todas as sentenças) são selecionados aleatoriamente no resumo e entre eles é feita a troca de posição. Esse processo é exemplificado na Figura 5.2.

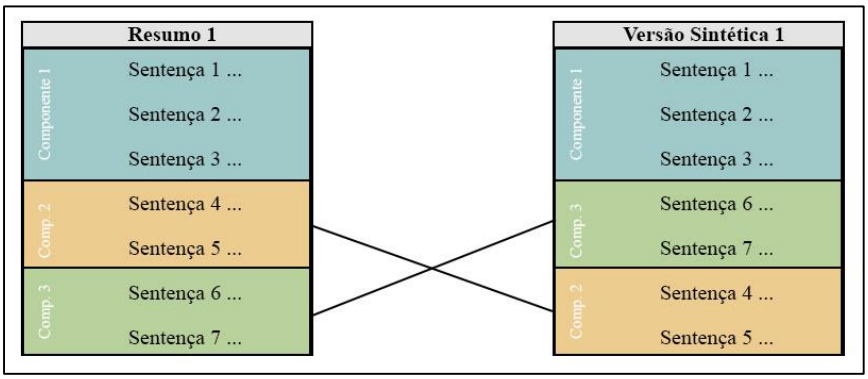

Figura 5.2. Exemplo de um resumo gerado sinteticamente pela inversão de componentes

b. Inversão de ordem de três componentes: nessa tarefa, três componentes completos são selecionados aleatoriamente e são reposicionados no resumo em outras posições escolhidas também de forma aleatória.

c. Inversão da ordem dos componentes: nessa tarefa, versões sintéticas dos resumos são geradas a partir da inversão total da ordem dos componentes retóricos completos.

A partir dos resumos do *Corpus*TCC foram gerados em cada uma das tarefas (**a**, **b** e **c**) 770 exemplos, sendo 385 resumos originais e 385 resumos sintéticos, com a média de 5,86 sentenças por texto.

Para as três tarefas (**a**, **b** e **c**) os 385 resumos originais do *Corpus*TCC foram considerados como "sem problema" de quebra de linearidade e as versões sintéticas foram consideradas "com problema" de quebra de linearidade.

5.2. Construção e Análise dos Modelos de Classificação

Para a construção dos dois modelos classificadores se fez necessária, por meio experimentos, a escolha do melhor subconjunto de textos gerados sinteticamente para cada modelo classificador.

Para cada um dos textos utilizados nos experimentos, foram construídas grades de entidades e extraídos os vetores de características conforme as quatro configurações formadas pela combinação das dimensões linguísticas disponíveis no modelo de Freitas (2013) (função sintática(+/-) e saliência(+/-)).

Todos os classificadores foram treinados e testados no ambiente WEKA (Witten e Frank, 2005) usando validação cruzada estratificada com 10 partições (*10-fold stratified cross-validation*). O algoritmo utilizado para a indução dos classificadores foi o J48, uma implementação em Java e de código aberto do indutor de árvores de decisão C4.5 (Quinlan, 1993). Embora vários trabalhos utilizam a grade de entidades para ranquear versões sintéticas de textos, o algoritmo J48 foi escolhido pois nesta pesquisa o modelo foi utilizado para classificar os textos como "com problemas" e "sem problemas", além disso, os experimentos de Freitas (2013) apresentaram bons resultados com o mesmo algoritmo. Como métrica de avaliação foram adotadas medidas comumente utilizadas na avaliação de classificadores, a saber: Taxa de Acerto (Acurácia) é a porcentagem dos documentos que foram corretamente classificados, Precisão (*Precison*) é a porcentagem dos documentos que são corretamente rotulados como pertencentes à classe, Cobertura (*Recall*) é a porcentagem de todos os documentos pertencentes à classe em questão que conseguiram ser recuperados, Medida-F (*F-measure*) é uma medida harmônica da Precisão (*Precision*) e Cobertura (*Recall*) e *Macro-F* que é uma média das Medidas-F (*F-Measures*) de todas as classes.

Os experimentos **1.a** e **1.b** foram feitos para a avaliação e seleção do melhor **(1) classificador de componentes.** Esses experimentos também buscaram avaliar se o modelo grade de entidades funcionaria para textos menores. Os experimentos **2.a**, **2.b** e **2.c** foram feitos

para a avaliação e seleção do melhor **(2) classificador de resumos completos.** Todos os experimentos são detalhados a seguir.

Experimento 1.a

Nesse experimento o **(1) classificador de componentes** foi treinado com os 268 pares de componentes extraídos do *Corpus*TCC, sendo 134 textos na ordem original das sentenças dos componentes, considerados como "sem problema" em relação à quebra de linearidade, e os outros 134 gerados a partir da **(a) inversão das sentenças de fronteira com no mínimo quatro sentenças**, considerados como "com problema".

Como mencionado no início desta seção, para a escolha dos classificadores, as quatro possíveis configurações das dimensões linguísticas foram utilizadas na construção da grade de entidades, a saber: Sintático+ Saliência+, Sintático+ Saliência-, Sintático- Saliência+ e Sintático- Saliência-.

Nesse experimento, o melhor resultado (*Macro-F* = 0,48) foi obtido com a configuração Sintático- Saliência- e Sintático- Saliência+, igualando à *baseline* feita com LSA. A taxa de acerto com essas configurações foi de 48,50% e a Tabela 5.1 apresenta os resultados das medidas *Precision, Recall, F-measure* e *Macro-F* para cada classe identificada.

Tabela 5.1. Resultados da avaliação do classificador (1.a)

Modelo	**Com Problema**			**Sem Problema**			
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*	*Macro-F*
LSA (*Baseline*)	0,488	0,582	0,531	0,481	0,388	0,43	**0,48**
Sintático- Saliência-	0,481	0,388	0,43	0,488	0,582	**0,531**	**0,48**
Sintático+ Saliência-	0,5	0,836	0,626	0,5	0,164	0,247	0,436
Sintático- Saliência+	0,481	0,388	0,43	0,488	0,582	**0,531**	**0,48**
Sintático+ Saliência+	0,509	0,851	**0,637**	0,545	0,179	0,27	0,453

Tomando por base os valores das F-*measures* da Tabela 5.1 podemos observar que os resultados dos classificadores foram bastante desbalanceados, ou seja, os resultados foram melhores somente para uma das classes, reduzindo a média (*Macro-F*) em todas as configurações. Esse comportamento pode ser provavelmente atribuído ao tamanho reduzido do *corpus* de treinamento utilizado no experimento. De fato, a sensibilidade do modelo grade de entidades ao tamanho do *corpus* também foi observada por Freitas (2013).

Experimento 1.b

Nesse experimento o **(1) classificador de componentes** foi treinado com os 1.160 pares de componentes extraídos do *Corpus*TCC, sendo que os 580 textos extraídos na ordem original

das sentenças dos componentes foram considerados como "sem problema" em relação à quebra de linearidade e os outros 580 foram gerados a partir da **(b) inversão das sentenças de fronteira com no mínimo três sentenças** considerados como "com problema".

Dada a quantidade de pares usados no treinamento era esperado que essa configuração obtivesse melhor resultado, uma vez que ela incorpora mais conhecimento sobre as entidades. A taxa de acerto mais alta (95,17%) foi obtida com a configuração Sintático+ Saliência+. As taxas de acerto dos classificadores gerados usando as quatro configurações possíveis para a grade de entidades são mostradas na Tabela 5.2.

Tabela 5.2. Resultados da avaliação do classificador (1.b)

Modelo	**Com Problema**			**Sem Problema**			
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*	*Macro-F*
LSA (*Baseline*)	0,5	1	0,667	0	0	0	0,333
Sintático- Saliência-	0,465	0,150	0,227	0,493	0,828	0,618	0,423
Sintático+ Saliência-	0,445	0,160	0,236	0,488	0,800	0,606	0,421
Sintático- Saliência+	0,933	0,910	0,921	0,912	0,934	0,923	0,922
Sintático+ Saliência+	0,955	0,948	**0,952**	0,949	0,955	**0,952**	**0,952**

Com base nos resultados mostrados na Tabela 5.2 e analisando as diferentes configurações do modelo grade de entidades, fica evidente a contribuição do aspecto saliência no desempenho do modelo, enquanto a informação sintática contribuiu apenas moderadamente. O melhor resultado em todas as medidas foi obtido com a configuração Sintático+ Saliência+ e o segundo melhor foi com a configuração Sintático- Saliência+. Outro fato a ser destacado é que nesse experimento os resultados das medidas para ambas as classes foram muito próximos ou até igual, como é o caso da *F-Measure* = 0,952 na configuração Sintático+ Saliência+, mostrando que com um número maior de exemplos de treinamento o modelo não apresenta nenhuma predisposição para atribuir uma classe específica, como ocorreu no experimento **1.a**.

Experimento 2.a

Nesse experimento o **(2) classificador de resumos completos** foi treinado com 770 textos, sendo os 385 resumos originais do *Corpus*TCC considerados como "sem problema" de quebra de linearidade e os outros 385 gerados a partir da **(a) inversão de ordem de dois componentes** considerados como "com problema".

Assim como no experimento **1.b**, o melhor desempenho foi obtido com a grade de entidades na configuração Sintático+ Saliência+, com taxa de acerto de 76,75%. Os resultados em termos de *Precision*, *Recall* e F-*measure* são mostrados na Tabela 5.3.

Como pode ser observado, os resultados desse experimento ficaram abaixo dos obtidos para o **classificador de componentes** no experimento **1.b**. O modelo classificou em média 22,59% de falsos positivos e que o seu comportamento foi similar para ambas as classes. Cabe destacar, no entanto, que o conjunto de exemplos utilizado nesse experimento é menor do aquele do experimento **1.b**. Além disso, é possível argumentar que o impacto da inversão de um par de sentenças na coerência de um texto composto de poucas sentenças é maior do que aquele causado pela inversão de dois componentes retóricos em um resumo completo. Dessa forma, as quebras de linearidade observada nos resumos sintéticos desse experimento seriam mais sutis do que aquelas observadas nos textos do experimento **1.b** e, consequentemente, mais difíceis de classificar.

Tabela 5.3. Resultados da avaliação do classificador (2.a)

Modelo	**Com Problema**			**Sem Problema**			
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*	*Macro-F*
LSA (*Baseline*)	0,496	0,594	0,541	0,494	0,396	0,44	0,49
Sintático- Saliência-	0,411	0,216	0,283	0,468	0,691	0,558	0,421
Sintático+ Saliência-	0,624	0,608	0,616	0,618	0,634	0,626	0,621
Sintático- Saliência+	0,762	0,740	0,751	0,747	0,769	0,758	0,754
Sintático+ Saliência+	0,771	0,761	0,766	0,749	0,774	0,769	0,767

Experimento 2.b

Para esse experimento o **(2) classificador de resumos completos** foi treinado com 770 textos, sendo os 385 resumos originais do *Corpus*TCC considerados como "sem problema" e os outros 385 gerados a partir de **(b) inversões de ordem três componentes** considerados como "com problema" de quebra de linearidade.

Assim como nos dois experimentos anteriores, o melhor resultado se obteve utilizando os dois recursos linguísticos disponíveis para a construção da grade de entidades (Sintático+ Saliência+), sendo que a taxa de acerto observada foi de 80,13%. Na Tabela 5.4 podemos observar que o desempenho do modelo foi ligeiramente melhor para a classe "sem problema" em três das quatro configurações.

Tabela 5.4. Resultados da avaliação do classificador (2.b)

Modelo	**Com Problema**			**Sem Problema**			
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*	*Macro-F*
LSA (*Baseline*)	0,496	0,594	0,541	0,494	0,396	0,44	0,49
Sintático- Saliência-	0,620	0,221	0,326	0,526	0,865	0,654	0,490
Sintático+ Saliência-	0,605	0,777	0,680	0,688	0,494	0,575	0,628
Sintático- Saliência+	0,837	0,642	0,726	0,709	0,875	0,784	0,755
Sintático+ Saliência+	0,882	0,696	0,778	0,749	0,906	0,820	0,799

Experimento 2.c

Nesse experimento o **(2) classificador de resumos completos** foi treinado com 770 textos, sendo os 385 resumos originais do *Corpus*TCC considerados como "sem problema" e os outros 385, gerados a partir da **(c) inversão total da ordem dos componentes**, considerados como "com problema".

Novamente, o melhor classificador, com 85,05% de taxa de acerto, foi obtido com a grade de entidades na configuração Sintático+ Saliência+. Essa taxa de acerto é menor do que a obtida com o **classificador de componentes** do experimento **1.b**, provavelmente devido à diferença na quantidade de exemplos de treinamento (770 resumos completos *versus* 1.160 pares de componentes). Os resultados em termos de *Precision*, *Recall* e *F-measure* com as quatro configurações possíveis da grade de entidades são mostrados na Tabela 5.5.

Tabela 5.5. Resultados da avaliação do classificador (2.c)

Modelo	**Com Problema**			**Sem Problema**			
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*	*Macro-F*
LSA (*Baseline*)	0,496	0,594	0,541	0,494	0,396	0,44	0,49
Sintático- Saliência-	0,494	0,196	0,281	0,498	0,799	0,614	0,448
Sintático+ Saliência-	0,494	0,196	0,281	0,498	0,799	0,614	0,448
Sintático- Saliência+	0,701	0,282	0,402	0,551	0,880	0,677	0,540
Sintático+ Saliência+	0,839	0,868	**0,853**	0,863	0,833	**0,848**	**0,851**

Assim como nos experimentos para o classificador de componentes, os modelos que incorporam informações de saliência obtiveram as melhores taxas de acerto para resumos completos. Vale notar que a contribuição da informação sintática foi mais expressiva nesse caso, em que os textos classificados são maiores. Especialmente para a classe "Com Problema", o uso da informação sintática aumentou consideravelmente a medida *Recall* e, consequentemente, a *F-measure* dessa classe.

Os resultados dos experimentos **2.a, 2.b** e **2.c** mostram que o desempenho do **classificador de resumos completos** foi melhorando à medida que o número de inversões realizadas nos textos do conjunto de treinamento aumentou. Em princípio, um número maior de inversões caracterizaria um texto mais "problemático", no qual as quebras de linearidade seriam identificadas mais facilmente. Isso reforça a observação feita no experimento **1.a** sobre o impacto das inversões na coerência do texto e mostra que o modelo grade de entidades consegue identificar quebras de linearidade, porém existe sensibilidade em relação ao impacto da quebra na coerência do texto como um todo.

Seleção dos Classificadores

Os resultados obtidos na avaliação dos classificadores mostraram que o modelo de grade de entidades é capaz de detectar problemas de quebra de linearidade mesmo em trechos pequenos (experimentos 1.a e 1.b), compostos de poucas sentenças. Mostraram também que a melhor configuração do modelo é a Sintático+ Saliência+ para quase todos os subconjuntos de resumos utilizados, especialmente para os resumos completos e quando se dispõe de um número maior de exemplos de treinamento. Assim, os classificadores que obtiveram os melhores resultados foram o **classificador de componentes 1.b** e o **classificador de resumos completos 2.c**. Atualmente, eles estão sendo utilizados na dimensão Quebra de Linearidade do MAC.

Na próxima seção é discutida a implementação desses classificadores combinados com a estrutura retórica do resumo no MAC, possibilitando a geração de sugestões para o usuário do sistema SciPo.

5.3. Protótipo Desenvolvido

Na Seção 1 foi citado que uma quarta dimensão para o MAC do *SciPo*, chamada Quebra de Linearidade, foi proposta por Souza (2011), mas não chegou a ser automatizada. Experimentos realizados por Freitas (2013) mostraram que o modelo grade de entidades poderia ser utilizado na automatização dessa dimensão, porém, a questão relativa ao uso do modelo na geração de sugestões aos usuários do sistema *SciPo* permanecia aberta.

A questão que se coloca quanto à integração do modelo grade de entidades no MAC advém do fato do modelo ter sido usado até o desenvolvimento deste trabalho para analisar textos completos (Barzilay e Lapata, 2008; Burstein et al., 2010; Yokono e Okumura, 2010; Elsner e Charniak, 2011; Lin et al., 2011; Freitas e Feltrim, 2013; Dias et al., 2014). A análise do texto como um todo permite identificar textos com quebras de linearidade, mas não permite a identificar a localização das quebras.

Informar que o texto possui quebras de linearidade sem dar uma indicação da região em que as quebras ocorrem é de pouca utilidade no contexto de uma ferramenta de auxílio à escrita como o *SciPo*. Assim, é preciso que as sugestões geradas pela ferramenta sejam mais específicas e informem, ainda que de forma aproximada, em qual trecho do texto a quebra foi encontrada.

A solução proposta por este trabalho é usar a grade de entidades na análise de trechos de texto menores, constituídos por um ou mais componentes retóricos. Essa análise por trechos

permite a geração de mensagens que indiquem possíveis problemas de quebra de linearidade em um componente ou grupo de componentes retóricos específicos. Cabe ressaltar que os experimentos realizados com textos menores, compostos por pares de componentes retóricos, mostraram que essa solução é viável.

A partir da identificação dos componentes retóricos, a análise da dimensão Quebra de Linearidade pode ser iniciada. Em uma primeira etapa, grades de entidades individuais são construídas para cada um dos componentes retóricos que possuem pelo menos duas sentenças, conforme exemplificado na Figura 5.3.

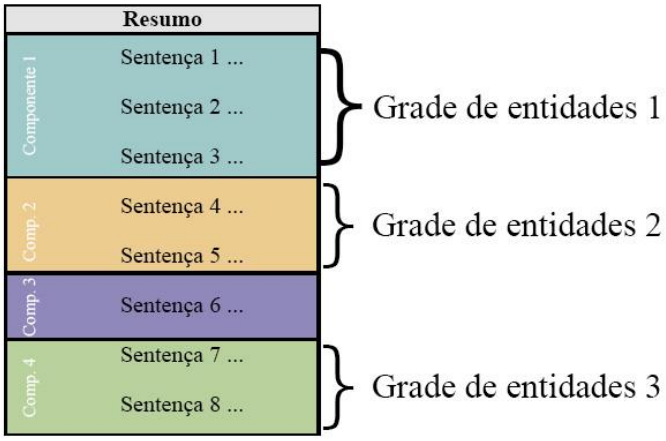

Figura 5.3. Exemplo de construção da grade de entidades para cada componente retórico

A partir de cada grade é extraído um vetor de características que é então testado pelo **classificador de componentes** apresentado na subseção anterior. Assim, cada trecho de texto representado pelos vetores é classificado como “com problema”/“sem problema” de quebra de linearidade. Quando um componente é classificado como “com problema”, uma sugestão é gerada ao usuário indicando que aquele trecho, composto pelo componente retórico específico, possui uma possível quebra de linearidade. O usuário, por sua vez, pode acatar a sugestão, retornar ao texto para modificá-lo e enviá-lo para uma nova análise, ou pode escolher ignorar a sugestão dada, o que faz com que o processo de análise da dimensão continue.

A Figura 5.4 mostra um exemplo de sugestão gerada pelo protótipo quando possíveis quebras de linearidade foram encontradas no componente Conclusão.

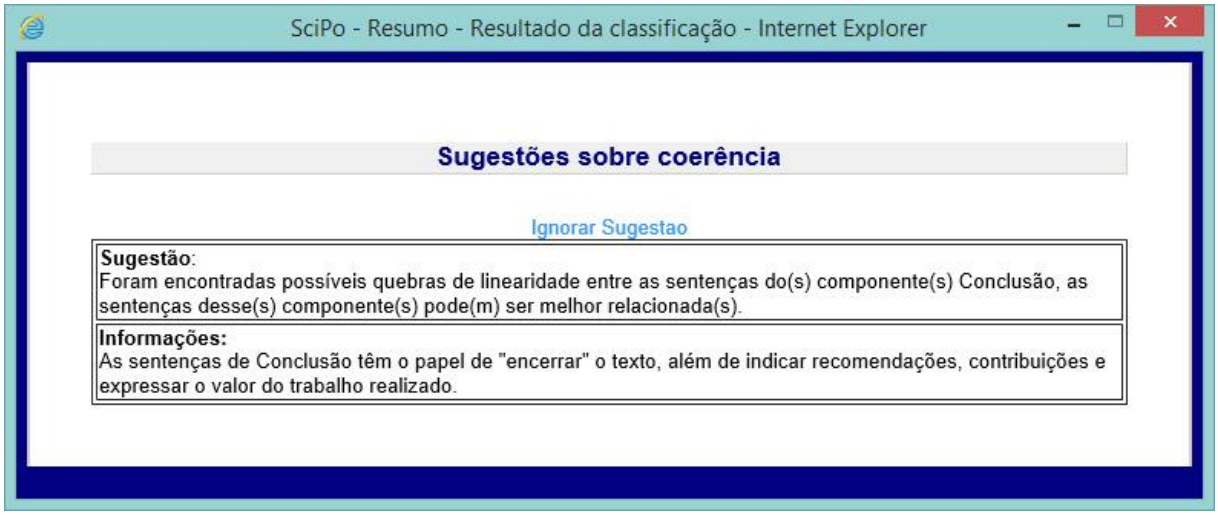

Figura 5.4. Captura de tela do protótipo com uma sugestão para quebras em um componente retórico

Caso a primeira etapa da análise não apresente sugestões ou caso o usuário ignore as sugestões, então uma segunda etapa é iniciada, na qual novas grades de entidades são construídas para todos os pares de componentes adjacentes, conforme exemplificado na Figura 5.5.

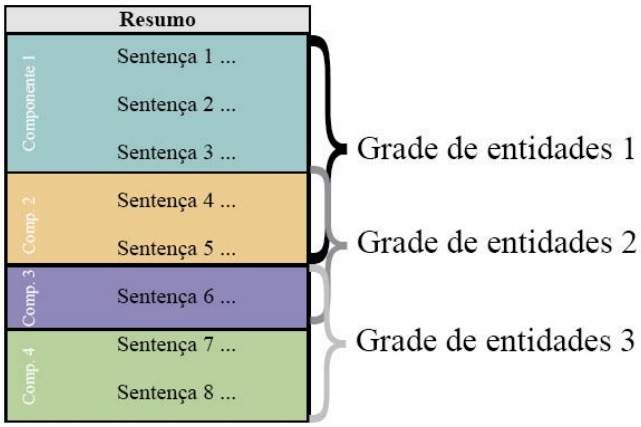

Figura 5.5. Exemplo de construção da grade para os pares de componentes retóricos

Na Figura 5.6 é apresentada uma nova sugestão, gerada pelo protótipo quando possíveis quebras de linearidade foram encontradas nos componentes Contexto e Lacuna.

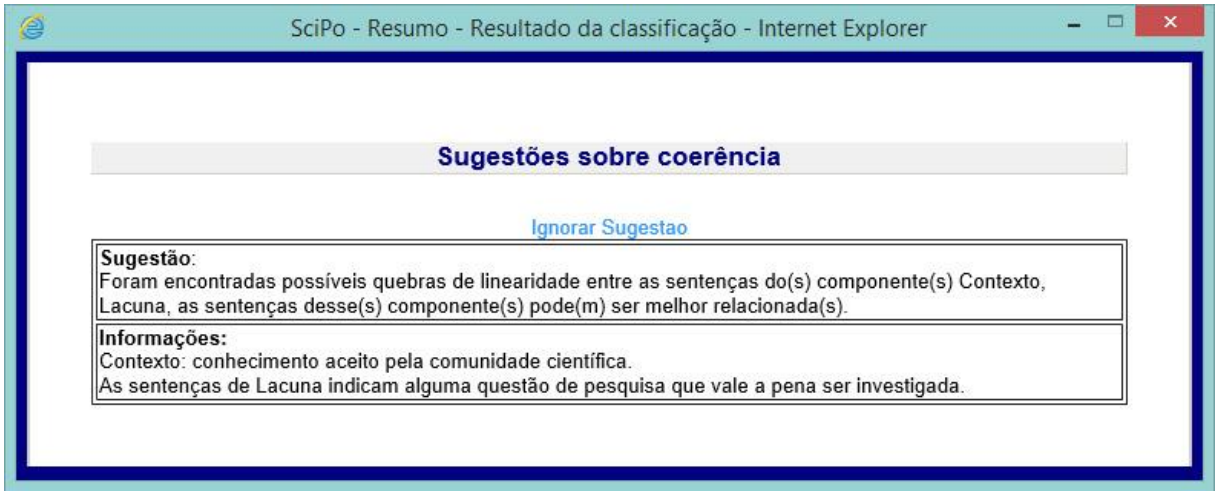

Figura 5.6. Captura de tela do protótipo com uma sugestão para quebras em pares de componente retóricos

O processo de classificação do par de componentes se repete como na primeira etapa e, caso a análise continue, uma nova etapa é iniciada. A cada nova etapa, grupos maiores de componentes retóricos, gerados por meio da adição de um componente adjacente, são usados para gerar as grades de entidades e realizar a classificação.

A análise da dimensão continua enquanto o **classificador de componentes** não detectar quebras de linearidade no texto e termina quando houver um único grupo de componentes retóricos que corresponde ao resumo completo. Para essa última etapa é construída uma única grade de entidades para o resumo e o vetor de características extraído é testado pelo **classificador de resumos completos**, apresentado na subseção anterior.

5.4. Primeira Avaliação do Protótipo

Um dos desafios de se usar um modelo como grade de entidades como parte do MAC está relacionado à escassez de textos científicos originais, em especial, resumos em português, anotados com problemas de quebra de linearidade. Coletar e anotar tal *corpus* não são tarefas triviais.

A dificuldade em relação à coleta vem do fato de resumos publicados serem versões já revisadas, muitas vezes por pesquisadores experientes, sendo pouco provável que contenham problemas de quebra de linearidade. Esse tipo de problema de coerência é mais comum em textos escritos por alunos, especialmente aqueles com pouca experiência em escrita científica. No entanto, o acesso a esse tipo de texto envolve contato direto com os autores e isso dificulta a coleta em quantidade suficiente para o treinamento de um modelo como a grade de entidades.

Já em relação à anotação do *corpus*, a dificuldade vem da subjetividade associada à avaliação de aspectos relacionados à coerência textual, como é o caso das quebras de linearidade. Segundo Koch e Travaglia (2003), a coerência diz respeito à possibilidade de se estabelecer um sentido entre diferentes sentenças do texto. Logo, é um princípio de interpretabilidade ligado à situação de comunicação e à capacidade do leitor em estabelecer o sentido do texto. Por isso, está vinculada ao texto, mas não depende somente dele. Dessa forma, um mesmo trecho de texto pode caracterizar uma quebra de linearidade para um anotador, e ser considerado coerente por outro.

A essas dificuldades, soma-se o custo em termos de hora/pessoa que a anotação de um *corpus* dessa natureza e tamanho demanda. Por essas razões foram criados os *corpora* de exemplos descritos nas subseções anteriores, o que permitiu treinar os classificadores e obter boas taxas de acerto com validação cruzada. Mas qual seria o desempenho dos classificadores na avaliação de textos reais? E, além disso, os trechos com quebra de linearidade seriam corretamente identificados?

Para responder a essas perguntas, o protótipo desenvolvido foi avaliado com um conjunto de 28 resumos originais, sendo 14 resumos "com problema" e 14 resumos "sem problema". Os resumos "com problema" foram selecionados manualmente por dois anotadores entre os resumos do *corpus* utilizado por Freitas (2013). Nesses resumos, os anotadores também identificaram os pares de sentenças que caracterizavam as quebras de linearidade. Nos 14 resumos "com problema" foram identificados 18 pares de sentenças com quebra de linearidade. Os 14 resumos "sem problema" foram selecionados aleatoriamente a partir do restante do *corpus* de Freitas (2013).

Após a seleção e a anotação dos resumos, foram conduzidos dois experimentos. O primeiro deles, experimento (1), foi realizado com os 28 resumos originais e buscou avaliar a acurácia do **classificador de resumos completos** com textos reais. O experimento (2) buscou verificar a acurácia **do classificador de componentes** em relação à identificação dos pares de sentenças anotados com quebra de linearidade.

A taxa de acerto observada no experimento (1) foi de 67,86%. A matriz de confusão correspondente é mostrada na Tabela 5.6. Conforme mostrado na tabela, o protótipo identificou 15 resumos, dez deles corretamente, como tendo quebra de linearidade, logo a precisão para essa classe foi de 66,67%. Dos 14 resumos anotados como "com problema", dez foram corretamente identificados, de modo que a cobertura para essa classe foi de 71,43%.

Tabela 5.6. Matriz de confusão para textos completos

Com Quebra	Sem Quebra	Classificados como ▼
10	5	Com quebra
4	9	Sem quebra
14	14	

Para o experimento (2), 73 pares de sentenças foram extraídos a partir dos 14 resumos "com problema", sendo 18 pares anotados manualmente como "com problema" e 55 anotados como "sem problema". Os pares foram testados com o **classificador de componentes** retóricos, por esse classificador ter sido treinado com textos menores. A matriz de confusão resultante desse segundo experimento é mostrada na Tabela 5.7.

Tabela 5.7. Matriz de confusão dos pares de sentenças

Com Quebra	Sem Quebra	Classificados como ▼
9	6	Com quebra
9	49	Sem quebra
18	55	

Como pode ser observado na Tabela 5.7, 15 pares de sentenças foram classificados como "com problema" de quebra de linearidade, 9 deles corretamente, correspondendo a uma precisão de 60% para essa classe. Dos 18 pares de sentenças manualmente anotados com essa classe, 9 foram corretamente identificados, correspondendo a uma cobertura de 50%. Em comparação ao experimento (1), tanto a precisão quanto a cobertura para a classe "com problema" foram mais baixas. De fato, identificar o par de sentenças que caracteriza uma quebra é uma tarefa mais difícil mesmo para os anotadores humanos, devido à alta subjetividade da tarefa.

Embora o número de resumos utilizados nessa avaliação tenha sido pequeno, foi possível perceber que é mais fácil haver concordância entre os anotadores em relação à existência ou não de quebra de linearidade no resumo, do que em relação à identificação dos pares de sentenças que caracterizam as quebras. Essa dificuldade se reproduziu nos resultados dos experimentos realizados com os classificadores.

5.5. Segunda Avaliação do Protótipo

A segunda avaliação realizada com o protótipo desenvolvido buscou verificar a sua utilidade como parte do MAC do sistema SciPo. Para isso, foi realizado um experimento com usuários, em que o protótipo foi avaliado simulando o seu contexto real de uso.

Para esse experimento foram selecionados nove usuários voluntários, sendo seis deles alunos do curso de Mestrado em Ciência da Computação da Universidade Estadual de Maringá e os outros três alunos do curso de Ciência da Computação da Universidade Tecnológica Federal do Paraná, *Campus* de Campo Mourão.

Primeiramente, foi solicitado aos participantes que utilizassem seus resumos científicos mais recentes, escritos durante a realização do mestrado ou graduação (resumo de artigo, qualificação, dissertação ou TCC). Aos usuários também foram apresentados: os objetivos e a metodologia do protótipo, uma breve explicação a respeito dos componentes retóricos que compõem a estrutura de um resumo e uma breve descrição do processo de análise de coerência realizada protótipo, esclarecendo, por exemplo, como as sugestões lhes seriam apresentadas.

Individualmente, os participantes submeteram seus resumos para a identificação automática da estrutura retórica pelo SciPo. Após fazerem as correções na estrutura retórica que julgavam necessárias, os usuários submeteram seus resumos à análise automática de quebra de linearidade. Paralelamente a essa tarefa, todos os participantes preencheram um questionário relatando suas impressões sobre o protótipo e as sugestões apresentadas por ele. O questionário é apresentado no Apêndice A. Cabe ressaltar que o autor do protótipo estava presente durante todo experimento, mas que só foram feitas intervenções quando diretamente solicitadas pelo participante.

Para quatro usuários, o protótipo apresentou uma única sugestão em relação à dimensão quebra de linearidade. Para três desses usuários, a sugestão foi que uma possível quebra tinha sido encontrada no componente Contexto. Para o quarto estudante a sugestão se referiu ao componente Lacuna. Para outros dois usuários, o protótipo apresentou duas sugestões. Para um deles as sugestões relativas à quebra de linearidade referiram-se ao componente Propósito e ao par de componentes Lacuna - Propósito. Para o segundo usuário as sugestões foram referentes ao componente Contexto e ao par Contexto - Lacuna. Para outros dois estudantes, o protótipo apresentou três sugestões. Para um deles as sugestões relativas à quebra de linearidade se referiram aos componentes Contexto, Propósito e Metodologia. Para o outro estudante as sugestões foram referentes aos componentes Lacuna, Metodologia e Resultados. Um dos participantes não recebeu sugestões do protótipo.

Durante o experimento, o protótipo não apresentou nenhuma sugestão em relação ao componente Conclusão. Isso não se deve só ao fato dos resumos analisados não apresentarem problemas em relação ao componente Conclusão, mas também por esse componente, na maioria das vezes, ser composto por apenas uma sentença. Componentes com uma única sentença não são analisados individualmente quanto à quebra de linearidade. Em outro caso, o

estudante classificou a sentença de Conclusão junto com as sentenças de Resultado, ainda que tenha sido previamente explicada aos estudantes a importância de cada componente na estrutura de um resumo científico.

Dois estudantes não acataram as sugestões emitidas pelo protótipo. Quatro acataram integralmente (para todos os componentes mostrados na sugestão, houve pelo menos uma sentença corrigida pelo usuário) e dois acataram parcialmente (nem todos os componentes destacados na sugestão foram corrigidos pelo usuário), de maneira que mesmo após a reescrita e adequação do resumo, o protótipo ainda apresentava alguma sugestão.

Em relação à relevância das sugestões apresentadas, os seis estudantes que acataram as sugestões dadas pelo protótipo as consideraram relevantes.

Ao comparar o resumo reescrito após a adequação de acordo com as sugestões com a versão inicial do mesmo resumo, quatro estudantes consideraram que o nível de coerência do resumo final aumentou razoavelmente, enquanto um estudante considerou um bom aumento no nível de coerência e outro estudante considerou que as modificações no resumo não alteraram o seu nível de coerência.

Na Figura 5.7 são apresentados gráficos referentes às opiniões dos nove participantes do experimento em relação à relevância das sugestões emitidas pelo protótipo e ao nível de coerência dos resumos após as adequações sugeridas.

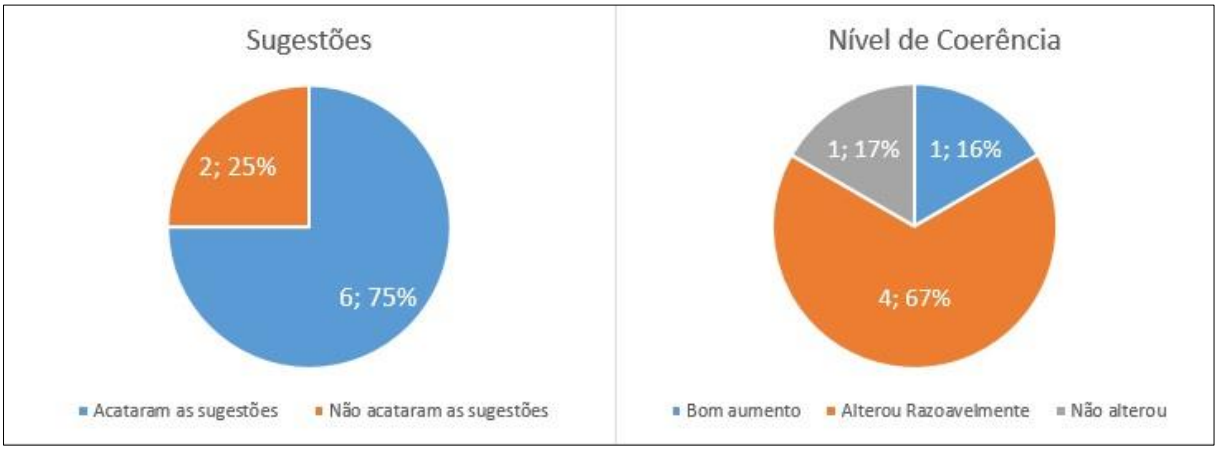

Figura 5.7. Gráficos de opiniões dos participantes do experimento

Conforme já comentado, o gráfico de Sugestões mostra que seis (75%) dos oito participantes que receberam sugestões acataram, mesmo que de forma parcial, as sugestões emitidas pelo protótipo. Já o gráfico de Nível de Coerência mostra que cinco (83%) dos seis participantes que acataram alguma sugestão avaliaram que o resumo final teve um nível maior de coerência comparado ao resumo inicial. Esses resultados mostram que, ao menos na visão

do autor do texto, as sugestões apresentadas pelo protótipo auxiliam na tarefa de escrever resumos científicos com um nível maior de coerência.

6. Conclusões

Este trabalho teve por objetivo estender o módulo de análise de coerência da ferramenta SciPo, chamado MAC (Souza e Feltrim, 2012), por meio da automatização da dimensão Quebra de Linearidade, originalmente proposta por Souza (2011), proporcionando aos usuários do SciPo auxílio na escrita científica no que concerne a coerência local.

A solução proposta neste trabalho utiliza a grade de entidades como modelo para a avaliação de coerência local em resumos e o seu principal diferencial está na forma como o modelo é aplicado no contexto do MAC. O uso da grade de entidades para a análise de trechos menores de textos, juntamente com as informações provenientes da estrutura retórica do resumo, permite a geração de sugestões mais específicas, tornando-as mais úteis para os usuários da ferramenta SciPo.

Os resultados experimentais para a escolha dos modelos classificadores mostraram que a proposta de analisar trechos menores de texto usando a grade de entidades como modelo de coerência é viável, embora o desempenho dependa do tamanho do *corpus* de treinamento. De fato, essa é uma característica do modelo de grade de entidades, assim como de outros modelos que utilizam aprendizagem de máquina, e já havia sido destacada por Freitas (2013).

Para ampliar o tamanho do *corpus* de treinamento os resumos com quebra de linearidade foram gerados artificialmente. Embora esse processo tenha buscado simular quebras sutis de linearidade, os experimentos com textos originais mostraram que as quebras existentes nesses textos, em geral, são ainda mais sutis, causando uma queda no desempenho do MAC em relação aos resultados obtidos para os classificadores com validação cruzada.

Visto que a primeira avaliação do protótipo, feita com textos originais, foi realizada com um número pequeno de resumos (28), uma atividade prevista para o futuro é aumentar o *corpus* de resumos com problemas de quebra de linearidade. Conforme comentado na Subseção 5.4,

essa não é uma tarefa trivial, mas permitirá a realização de uma avaliação mais abrangente do MAC, além do uso do *corpus* como parte do treinamento dos classificadores.

Mesmo que as quebras de linearidade em resumos originais sejam sutis, os resultados da segunda avaliação do protótipo, feita com usuários reais, mostraram que as sugestões apresentadas são relevantes, principalmente para iniciantes da escrita científica, guiando os alunos na construção de um texto com um nível maior de coerência.

Conforme descrito na seção 5, atualmente são utilizados dois modelos classificadores no protótipo da dimensão Quebra de Linearidade: um para resumos completos e outro para componentes retóricos, sendo o último treinado com pares de componentes. Como o tamanho do texto classificado pode influenciar o desempenho dos modelos, um próximo passo é o treinamento e teste de modelos classificadores para grupos maiores de componentes retóricos (trios, quartetos, etc.).

O protótipo que foi desenvolvido está disponível apenas como uma versão *offline* e teve como principal objetivo a validação da solução proposta neste trabalho. Assim, outra atividade prevista para o futuro é a integração do MAC à versão *online* da ferramenta SciPo, que está atualmente disponível a partir do *site* do NILC[4] - Núcleo Interdisciplinar de Linguística Computacional.

Este trabalho abordou a análise de coerência local em resumos científicos, pois esse tipo de texto tem uma estrutura retórica bem definida, possibilitando assim a avaliação de quebra de linearidade nos componentes retóricos do texto. Um desdobramento natural deste trabalho é a extensão, não somente da dimensão Quebra de Linearidade, mas de todo o MAC para a análise de coerência em introduções, uma vez que esse tipo de texto também tem a estrutura bem definida (Feltrim, 2004).

Diante do que foi apresentado, entende-se que este trabalho contribui para o desenvolvimento do ambiente de auxílio à escrita SciPo e, consequentemente, para a ampliação dos estudos na área de ferramentas de auxílio à escrita científica, mais precisamente aquelas voltadas para o auxílio de escritores nativos da língua portuguesa.

[4] http://www.nilc.icmc.usp.br/scipo/

Referências

ALTHAUS, E.; KARAMANIS, N.; KOLLER, A. **Computing Locally Coherent Discourses** Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume. Barcelona, Spain: 2004.

ALUÍSIO, S.; SCHUSTER, E.; FELTRIM, V.; PESSOA, A.; OLIVEIRA, O. **Evaluating Scientific Abstracts with a Genre-specific Rubric** Proceedings of the 2005 Conference on Artificial Intelligence in Education: Supporting Learning Through Intelligent and Socially Informed Technology. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2005.

ALUÍSIO, S. M. **Ferramentas de auxilio a escrita de artigos científicos em inglês como língua estrangeira**. Tese de Doutoramento, Universidade de São Paulo, 1995.

ALUÍSIO, S. M.; BARCELOS, I.; SAMPAIO, J.; OLIVEIRA O.N. **How to learn the many unwritten "rules of the game" of the academic discourse: a hybrid approach based on critiques and cases to support scientific writing** Advanced Learning Technologies, 2001. Proceedings. IEEE International Conference on. 2001

ATTALI, Y.; BURSTEIN, J. Automated essay scoring with e-rater V. 2. **The Journal of Technology, Learning and Assessment**, v. 4, n. 3, 2006.

BARZILAY, R.; LAPATA, M. Modeling local coherence: An entity-based approach. **Computational Linguistics**, n. May 2007, 2008.

BICK, E. The parsing system Palavras : automatic grammatical analysis of Portuguese in a constraint grammar framework. **Aarhus University Press**, 2000.

BURSTEIN, J.; KUKICH, K.; WOLFF, S.; LU, C.; CHODOROW, M.; BRADEN-HARDER, L.; HARRIS, M. D. Automated Scoring Using a Hybrid Feature Identification Technique Proceedings of the 17th International Conference on Computational Linguistics - Volume 1: COLING '98. Stroudsburg, PA, USA: Association for Computational Linguistics, 1998.

BURSTEIN, J.; CHODOROW, M.; LEACOCK, C. Automated essay evaluation: The Criterion online writing service. **Ai Magazine**, v. 25, n. 3, p. 27, 2004.

BURSTEIN, J.; MARCU, D.; KNIGHT, K. **CriterionSM: Online essay evaluation: An application for automated evaluation of student essays** Proceedings of the fifteenth annual conference on innovative applications of artificial intelligence. 2003.

BURSTEIN, J.; TETREAULT, J.; ANDREYEV, S. Using entity-based features to model coherence in student essays. Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL. p.681–684, 2010. Los Angeles, California: Association for Computational Linguistics.

CARDOSO, P.; MAZIERO, E.; JORGE, M.; SENO, E.; DI FELIPPO, A.; RINO, L.; NUNES, M.; PARDO, T. Cstnews - a discourse-annotated *corpus* for single and multi-document summarization of news texts in brazilian portuguese. **the Proceedings of the 3rd RST Brazilian Meeting**, p. 88–105, 2011.

COLLOVINI, S.; CARBONEL, T.; FUCHS, J.; COELHO, J.; RINO, L.; VIEIRA, R. Summ-it : Um *corpus* anotado com informações discursivas visando a sumarização automática. **Anais do V Workshop em Tecnologia da Informação e da Linguagem Humana**, p. 1605–1614, 2007.

DEERWESTER, S.; DUMAIS, S.; FURNAS, G.; LANDAUER, T.; HARSHMAN, R. Indexing by Latent Semantic Analysis. **Journal of the American society for information science, Citeseer**, v. 41, p. 391–407, 1990.

DIAS, M. S.; FELTRIM, V. D.; PARDO, T. A. S. Using Rhetorical Structure Theory and Entity Grids to Automatically Evaluate Local Coherence in Texts. In: BAPTISTA, J. et al. (Eds.). . **Computational Processing of the Portuguese Language**. 11. ed. São Carlos/SP: Springer International Publishing, 2014. p. 232–243.

ELSNER, M.; CHARNIAK, E. Extending the entity grid with entity-specific features. **Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics**, p. 125–129, 2011.

FELTRIM, V. D. **Uma abordagem baseada em córpus e em sistemas de crítica para a construção de ambientes web de auxílio à escrita acadêmica em português**. Tese de Doutoramento, Universidade de São Paulo, São Carlos, SP, 2004.

FELTRIM, V.; PELIZZONI, J.; TEUFEL, S.; GRAÇA, M; ALUÍSIO, S. Advances in Artificial Intelligence -- SBIA 2004: 17th Brazilian Symposium on Artificial Intelligence, Sao Luis, Maranhao, Brazil, September 29-Ocotber 1, 2004. Proceedings. In: BAZZAN, A. L. C.; LABIDI, S. (Eds.). **Lecture Notes in Computer Science**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. p. 214–223.

FELTRIM, V. D.; TEUFEL, S.; NUNES, M. G. V.; ALUÍSIO, S. M. Argumentative Zoning Applied to Critiquing Novices' Scientific Abstracts. In: SHANAHAN, J. G.; Y., Q.; WIEBE, J. (Eds.). . **Computing Attitude and Affect in Text: Theory and Applications**. Dordrecht, The Netherlands: Springer Netherlands, 2006. p. 233–246.

FILIPPOVA, K.; STRUBE, M. Extending the Entity-grid Coherence Model to Semantically Related Entities. **ENLG '07 Proceedings of the Eleventh European Workshop on Natural Language Generation**, p. 139–142, 2007.

FOLTZ, P.; STREETER, L.; LOCHBAUM, K.; LANDAUER, T. Implementation and Applications of the Intelligent Essay Assessor. In: **Handbook of Automated Essay Evaluation - Current Applications and New Directions**. 1. ed. New York, NY, USA: Taylor & Francis, 2013. p. 68–88.

FREITAS, A. R. P. **Análise automática de coerência usando o modelo grade de entidades para o português** Maringá-PR, 2013.

FREITAS, A. R. P.; FELTRIM, V. D. Análise automática de coerência usando o modelo grade de entidades para o português. **Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology**, p. 69–78, 2013.

GARCIA, O. M. **Comunicacao Em Prosa Moderna**. 26. ed. Rio de Janeiro - RJ: EDITORA

FGV, 2006.

GOLUB, G. H.; REINSCH, C. Singular value decomposition and least squares solutions. **Numerische Mathematik**, v. 14, n. 5, p. 403–420, 1970.

GROSZ, B. J.; JOSHI, K. A.; WEINSTEIN, S. Centering : A Framework for Modeling the Local Coherence of Discourse. **Computational Linguistics**, v. 21, n. 1995, p. 203–225, 1995.

GUINAUDEAU, C.; STRUBE, M. Graph-based Local Coherence Modeling. **The 51st Annual Meeting of the Association for Computational Linguistics**, n. 2008, p. 93–103, 2013.

HASLER, L. An investigation into the use of Centering transitions for summarisation. **Proceedings of the 7th Annual CLUK Research Colloquium**, p. 100–107, 2004.

JOACHIMS, T. Training Linear SVMs in Linear Time. **Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**, p. 217–226, 2006.

KARAMANIS, N.; POESIO, M.; MELLISH, C.; OBERLANDER, J. **Evaluating Centering-based Metrics of Coherence for Text Structuring Using a Reliably Annotated *Corpus*** Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics. ACL '04.Stroudsburg, PA, USA: Association for Computational Linguistics, 2004.

KINNUNEN, T. et al. **SWAN - Scientific Writing AssistaNt: A Tool for Helping Scholars to Write Reader-friendly Manuscripts** Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics. EACL '12.Stroudsburg, PA, USA: Association for Computational Linguistics, 2012.

KINTSCH, W. The potential of latent semantic analysis for machine grading of clinical case summaries. **Journal of biomedical informatics, Elsevier**, v. 35, p. 3–7, 2002.

KOCH, I. G. V.; TRAVAGLIA, L. C. **A Coerência Textual**. 1. ed. São Paulo: Contexto, 2003.

KOCH, I. G. V.; TRAVAGLIA, L. C. **A Coerência Textual**. 17. ed. São Paulo - SP: Contexto, 2008.

KRIEGSMAN, M.; BARLETTA, R. Building a Case-Based Help Desk Application. **IEEE**

Expert: Intelligent Systems and Their Applications, v. 8, n. 6, p. 18–26, 1993.

LANDAUER, T.; FOLTZ, P.; LAHAM, D. **An introduction to latent semantic analysis** Discourse processes, , 1998.

LANDAUER, T. K.; DUMAIS, S. T. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. **PSYCHOLOGICAL REVIEW**, v. 104, n. 2, p. 211–240, 1997.

LIMA, A. DE O. **Interpretação de Textos - Aprenda Fazendo**. 1. ed. Rio de Janeiro - RJ: Elsevier Brasil, 2008.

LIN, Z.; NG, H. T.; KAN, M. Automatically Evaluating Text Coherence Using Discourse Relations. p. 997–1006, 2011.

MANN, W. C.; THOMPSON, S. A. Rhetorical structure theory: Toward a functional theory of text organization. **Text**, v. 8, n. 3, p. 243–281, 1988.

MILTSAKAKI, E.; KUKICH, K. **Evaluation of text coherence for electronic essay scoring systems**. Natural Language Engineering, 10, pp 25-55.

MÜLLER, A. L.; NEGRÃO, E. V; FOLTRAN, M. J. **Semântica formal**. São Paulo: Contexto, 2003.

POESIO, M.; STEVENSON, R.; DI EUGENIO, B.; HITZEMAN, J. Centering: A Parametric Theory and Its Instantiations. **Computational Linguistics**, v. 30, n. 3, p. 309–363, 2004.

QUINLAN, J. R. **C4.5: Programs for Machine Learning**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993. v. 240

RINO, L. H.; PARDO, T. A. A Coleção TeMário e a Avaliação de Sumarização Automática. 2007.

SCHULTZ, M. T. The IntelliMetric^{TM} Automated Essay Scoring Engine – A Review and an Application to Chinese Essay Scoring. In: **Handbook of Automated Essay Evaluation - Current Applications and New Directions**. 1. ed. New York, NY, USA: Taylor & Francis, 2013. p. 89–98.

SCHUSTER, E.; ALUÍSIO, S. M.; FELTRIM, V. D.; JR., A. P.; JR, O. N. O. **Enhancing the writing of scientific abstracts: a two-phased process using software tools and human evaluation,** Anais do Encontro Nacional de Inteligência Artificial (ENIA), 2005

SOUZA, V. M. A. DE; FELTRIM, V. D. A coherence analysis module for SciPo: providing suggestions for scientific abstracts written in Portuguese. **Journal of the Brazilian Computer Society**, v. 19, n. 1, p. 59–73, 23 jun. 2012.

SOUZA, V. M. A. **Análise automática de coerência semântica em recursos acadêmicos escritos em português** Maringá-PR, 2011.

SOUZA, V. M. A.; FELTRIM, V. D. Automatic Analysis of Semantic Coherence in Academic Abstracts Written in Portuguese. **International Joint Conference on Natural Language Processing**, p. 1144–1152, 2011.

SOUZA, V. M. A.; FELTRIM, V. D. A coherence analysis module for SciPo: providing suggestions for scientific abstracts written in Portuguese. **Journal of the Brazilian Computer Society**, p. 1–15, 2013.

STRUBE, M.; PONZETTO, S. P. WikiRelate ! Computing Semantic Relatedness Using Wikipedia. **Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2**, n. February, p. 1419–1424, 2006.

TEUFEL, S.; MOENS, M. Summarizing scientific articles: experiments with relevance and rhetorical status. **Computational Linguistics**, v. 28, n. 4, p. 409–445, 2002.

VAN DIJK, T. A. **Studies in the pragmatics of discourse**. Berlim/New York: Mouton Publishers, 1981.

VAN DIJK, T.; KINTSCH, W. **Strategies in Discourse Comprehension**. New York, Academic Press: Mouton Publishers, 1983.

WEBBER, B. L. D-LTAG: extending lexicalized TAG to discourse. **Cognitive Science**, v. 28, n. 5, p. 751–779, 2004.

WITTEN, I. H.; FRANK, E. **Data Mining - Practical Machine Learning Tools and**

Techniques. 2. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005.

YOKONO, H.; OKUMURA, M. Incorporating Cohesive Devices into Entity Grid Model. In: GELBUKH, A. (Ed.). **Computational Linguistics and Intelligent Text Processing**. Iasi, Romania: Springer Berlin Heidelberg, 2010. p. 303–314.

Apêndice A

Questionário Utilizado na Avaliação do Protótipo

01. Origem do Resumo:
( ) Artigo ( ) Dissertação Mestrado ( ) TCC ( ) Outro _____

02. Quantidade de Sentenças do resumo: _____

03. Antes da explicação, qual era o seu grau de conhecimento sobre a estrutura retórica do resumo acadêmico (Contexto, Lacuna, Propósito, Metodologia, Resultado e Conclusão)?
( )Avançado ( )Intermediário ( )Nenhum

04. Para cada sugestão de quebra de linearidade emitida pelo protótipo, responda as questões a seguir:

	Qual(is) é(são) o(s)componente(s)identificado(s) pelo protótipo? Ex.: Propósito + Metodologia	**Você acatou a sugestão do protótipo?**	**A sugestão é relevante para o seu resumo?**
1		☐SIM ☐NÃO ☐PARCIALMENTE	☐SIM ☐NÃO
2		☐SIM ☐NÃO ☐PARCIALMENTE	☐SIM ☐NÃO
3		☐SIM ☐NÃO ☐PARCIALMENTE	☐SIM ☐NÃO
4		☐SIM ☐NÃO ☐PARCIALMENTE	☐SIM ☐NÃO
5		☐SIM ☐NÃO ☐PARCIALMENTE	☐SIM ☐NÃO

05. Caso tenha acatado alguma das sugestões dadas pelo protótipo, como você considera o nível de coerência do seu resumo final em relação à primeira versão?
( ) Não alterou o nível de coerência
( ) Alterou razoavelmente o nível de coerência
( ) Alterou bastante o nível de coerência